Draft and Edit: Automatic Storytelling Through Multi-Pass Hierarchical Conditional Variational Autoencoder

Meng-Hsuan Yu,[1*] **Juntao Li,**[1,3*] **Danyang Liu,**[1,2] **Bo Tang,**[4] **Haisong Zhang,**[5]
Dongyan Zhao,[1,3] **Rui Yan**[1,2,3†]

[1]Wangxuan Institute of Computer Technology, Peking University, Beijing, China
[2]Key Laboratory of Artificial Intelligence, Ministry of Education, Shanghai Jiao Tong University, Shanghai, 200240
[3]Center for Data Science, AAIS, Peking University, Beijing, China
[4]Department of Computer Science and Engineering, Southern University of Science and Technology
[5]Tencent AI Lab
{yumenghsuan, lijuntao, zhaody, ruiyan}@pku.edu.cn,
danyliu@sjtu.edu.cn, tangb3@sustech.edu.cn, hansonzhang@tencent.com

Abstract

Automatic Storytelling has consistently been a challenging area in the field of natural language processing. Despite considerable achievements have been made, the gap between automatically generated stories and human-written stories is still significant. Moreover, the limitations of existing automatic storytelling methods are obvious, e.g., the consistency of content, wording diversity. In this paper, we proposed a multi-pass hierarchical conditional variational autoencoder model to overcome the challenges and limitations in existing automatic storytelling models. While the conditional variational autoencoder (CVAE) model has been employed to generate diversified content, the hierarchical structure and multi-pass editing scheme allow the story to create more consistent content. We conduct extensive experiments on the ROCStories Dataset. The results verified the validity and effectiveness of our proposed model and yields substantial improvement over the existing state-of-the-art approaches.

Christmas Shopping
Frankie had Christmas shopping to do.
She went to the store.
Inside, she walked around looking for gifts.
Soon her cart was full.
She paid and took her things home.

Table 1: An example of the story in the ROCStories dataset. Each story title is paired with five sentences.

Introduction

Automatic storytelling is on the frontier of natural language generation (Fan, Lewis, and Dauphin 2018). It generates a sequence of sentences which logically links events sets and shared characters throughout a passage. Due to the requirements of aesthetic and long-range dependency modelling, automatic storytelling needs to not only consider the semantic consistency among sentences, but also ensure wording diversity in the whole story. The above requirements are the main challenges in the automatic storytelling task.

To tackle the consistency challenge, most of the state-of-the-art approaches (Martin et al. 2018; Fan, Lewis, and Dauphin 2018; Xu et al. 2018; Yao et al. 2019; Li et al. 2019) exploit mid-level representations, such as keywords, events and skeleton, to provide better guidance during the story generation process. Even though these approaches have prominent performance, it is still insufficient to yield satisfactory results in general case as the intermediate representations are extracted independently, which did not consider the context. Specifically, these approaches may loss the consistency in the whole story and enlarge the interrelation gap between the story theme and the generated content when the representation is presented improperly, or not closely related to the story title. Moreover, the generation process of these approaches always requires multiple stages, in other words, it is not trained in an end-to-end manner. As a result, the model training is relatively complex.

On the other hand, to address the challenge of wording diversity, some related text generation works have proposed to use variational autoencoder(VAE) or conditional variational autoencoder (CVAE) model(Shen et al. 2017; Zhao, Zhao, and Eskenazi 2017; Serban et al. 2017) to generate diversified content. Even though both the VAE and CVAE model have been proved to model wording diversity successfully (Li, Luong, and Jurafsky 2015), as mentioned in (Li et al. 2013), consistency and wording diversity are to some extent, mutually exclusive. To be more specific, consistent stories may have restricted wording, while diversified wordings could lead to inconsistency. Apart from this, the current approaches are mainly trained in a single-pass manner, which makes the model fail to capture global information of the story.

In this paper, we proposed a multi-pass hierarchical CVAE generation model, targeting to enhance the quality of

[*] Equal contribution

[†] Corresponding author: Rui Yan (ruiyan@pku.edu.cn)

the generated story regarding attributes, including wording diversity and content consistency. We build our model based on the CVAE structure to benefit from the advantage of generating diversified words. Besides, to ensure the consistency at the same time, we modified the CVAE decoder by replacing the flat RNN decoder with a hierarchical structure that constantly guides the story generation process. Moreover, unlike traditional single-pass generation, we applied multi-pass generation to polish the generated story recursively. By doing so, it enhances the quality of the story incrementally.

To evaluate the effectiveness of our model, we conducted experiments on the ROCStories dataset(Mostafazadeh et al. 2016a). As shown in Figure 1, given the story title, we target to generate diversified and consistent stories. Experiment results have demonstrated that the stories generated by our proposed model have increased the diversity and consistency effectively, and yields substantial improvement over the existing state-of-the-art methods.

Preliminary

VAE and CVAE

Variational Autoencoder (VAE) (Kingma and Welling 2013) is one of the most popular neural networks for image generation (Yan et al. 2016). It is responsible for reconstructing a model or matching the target outputs to the provided inputs. The VAE contains an encoder and a decoder. While the encoder maps the input *X* into a latent variable *Z*, the decoder reconstructs the input *X* from the latent variable.

To be more specific, given the input *X*, the encoder computes a posterior distribution $q_\phi(z|x)$, regarded as the probability distribution of generating *Z* conditioned on input *X*. The decoder then computes the probability distribution $p_\theta(x|z)$ of *X* based on the latent variable *Z*. Noted that the latent variable *Z* is a standard Gaussian distribution and can be written as $p_\theta(z)$.

Through training the VAE model, we should ensure the reconstructed output to be as identical to the original input. Thus, the objective of the training should be maximizing the log-likelihood (log $p_\theta(x)$) over the input *X*. We now state the training objective of the model below:

$$\begin{aligned} L(\theta, \phi; x) = &- KL(q_\phi(z|x) \parallel p_\theta(z)) \\ &+ E_{q_\phi(z|x)}[log p_\theta(x|z)] \end{aligned} \tag{1}$$

Herein, the loss function can be divided into two parts, the KL-divergence loss (KL[·]) and the reconstruction loss (E[·]). The KL[·], can be viewed as the regularization for encouraging the distribution of the approximated posterior $q_\phi(z|x)$ to be as close as the prior $p_\theta(z)$. Besides, the E[·] can be regarded as a guidance of the decoding process, which shows how well the model does the reconstruction work.

Based on the above concept of VAE, a Conditional Variational Autoencoder (CVAE) is introduced. As a modified model of VAE, CVAE has a similar structure as the VAE; however, each component of the model is conditioned on an additional condition *C*. The objective of the CVAE is similar to VAE, which also targeting to maximize the log-likelihood over input *X*; however, all items are introduced with an additional attribute *C*. The loss function of CVAE is stated below:

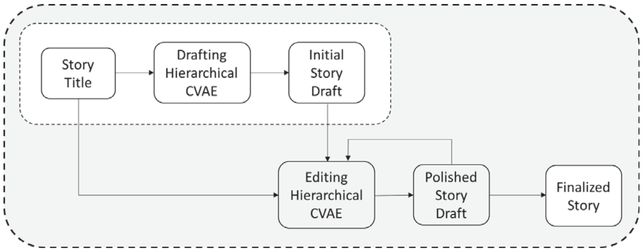

Figure 1: The overview of our proposed model. The drafting H-CVAE generates the initial story draft and further pass it to the editing H-CVAE for polishing.

$$\begin{aligned} L(\theta, \phi; x, c) = &- KL(q_\phi(z|x, c) \parallel p_\theta(z|c)) \\ &+ E_{q_\phi(z|x,c)}[log p_\theta(x|z, c)] \end{aligned} \tag{2}$$

Moreover, since CVAE is known to encounter the vanishing latent variable problem (Bowman et al. 2015), we apply the same strategy as (Zhao, Zhao, and Eskenazi 2017) to tackle this issue. Specifically, we optimize the bag-of-words loss to force the model to capture global information of the target output.

Problem Formulation

Following the prior works (Fan, Lewis, and Dauphin 2018; Yao et al. 2019; Xu et al. 2018; Li et al. 2019), we state our task with a similar problem setting, where given a story title, the model should generate the corresponding story. We define the problem as follows:

- Input: The story title T = (w_1, w_2, w_3, ..., w_n), where w_i represents the i-th word and n denotes to the length of the given title.
- Output: The story S_1,S_2, ..., S_N, where S_i represents the i-th sentence of the story. Each sentence in the story is represented as S_i = ($w_{i,1}$, $w_{i,2}$, $w_{i,3}$, ...,$w_{i,m}$); $w_{i,j} \in V$, where $w_{i,j}$ is the j-th word in the i-th sentence. Noted that, in our case, we target to generate stories with exactly five sentences.

The Model

As shown in figure 1, our model works in an encoding-decoding manner. We take the story title as the input targeting to generate a consistent story. Specifically, the model is built upon two hierarchical CVAE (H-CVAE) models. The drafting H-CVAE is responsible for generating a story draft, and the editing H-CVAE takes the story draft as an input to generate a polished story in a multi-pass manner. In below, we will elaborate on the implementation of the H-CVAE and multi-pass editing scheme.

Hierarchical CVAE

The hierarchical CVAE is a modified model of the CVAE. It consists of an encoder and a hierarchical decoder which

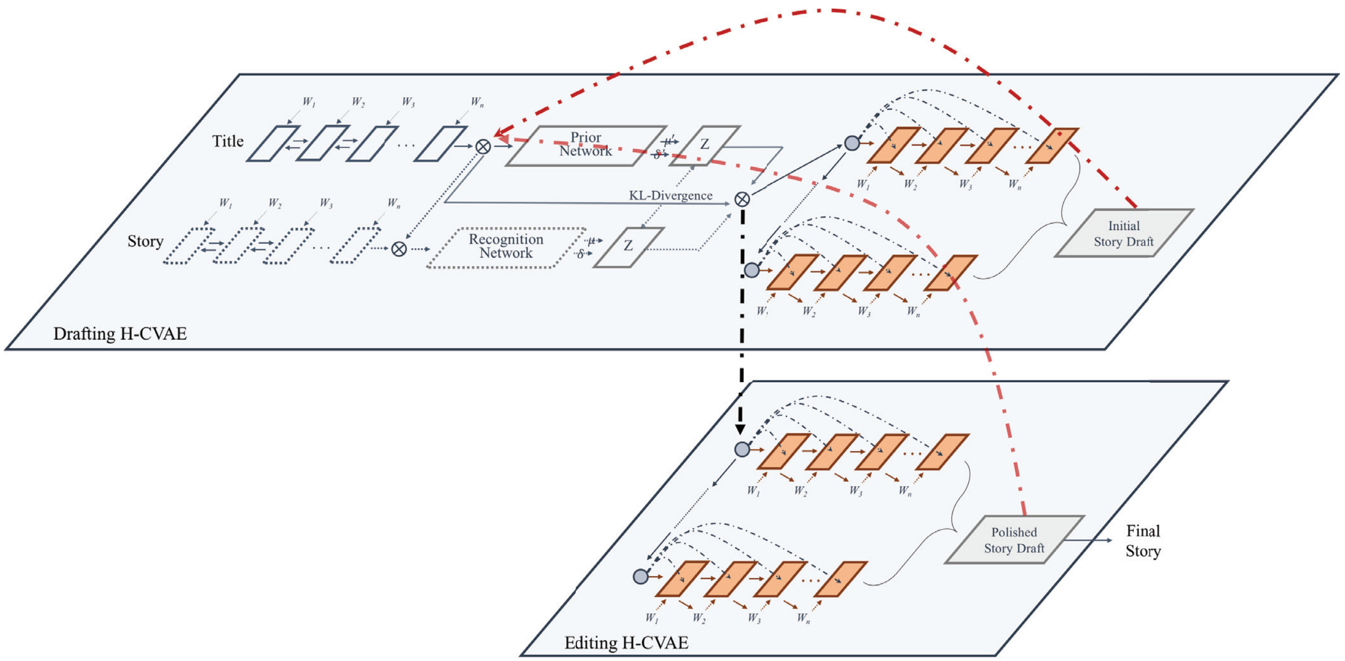

Figure 2: The detail presentation of the Multi-Pass Hierarchical Conditional Variational Autoencoder. The upper area denotes to the drafting H-CVAE; The lower area denotes to the editing H-CVAE; The entire H-CVAE model is used during the training stage, while only the part with solid lines is used during the testing stage; the shaded blue circles indicate the global hierarchies hidden states; the shaded orange square indicates the local hierarchies hidden states;

play as the core part of our model to generate stories. The encoder is a bidirectional RNN (Berglund et al. 1997) with gated recurrent units (GRU)(Cho et al. 2014), that encodes the input *X*, and the condition *C* into the following representation respectively: $X = [\overrightarrow{x}, \overleftarrow{x}]; C = [\overrightarrow{c}, \overleftarrow{c}]$.

Following the setting of prior works (Kingma and Welling 2013; Zhao, Zhao, and Eskenazi 2017; Yang et al. 2018), we assume the variational approximate posterior is a multivariate Gaussian N with a diagonal covariance structure $q_\phi(z|x, c) = N(\mu, \sigma^2 I)$, which can be computed by the below equation:

$$\begin{bmatrix} \mu \\ log(\sigma^2) \end{bmatrix} = W_q \begin{bmatrix} x \\ c \end{bmatrix} + b_q \tag{3}$$

Similarly, the prior $p_\theta(z|c)$ can be formulated as another multivariate Gaussian $N(\mu', \sigma'^2 I)$ which is computed by a single-layer fully-connected neural network (MLP) with the tanh(·) activation function:

$$\begin{bmatrix} \mu' \\ log(\sigma'^2) \end{bmatrix} = MLP_p(c) \tag{4}$$

After encoding, we then take [*Z*, *C*] as the input of the decoder to generate a story. As mentioned before, instead of using a flat RNN decoder, we replaced it by a hierarchical structure. Specifically, the hierarchical decoder consists of one global-level RNN decoder and N local-level RNN decoders (in our case, N=5).

As shown in figure 2, the global-level RNN decoder captures the global information of the story and also leads the generation process of every single sentence. The initial hidden state of the global RNN is the previous calculated [*Z*, *C*]. Through the generation process, the hidden state of the global-level RNN will be sequentially updated by concatenating the last hidden state of the previous generate sentence.

$$h_i^{<global>} = f(W_x h_{i-1}^{<local>} + W_y h_{i-1}^{<global>}) \tag{5}$$

where h<local>is the last hidden state of the local-level RNN in the lower hierarchy.

The local-level RNN decoder serves as the guidance of the sentence generator. It captures the lower-level information and is responsible for generating every single word. The input of the local-level RNN includes the word embedding of the previously generated word, the previous RNN hidden state and the global state of the current sentence.

$$h_i^{<local>} = f(W_g X_{i-1} + W_x h_{i-1}^{<local>} + W_y h_i^{<global>}) \tag{6}$$

With the interactive generation process of the decoder, it will eventually generate the whole story.

Multi-pass Editing Scheme

Generally, the process of writing a story is not in a single-pass manner. The preliminary draft is usually written first, then multiple polishing of the draft is required to enrich the content. We were inspired by the natural process a human would take to write a story. As such, we integrated a multi-pass generation mechanism in our model to mimic this human behavior.

To implement multi-pass generation, we integrated two H-CVAEs, one for creating the story draft and the other for recursively polishing the story draft to increase the quality of the story. Figure 2 shows the detailed implementation of our proposed model. We will further elaborate on the details of the drafting and editing H-CVAE below.

Drafting H-CVAE, takes the story title *T* as the input and pass it through a H-CVAE model to generate the first draft *D*. This story draft will then be passed to the editing H-CVAE. Through this process, it enables the editing H-CVAE to gain the knowledge of the present situation of the story draft.

Editing H-CVAE, unlike the drafting H-CVAE, the editing H-CVAE takes both the story title *T* and previously generated story draft *D* as an input (see the red dotted line in figure 2) targeting to generate a polished story *S*. Since it gathered information from not only the story title but also the initial generated story draft, this facilitates an overall semantic consistency for the generated content and ensures the polished story to not deviate from the story theme.

$$C = \begin{cases} T[\overrightarrow{t}, \overleftarrow{t}] & \text{(Drafting H-CVAE)} \\ T[\overrightarrow{t}, \overleftarrow{t}] + D[\overrightarrow{d}, \overleftarrow{d}] & \text{(Editing H-CVAE)} \end{cases} \tag{7}$$

Noted that the editing process can be recursively implemented by passing the polished story draft back as the input of the editing H-CVAE(see the light red dotted line in figure 2). However, it is still necessary to incorporate a termination schedule for the generator after repeated polishing. We empirically stop the editing process after three iterations.

Experiment

Dataset

We trained our model using the ROCStories Corpus [1] as our dataset. The ROCStories corpus contained short stories with rich content of causal and temporal common-sense relations of daily events. Since the corpus has a high-quality collection of real-life stories, it serves as an adequate training resource for automatic story generation models. The statistic of the ROCStories corpus is shown in table 2. In total, the ROCStories corpus has 98,163 stories. Each story contains one story title and five sentences. For training, we preprocessed the ROCStories by applying NLTK for tokenization and split the processed data into 8:1:1 for training, validation and testing.

Baselines

To evaluate the effectiveness of our proposed model, we have compared our methods with the following baselines:

S2S, the conventional Sequence to Sequence model (Sutskever, Vinyals, and Le 2014), that has been applied to a significant number of natural language generation tasks and proved to be effective in many related works. Therefore, we have employed the Seq2Seq model to generate the story; to do so, the title is used as the input of the model to generate stories sentence by sentence.

Numbers of Stories	98161
Average number of words (title)	2
Average number of words (story)	43
Vocabulary Size	35,595

Table 2: Statistics of the ROCStories Corpus

AS2S, denotes to the sequence to sequence model with the attention mechanism(Bahdanau, Cho, and Bengio 2014). Since it serves as the benchmark of various language generation tasks and recent automatic story generation approaches are built upon it, we use it as our baseline (Jain et al. 2017; Martin et al. 2018). The implementation pipeline is the same as the S2S.

CVAE, is the conventional Conditional Variational Autoencoder(CVAE), which uses the same generation pipeline as the S2S. This baseline is used to investigate the performance of our proposed model without applying the hierarchical decoder and multi-pass editing scheme.

Hierarchical, is one of the top-performing story generation model(Fan, Lewis, and Dauphin 2018). This baseline is built upon a Convolutional Seq2Seq model with model fusion, and self-attention mechanisms applied. Given the story title, the model should generate the corresponding story.

Plan and Write, is also one of the top-performing story generation model(Yao et al. 2019). This model first extracts the storyline from the given title then further generates the story based on it.

Model Settings

Our proposed model is trained under the following parameters and hyperparameters. We set the word embedding size to 300 and the hidden state dimension of both the encoder and decoder to 500. The vocabulary size of the model is limited to the most frequent 30,000 words. For the prior network, we set the hidden state dimension to 400.

Furthermore, the size of the latent variable *Z* is set to 300. All initial weights are initialized from a uniform distribution: $[-0.08, 0.08]$. To optimize our model, we have applied the Adam optimizer (Kingma and Ba 2014) and set the mini-batch size to 80. To avoid gradient explosion, the gradient clipping strategy is applied (Pascanu, Mikolov, and Bengio 2013), and the clipping value is set to 5. Lastly, we train our model with the learning rate of 0.001.

Evaluation Metrics

To evaluate the effectiveness of the model from the generated stories, we have employed the following evaluation metrics:

BLEU, is an automatic metric that has been widely used to evaluate machine translation models (Papineni et al. 2002). The scenario of the BLEU is that it measures the degree of overlapped words between the generated sentence and the ground truth sentence.

Even though BLEU is not a precise evaluation metric for story generation task since it cannot quantify the "creativity" of the story. As automatic storytelling is an NLG related

[1] http://www.cs.rochester.edu/nlp/rocstories/

Model	BLEU				Distinctness				Embedding-based Metric		
	B-1	B-2	B-3	B-4	D-1	D-2	D-3	D-4	Greedy	Average	Extrema
S2S	23.34	8.46	3.50	1.62	0.60	2.22	5.19	8.87	0.91	22.39	0.40
AS2S	24.82	7.05	2.34	0.90	0.73	4.13	8.66	12.16	0.93	21.30	0.33
CVAE	25.09	8.28	2.73	1.01	1.89	17.54	46.41	73.12	0.91	22.19	0.34
Hierarchical	15.01	6.21	2.64	1.22	1.57	13.36	36.84	62.03	0.93	22.80	0.42
Plan and Write	27.49	11.65	5.44	2.77	0.82	4.62	12.36	23.55	0.94	23.18	0.44
H-CVAE	29.41	11.20	4.37	1.85	1.88	12.33	35.03	61.02	0.95	23.04	0.39
H-CVAE (Multi-pass)	29.39	11.02	4.25	1.76	1.99	14.82	40.73	67.40	0.95	23.04	0.39

Table 3: Automatic Evaluation Results: B-n represents BLEU scores on n-gram (n = 1 to 4); D-n denotes to the distinctness score of n-gram (n = 1 to 4); Greedy, Average, Extrema represents the Greedy Matching, Embedding Average and Vector Extrema of the embedding-based metrics.

task, we regard BLEU as a fundamental indicator and can be a preliminary standard for our generated stories. In addition, since this metric has been applied in other credible text generation works such as (Martin et al. 2018; Xu et al. 2018; Li et al. 2018; Yao et al. 2019), we believe that this is a fair point for comparison among different methods.

Distinctness, measures the wording diversity by calculating the distinctive n-grams (1<n<4) of the generated sentence. To evaluate the diversity of the generated story, we have applied this metric as one of our evaluation methods.

Embedding-based Metrics, are used to measure the distance between given sentences (Liu et al. 2016). Unlike the word-overlap based metrics, it takes account of the semantic level information of the given sources by using different calculation methods, including Greedy Matching, Embedding Average and Vector Extrema. By employing these metrics, we could show how the generated stories are semantically close to the provided ground truth stories. [2]

Human Evaluation, is used to evaluate the story in the perspective of human beings. We asked four well-educated humans to score the stories generated from the models. As shown in table 5, they score the story in three criteria, including the readability, consistency and wording diversity of the story. Each criteria is annotated with three score levels: 1, 2, and 3; the higher, the better. In total, 100 randomly selected stories for each model are evaluated blindly.

Model	Rd.	Con.	Div.	Avg.
S2S	1.61	1.48	1.37	1.49
AS2S	1.55	1.18	1.52	1.42
CVAE	1.68	1.61	2.19	1.83
Hierarchical	2.29	2.02	1.93	2.08
Plan and Write	1.86	1.85	1.59	1.77
H-CVAE	2.25	2.22	2.10	2.19
H-CVAE (Multi-pass)	2.14	2.20	2.31	2.22

Table 4: Human Evaluation Results: Rd., Con., Div., Avg. represents readability, consistency, wording diversity, and average score of the stories.

Readability	Is the story grammatically formed?
Consistency	Does the story display a consistent theme?
Wording Diversity	Does the story narrated with diversified wording?
Average Score	The average score of the above three criteria.

Table 5: Three Criteria of human evaluation

Results and Discussion

In this paper, we proposed a multi-pass hierarchical CVAE model to address the deficiency of the existing story generation model. Below, we will analyze the model in the following perspectives.

The effect of the Hierarchical CVAE

As previously mentioned, we introduced the CVAE model to increase the wording diversity of the generated story. Demonstrated in table 3, by looking at the distinctness score of the CVAE, H-CVAE and H-CVAE (Multi-pass) models, we can confirm that through applying the CVAE model, it effectively increased the wording diversity of the generated story and showed the validity of introducing CVAE as the basic model for addressing the issue of common wording in the conventional Seq2Seq model. Such improvements is also confirmed by the consistency score in table 5.

However, since wording diversity and content consistency are, to some extent, mutually exclusive (Li et al. 2013). To resolve this issue, we replaced the flat RNN decoder by a hierarchical decoder which captures the global and local information of the story through a sequential generation process. By looking at the results in table 3 and 4, we can confirm that through combining the hierarchical decoder with the CVAE model, it resolved the main shortcoming of the conventional CVAE model and effectively enhanced the consistency of the generated story.

The effect of the Multi-pass Generation Mechanism

Prior works on story generation mainly trains the model in a single-pass manner; however, it somehow is conflicted to how human writes a story. To mimic how human writes a story, we introduced the multi-pass editing scheme to our

[2] Following the setting in (Yao et al. 2019), we adopt the pre-trained Glove embedding to measure the correlation. http://nlp.stanford.edu/ data/glove.840B.300d.zip

Story Title: Baseball Game	
CVAE	Fred loves to play the game. He tries to play it. Dan takes a long hike to the barbershop. One morning of the wedding is due to his diligence. Mike had to purchase a new smartwatch.
H-CVAE	My brother taught how to play baseball. He practiced every day for a while. He practiced very hard. A year later, he got a good score of his hits. He won a lot.
H-CVAE (Multi-pass)	Toby wanted to play baseball. He went to his first baseball game. He played the slots in his game. He scored all over his game. His team won the game and won the game.
Story Title: The School Award	
CVAE	Barry was very nervous for his first child in English. He taught him to keep the secret. He is still convinced he is a good rapper. Every day he sees that his favorite tutorial is raw. He does not tell his friends how he could do it.
H-CVAE	Kyra's teacher announced a letter. He had to get his wife's attention. He hated the preacher. He told her he was a charming person. Cedric's teacher gave him extra credit.
H-CVAE (Multi-pass)	It was the first day of the awards ceremony when her teacher announced it. All her friends had been assigned her to the teacher for her. She had to explain her team to do her performance. The class was very good. Everyone cheered her.

Table 6: Case studies of the generated stories.

model. Through the results of both automatic evaluation and human evaluation, we can discover that while not damaging the consistency of the story, through multi-pass generation, it somehow increases the diversity compared to the H-CVAE model. The result proves that the multi-pass generation mechanism is able to polish the generated story draft by enriching the content effectively.

Discussion

Based on the results on the ROCStories dataset, our proposed model substantially outperforms the baselines in both the automatic metrics and human evaluation. As shown in table 3, we observed that the H-CVAE model and H-CVAE (multi-pass) model has an excellent improvement over the sequence to sequence model in both the BLEU scores and distinctness score. This confirms the validity of the CVAE model regarding generating diversified stories and shows that through combining a hierarchical decoder with the CVAE, it does effectively enhance the consistency of the generated content.

On the other hand, compared to the state-of-the-art methods, our proposed model has excellent improvements over the hierarchical convolutional Seq2Seq model(Fan, Lewis, and Dauphin 2018) and has substantial improvements over the plan and write model (Yao et al. 2019) regarding the BLEU score. For the distinctness of the generated stories, our model has greatly improved all the existing story generation approaches.

Last but not least, based on the results of the embedding-based metrics, we discover that our multi-pass H-CVAE model has comparable results over the baseline models. It once again confirmed the model still maintains the consistency even using a CVAE as the basis model.

Case Study

Table 6 shows two examples of the generated stories from the CVAE model, the H-CVAE model and the H-CVAE model with the multi-pass editing scheme. By comparing our model with the CVAE baseline, we would like to show how the hierarchical decoder and multi-pass generation process influences the quality of the story. By looking at the generated examples, we can easily observe that our proposed model had outperformed the CVAE model. Through looking at details, based on the stories generated from the CVAE model, even though it has a relatively diversified content, it somehow sacrificed the consistency throughout the story. The first two sentences surely are theme-related, however starting from the third sentence, the story starts to generate contents that are totally irrelevant to the title (i.e. barbershop, wedding, smartwatch).

When we look at the stories generated from H-CVAE, even the diversity of the story is relatively low compared to the CVAE ones; it has quite an impressive consistent content. Moreover, through the multi-pass editing scheme, the polished story has a better flow and longer content. The second example (i.e. the school award) also showcased the mentioned improvements.

However, in our experiments, we also discover some inferior cases, which is shown in table 7. Specifically, we found two significant problems from the generated stories. The first problem is that the generated story is lack of emotion or sentiment. By looking at the example, we can observe that even the content and the story title is well connected, it somehow is just describing an event without emotional ups and downs. To tackle such issue, we could consider to add sentimental/emotional information to the model and ensure the story is generated with an emotion flow. The second problem is that there exist several cases that are off-topic. By looking at the example, we can see that the story of the title ”Shaving Accident” has not mentioned or describe relevant content to the word ”shaving”, but instead paid more attention on generating stories related to ”accident”. To address this problem, we could apply better topic modelling methods to our model to enable the model to learn a better topic representation. We could also introduce reinforcement learning (RL) or generative adversarial (GAN) (Goodfellow et al. 2014) network to our model for further improvements, which points out the direction of future work.

Lack of Sentiment/Emotion	
Riding a Bike	Betsy and her dad were riding a bike. They went to the pound. They went to the pound and got a bike. The bike would not stop. Tre got the bike and rode it home.
Off Topic	
Shaving Accident	Tim was at work when a new job was in the shop. He figured it would be faster than usual. He started to notice the other way it was getting worse. One day, he got injured and fell off. Kate decided to be more careful about quitting.

Table 7: Inferior cases of the generated stories.

Related Work

Neural Story Generation

Automatic story generation can be dated back to the 1970’s(Meehan 1977). Early story generation methods focused on planning-based or case-based approaches (Riedl and Young 2010; Gervás et al. 2004; Montfort, Marcus, and Prince 2007). Though these methods have had impressive performance, they were still restricted to specific domain. To further improve the adaptability of the models, approaches such as (Li et al. 2013) are proposed to generate stories from unknown domains.

Recently, many researches have focused on structure planning or applied additional content to ensure the consistency of the automatic generated story. For instance, Martin et.al (2018) has presented an automatic story generation framework that decomposed the story generation process into two steps. The story event is first generated from plain text, then based on successive events, the stories are further composed. Fan et.al (2018) proposed a hierarchical framework, which is built upon the convolutional sequence to sequence model for improving the consistency and creativity of the stories. Xu et.al (2018) generated stories from pre-extracted skeleton that is learned by a reinforcement learning method. Yao et.al (2019) focused more on story planning, which plans the storyline first and then generates the story based on it.

The methods mentioned above focused mainly on using intermediate representation to assist the process of story generation. This restricts the expansibility of the generated story, and is also risky in situations when mid-level representation are not adequately planned.

To ensure the story are diversified while remaining consistency, our proposed model made use of the conditional variational autoencoder. However, we replaced the decoder by a hierarchical structure and applied multi-pass editing scheme to tackle with these issues.

VAE and CVAE

Variational autoencoder (VAE) proposed by (Kingma and Welling 2013) has been applied to a great numbers of applications in different tasks such as image generation (Yan et al. 2016), machine translation (Zhang et al. 2016) and dialogue generation (Serban et al. 2017; Shen et al. 2017). Based on the founding in previous research, VAE and CVAE based models have relatively good performances in text generation (Serban et al. 2017; Shen et al. 2017; Zhao, Zhao, and Eskenazi 2017; Li et al. 2018; 2019; Qiu et al. 2019; Chan et al. 2019). Specifically, through combining Seq2Seq models with latent variable representations, it has been proved to have outstanding ability in managing properties including wording diversity. (Li, Luong, and Jurafsky 2015; Mostafazadeh et al. 2016b). Since story generation drives concerns regarding the above-mentioned properties, we have chosen to apply CVAE as the basis of our model.

Conclusion

Automatic Storytelling has consistently been a challenging area in the field of natural language processing. In this paper, we proposed a multi-pass hierarchical CVAE model which combines a hierarchical CVAE model with a multi-pass editing scheme. Specifically, we utilized the CVAE to generate diversified stories. To further improve the consistency of the generated content, we replaced the flat RNN decoder of the CVAE with a hierarchical structure, that captures both global and local information of the story. Moreover, we applied the multi-pass mechanism to polish the generated initial story draft recursively. By utilizing global information, the multi-pass editing scheme provides better guidance on the generation process. Experimental results on the ROCStories dataset indicates that our model can generate stories with enhanced wording diversity and consistency and have substantial improvements over both the baseline and the existing state-of-the-art methods.

Acknowledgments

We would like to thank the reviewers for their constructive comments. This work was supported by the National Key Research and Development Program of China (No. 2017YFC0804001), the National Science Foundation of China (NSFC No. 61876196 and NSFC No. 61672058) and the foundation of Key Laboratory of Artificial Intelligence, Ministry of Education, P.R. China. Rui Yan was sponsored by Tencent Research Fund and Beijing Academy of Artificial Intelligence (BAAI). Bo Tang was supported by Guangdong Natural Science Foundation (Grant No. 2018A030310129), the National Science Foundation of China (NSFC No. 61802163) and PCL Future Regional Network Facilities for Large-scale Experiments and Applications (PCL2018KP001).

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Berglund, M.; Raiko, T.; Honkala, M.; Kärkkäinen, L.; Vetek, A.; and Karhunen, J. 1997. Bidirectional recurrent neural networks as generative models - reconstructing gaps in time series. *IEEE Transactions on Signal Processing*.

Bowman, S. R.; Vilnis, L.; Vinyals, O.; Dai, A. M.; Józefowicz, R.; and Bengio, S. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.

Chan, Z.; Li, J.; Yang, X.; Chen, X.; Hu, W.; Zhao, D.; and Yan, R. 2019. Modeling personalization in continuous space for response generation via augmented wasserstein autoencoders. In *EMNLP*.

Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*.

Fan, A.; Lewis, M.; and Dauphin, Y. 2018. Hierarchical neural story generation. In *ACL*.

Gervás, P.; Díaz-Agudo, B.; Peinado, F.; and Hervás, R. 2004. Story plot generation based on cbr. In *ICIAI*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NIPS*.

Jain, P.; Agrawal, P.; Mishra, A.; Sukhwani, M.; and Sankaranarayanan, K. 2017. Story generation from sequence of independent short descriptions. *CoRR abs/1707.05501*.

Kingma, D. P., and Ba, J. 2014. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Li, B.; Lee-Urban, S.; Johnston, G.; and Riedl, M. 2013. Story generation with crowdsourced plot graphs. In *AAAI*.

Li, J.; Song, Y.; Haisong, Z.; Chen, D.; Shi, S.; Zhao, D.; and Yan, R. 2018. Generating classical chinese poems via conditional variational autoencoder and adversarial training. In *EMNLP*.

Li, J.; Bing, L.; Qiu, L.; Chen, D.; Zhao, D.; and Yan, R. 2019. Learning to write stories with thematic consistency and wording novelty. In *AAAI*.

Li, J.; Luong, M.-T.; and Jurafsky, D. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *ACL*.

Liu, C.-W.; Lowe, R.; Serban, I. V.; Noseworthy, M.; Charlin, L.; and Pineau, J. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. *arXiv preprint arXiv:1603.08023*.

Martin, L. J.; Ammanabrolu, P.; Wang, X.; Hancock, W.; Singh, S.; Harrison, B.; and Riedl, M. O. 2018. Event representations for automated story generation with deep neural nets. In *AAAI*.

Meehan, J. R. 1977. Tale-spin, an interactive program that writes stories. In *IJCAI*.

Montfort, N.; Marcus, M. P.; and Prince, G. 2007. *Generating narrative variation in interactive fiction*. University of Pennsylvania Philadelphia, PA.

Mostafazadeh, N.; Chambers, N.; He, X.; Parikh, D.; Batra, D.; Vanderwende, L.; Kohli, P.; and Allen, J. F. 2016a. A corpus and evaluation framework for deeper understanding of commonsense stories. In *NAACL*.

Mostafazadeh, N.; Grealish, A.; Chambers, N.; Allen, J.; and Vanderwende, L. 2016b. Caters: Causal and temporal relation scheme for semantic annotation of event structures. In *NAACL*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Pascanu, R.; Mikolov, T.; and Bengio, Y. 2013. On the difficulty of training recurrent neural networks. In *ICML*.

Qiu, L.; Li, J.; Bi, W.; Zhao, D.; and Yan, R. 2019. Are training samples correlated? learning to generate dialogue responses with multiple references. In *ACL*.

Riedl, M. O., and Young, R. M. 2010. Narrative planning: Balancing plot and character. *JAIR*.

Serban, I. V.; Sordoni, A.; Lowe, R.; Charlin, L.; Pineau, J.; Courville, A.; and Bengio, Y. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*.

Shen, X.; Su, H.; Li, Y.; Li, W.; Niu, S.; Zhao, Y.; Aizawa, A.; and Long, G. 2017. A conditional variational framework for dialog generation. In *ACL*.

Sutskever, I.; Vinyals, O.; and Le, Q. 2014. Sequence to sequence learning with neural networks. *NIPS*.

Xu, J.; Ren, X.; Zhang, Y.; Zeng, Q.; Cai, X.; and Sun, X. 2018. A skeleton-based model for promoting coherence among sentences in narrative story generation. In *EMNLP*.

Yan, X.; Yang, J.; Sohn, K.; and Lee, H. 2016. Attribute2image: Conditional image generation from visual attributes. In *ECCV*.

Yang, X.; Lin, X.; Suo, S.; and Li, M. 2018. Generating thematic chinese poetry using conditional variational autoencoders with hybrid decoders. In *IJCAI*.

Yao, L.; Peng, N.; Weischedel, R.; Knight, K.; Zhao, D.; and Yan, R. 2019. Plan-and-write: Towards better automatic storytelling. In *AAAI*.

Zhang, B.; Xiong, D.; Su, J.; Duan, H.; and Zhang, M. 2016. Variational neural machine translation. In *EMNLP*.

Zhao, T.; Zhao, R.; and Eskenazi, M. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*.