Variational Attention for Commonsense Knowledge Aware Conversation Generation

Guirong Bai[1,2], Shizhu He[1], Kang Liu[1,2], and Jun Zhao[1,2]

[1] National Laboratory of Pattern Recognition Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China
[2] University of Chinese Academy of Sciences, Beijing, 100049, China
{guirong.bai,shizhu.he,kliu,jzhao}@nlpr.ia.ac.cn

Abstract. Conversation generation is an important task in natural language processing, and commonsense knowledge is vital to provide a shared background for better replying. In this paper, we present a novel commonsense knowledge aware conversation generation model, which adopts variational attention for incorporating commonsense knowledge to generate more appropriate conversation. Given a post, the model retrieves relevant knowledge graphs from a knowledge base, and then attentively incorporates knowledge to its response. For enhancing attention to incorporate more clean and suitable knowledge into response generation, we adopt variational attention rather than standard neural attention on knowledge graphs, which is unlike previous knowledge aware generation models. Experimental results show that the variational attention based model can incorporate more clean and suitable knowledge into response generation.

Keywords: Conversation generation · Commonsense knowledge · Variational attention.

1 Introduction

Commonsense knowledge is a key factor for conversational systems. Without commonsense knowledge background, it may be difficult to understand posts and generate responses in conversational systems [16, 15, 23, 18]. For instance, to understand the post "did you use color pencils?" and then generate the response "sure did mate. green and blue as well as grey lead", we need the relevant commonsense knowledge, such as (green, RelatedTo, color), (blue, RelatedTo, color), (grey, RelatedTo, color) and (lead, RelatedTo, pencils). It is shown in Figure 1.

Recently, [29] first attempted to incorporate commonsense knowledge into conversation generation. Given a post, the model attentively reads corresponding knowledge graphs at every step, and establishes effective interaction between posts and responses. Concretely, commonsense knowledge is incorporated with an attention mechanism [1] in sequence-to-sequence model [24].

However [29] only considers the recall of attentional knowledge and ignores its precision, and the models are very likely to incorporate unsuitable knowledge

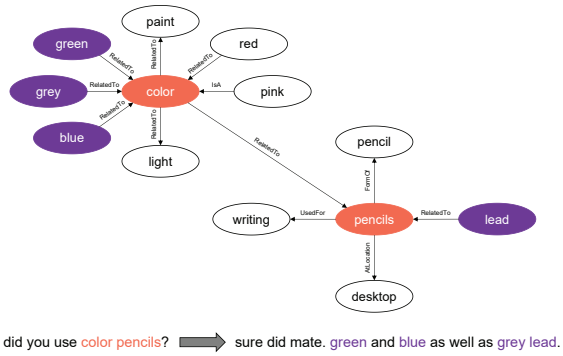

Fig. 1. Example of how commonsense knowledge facilitates post understanding and response generation in conversation. Every pair of nodes formulates a triple.

into response generation. In machine translation, the semantic representations of targets are fixed and what should be generated are certain before decoding, so the decoder can attentively read source words base on unfinished generation. But attention on knowledge graphs is different. The response to the post is not certain, and we can't ensure which background knowledge needs to share at current step. Thus, unless we know what the whole response will say, we can't ensure which background knowledge needs to be shared at current step. For example, as shown in Figure 2, when generating the word y_6, based on the complete output we can know it should be an entity around the "color" node following "and" rather than one around the "pencil" node, so that the attention probability on the knowledge graph of the "color" node should be more heavily weighted.

Thus, there is a valuable challenge of how to skillfully utilize the complete output information to help the model ensure which background knowledge needs to share. To this end, we propose to use variational attention rather than standard attention mechanism on knowledge graphs. Concretely, we first model posterior distributions of attention on knowledge graphs, which contain the complete output words as a condition. Then, corresponding prior distributions without output are enhanced by KL loss with the posterior distributions. In this way, attention on knowledge graphs can be enhanced by KL loss with the complete output information.

In brief, our main contribution is that we propose a variational attention approach for commonsense knowledge aware conversation generation. In addition, we also implement an extra evaluation for the precision of the incorporated knowledge facts. Experimental results show that the proposed method is able to incorporate more clean and suitable knowledge.

2 Related Work

Data Driven Conversation Generation
Recently, sequence-to-sequence models [24] make effects on large-scale conversation generation, such as neural responding system [21, 22], hierarchical recurrent models [19, 20] and many others. Some studies attempted to facilitate improvement for the content quality of generated responses, including promoting diversity [11, 12, 27], considering additional information such as topic [25] or keyword [17], dealing with out-of-vocabulary words [8], and so on.
Knowledge Based Conversation Generation
External knowledge incorporated into conversation generation can be divided into two types. One belongs to unstructured texts. [7] improved conversation generation with memory network which stores relevant comments left by customers as external facts. [13] generated multi-turn conversations with a search engine capturing external knowledge which is encoded by convolutional neural network. The other belongs to structured knowledge. [26] use a recall-gate mechanism to incorporate structured domain-specific knowledge base. [9] and [30] presented an end-to-end knowledge grounded conversational model with a copy network [8]. [29] presented a novel open-domain conversation generation model with commonsense knowledge.
Variational Methods
Recently, variational methods have shown great promise in natural language processing, such as modeling topic, emotion, style, intention or others in conversation for more meaningful generation [4, 20, 28, 2], latent alignment in machine translation [6] for more certain alignment, and latent variables for unlabeled alignments in abstract meaning representations[14].

3 Model Description

3.1 Task Definition

Given a post $X = x_1 x_2 \cdots x_n$, the goal is to generate a proper response $Y = y_1 y_2 \cdots y_m$. Besides, there are some relevant knowledge graphs $G = \{g_1, g_2, \cdots, g_{N_G}\}$ retrieved from a commonsense knowledge base. By using the words in the post as queries, we can retrieve them, like Figure 1. They are extra input. Each word in the post corresponds to a graph in G, each graph consists of a set of knowledge triples $g_i = \left\{\tau_1, \tau_2, \cdots, \tau_{N_{g_i}}\right\}$, which surround each word in the post. Each triple (head entity, relation, tail entity) is denoted as $\tau = (h, r, t)$. Finally, the generation probability estimated by the model is: $P(Y|X, G) = \prod_{t=1}^{m} P(y_t|y_{<t}, X, G)$.

3.2 Background: Knowledge Aware Framework

Except the attention on knowledge graphs is different, the framework of conversation generation with commonsense knowledge is similar to [29]. First, we adopt TransE [3] to represent the entities and relations in the knowledge base. Next, we transform TransE embeddings with MLP $\boldsymbol{k} = (\boldsymbol{h}, \boldsymbol{r}, \boldsymbol{t}) = \boldsymbol{MLP}(TransE(h, r, t))$.

Then the retrieved knowledge triples vectors $\boldsymbol{K}(g_i) = \left\{\boldsymbol{k}_1, \boldsymbol{k}_2, \cdots, \boldsymbol{k}_{N_{g_i}}\right\}$ in the graph g_i will produce a knowledge graph vector $\boldsymbol{g_i}$ as follows:

$$\boldsymbol{g_i} = \sum_{n=1}^{N_{g_i}} \alpha_n^s [\boldsymbol{h_n}; \boldsymbol{t_n}] \tag{1}$$

$$\alpha_n^s = \frac{\exp(\beta_n^s)}{\sum_{j=1}^{N_{g_i}} \exp(\beta_j^s)} \tag{2}$$

$$\beta_n^s = (\boldsymbol{W_r r_n})^\top \tanh(\boldsymbol{W_h h_n} + \boldsymbol{W_t t_n}) \tag{3}$$

Where $(\boldsymbol{h_n}, \boldsymbol{r_n}, \boldsymbol{t_n}) = \boldsymbol{k_n}$, $\boldsymbol{W_h}$, $\boldsymbol{W_r}$, $\boldsymbol{W_t}$ are weight matrices. And $\boldsymbol{e}(x_t) = [\boldsymbol{w}(x_t); \boldsymbol{g_i}]$ during encoding. $[;]$ denotes concatenation operation.

Then we use sequence-to-sequence model with $\boldsymbol{GRU}$ [5] to generate the response. The decoder makes full use of the retrieved knowledge graphs.

$$\boldsymbol{s_{t+1}} = \boldsymbol{GRU}(\boldsymbol{s_t}, [\boldsymbol{c_t}; \boldsymbol{c_t^g}; \boldsymbol{c_t^k}; \boldsymbol{e}(y_t)]) \tag{4}$$

$$\boldsymbol{e}(y_t) = [\boldsymbol{w}(y_t); \boldsymbol{k_j}] \tag{5}$$

$\boldsymbol{s_t}$ is decoder state, y_t is the output word, $\boldsymbol{e}(y_t)$ is the concatenation of the word vector $\boldsymbol{w}(y_t)$. $\boldsymbol{k_j}$ is the previous knowledge triple vector, which is from the previous selected word y_t. Then $\boldsymbol{c_t}$ is context vector which is weighted sum of encoder's hidden states with standard attention mechanism at every step [1]. $\boldsymbol{c_t^g}$ and $\boldsymbol{c_t^k}$ are states of incorporated knowledge, which belong to networks named dynamic graph attention and aim at enhancing generation via incorporated knowledge. $\boldsymbol{c_t^g}$ is defined as below:

$$\boldsymbol{c_t^g} = \sum_{i=1}^{N_G} \alpha_{ti}^g \boldsymbol{g_i} \tag{6}$$

$$\alpha_{ti}^g = \frac{\exp(\beta_{ti}^g)}{\sum_{j=1}^{N_G} \exp(\beta_{tj}^g)} \tag{7}$$

$$\beta_{ti}^g = \boldsymbol{V_b}^\top \tanh(\boldsymbol{W_b s_t} + \boldsymbol{U_b g_i}) \tag{8}$$

The graph context vector $\boldsymbol{c_t^g}$ is a weighted sum of the graph vectors $\boldsymbol{g_i}$ base on α_{ti}^g, which is the probability of choosing knowledge graph g_i at step t. $\boldsymbol{V_b}$, $\boldsymbol{W_b}$, $\boldsymbol{U_b}$ are parameters, which measure the association between the decoder's state $\boldsymbol{s_t}$ and a graph vector $\boldsymbol{g_i}$. $\boldsymbol{c_t^k}$ is defined as below:

$$\boldsymbol{c_t^k} = \sum_{i=1}^{N_G} \sum_{j=1}^{N_{g_i}} \alpha_{ti}^g \alpha_{tj}^k \boldsymbol{k_j} \tag{9}$$

$$\alpha_{tj}^k = \frac{\exp(\beta_{tj}^k)}{\sum_{n=1}^{N_{g_i}} \exp(\beta_{tn}^k)} \tag{10}$$

$$\beta_{tj}^k = \boldsymbol{k_j}^\top \boldsymbol{W_c s_t} \tag{11}$$

$\boldsymbol{c_t^k}$ denotes vectors of weighted knowledge triples $\boldsymbol{K}(g_i) = \left\{\boldsymbol{k}_1, \boldsymbol{k}_2, \cdots, \boldsymbol{k}_{N_{g_i}}\right\}$ within each graph g_i by α_{tj}^k and α_{ti}^g. α_{tj}^k is the probability of choosing triple τ_j from all triples in graph g_i at step t. α_{ti}^g is the same as the former in Eq(6) definition, denoting the probability of selecting graph g_i. $\boldsymbol{W_c}$ are parameters, and β_{tj}^k can be viewed as the similarity between each knowledge triple vector $\boldsymbol{k_j}$ (from previous output y_t) and the decoder state $\boldsymbol{s_t}$.

Finally, the knowledge aware generator selects a generic word or an entity word with distributions as follows:

$$\boldsymbol{\alpha_t} = [\boldsymbol{s_t}; \boldsymbol{c_t}; \boldsymbol{c_t^g}; \boldsymbol{c_t^k}] \tag{12}$$

$$\gamma_t = sigmoid(\boldsymbol{V_o}^\top \boldsymbol{\alpha_t}) \tag{13}$$

$$P_c(y_t = w_c) = softmax(\boldsymbol{W_o}\boldsymbol{\alpha_t}) \tag{14}$$

$$P_e(y_t = w_e) = \alpha_{ti}^g \alpha_{tj}^g \tag{15}$$

$$y_t \sim \boldsymbol{o_t} = P(y_t) = \begin{bmatrix} (1-\gamma_t) P_g(y_t = w_c) \\ \gamma_t P_e(y_t = w_e) \end{bmatrix} \tag{16}$$

Where $\tau_j \in [0, 1]$ is a scalar to balance the choice between an entity word w_e and a generic word w_c, P_c/P_e is the distribution over generic/entity words respectively. $\boldsymbol{\alpha_t}$ controls generation of generic words and selection of distribution over generic/entity. $\boldsymbol{V_o}$ and $\boldsymbol{W_o}$ are parameters. The final distribution $P(y_t)$ is a concatenation of two distributions.

3.3 Variational Attention for Knowledge Incorporation

In previous methods like knowledge aware framework above, attention on knowledge graph like α_{ti}^g above is calculated with unfinished generation and partial generated states $\boldsymbol{s_t}$. In this paper, we adopt variational method for computing α_{ti}^g in the knowledge aware framework, which is the attention on knowledge graph g_i. Thus the attention on knowledge triples $\alpha_{ti}^g \alpha_{tj}^k$ will be enhanced at the same time.

Concretely, we introduce posterior distributions $q_\phi(\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}|\boldsymbol{x}, \overline{\boldsymbol{x}}, \boldsymbol{y})$ for attention α_{ti}^g with the complete output information $\boldsymbol{y}$ as condition. Then we can enhance corresponding prior attention distributions $p_\theta(\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}|\boldsymbol{x}, \overline{\boldsymbol{x}})$ with training by KL loss between them. Our variational attention based model is trained by maximizing:

$$\begin{aligned} &L(\theta, \phi; \boldsymbol{x}, \overline{\boldsymbol{x}}, \boldsymbol{y}) = KL(q_\phi(\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}|\boldsymbol{x}, \overline{\boldsymbol{x}}, \boldsymbol{y}))||p_\theta(\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}|\boldsymbol{x}, \overline{\boldsymbol{x}}) \\ &+ E_{q_\phi(\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}|\boldsymbol{x}, \overline{\boldsymbol{x}}, \boldsymbol{y})}[\log p(\boldsymbol{y}|\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}, \boldsymbol{x}, \overline{\boldsymbol{x}})] \end{aligned} \tag{17}$$

$\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{ti}}^{\boldsymbol{g}}}$ is distributions of α_{ti}^g, which is attention on knowledge graph g_i. Note that $\boldsymbol{z}_{\boldsymbol{\alpha}_{\boldsymbol{tj}}^{\boldsymbol{g}}}$ is also calculated but omitted in the equation, because there is no variational method designed for it. They are distributions of α_{tj}^k, which is attention score on knowledge triples vectors $\boldsymbol{K}(g_i)$ in knowledge graph g_i, it's calculated as Eq(10). The first loss is the KL loss between prior distributions and posterior distributions. The second loss is for generation loss of words in common

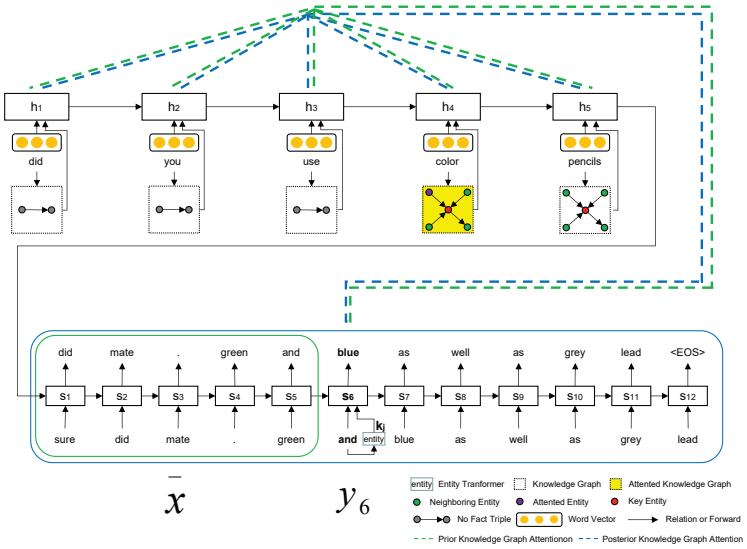

Fig. 2. This figure show the process of our model. Every node is an entity from a knowledge triple in a knowledge graph. Entity tranformer is to produce triple vector $\boldsymbol{k_j}$ base on previously selected output y_t.

sequence-to-sequence models. $\boldsymbol{x}$ is the input post, $\overline{\boldsymbol{x}}$ is attention query denoting current states at every step. Next:

$$p_\theta(\boldsymbol{z}_{\boldsymbol{\alpha}^g_{ti}}|\boldsymbol{x},\overline{\boldsymbol{x}}) = softmax(\boldsymbol{z}_{\boldsymbol{\beta}^g_{ti}})\boldsymbol{z}_{\boldsymbol{\beta}^g_{ti}} \sim \mathcal{N}(\boldsymbol{u},\boldsymbol{\sigma}^2) \tag{18}$$

$$q_\phi(\boldsymbol{z}_{\boldsymbol{\alpha}^g_{ti}}|\boldsymbol{x},\overline{\boldsymbol{x}},\boldsymbol{y}) = softmax(\boldsymbol{z}'_{\boldsymbol{\beta}^g_{ti}})\boldsymbol{z}'_{\boldsymbol{\beta}^g_{ti}} \sim \mathcal{N}'(\boldsymbol{u}',\boldsymbol{\sigma}'^2) \tag{19}$$

$$\begin{bmatrix} \boldsymbol{u}' \\ log(\boldsymbol{\sigma}'^2) \end{bmatrix} = \begin{bmatrix} \boldsymbol{u} \\ log(\boldsymbol{\sigma}^2) \end{bmatrix} + \begin{bmatrix} \boldsymbol{u}'' \\ log(\boldsymbol{\sigma}''^2) \end{bmatrix} \tag{20}$$

$$\begin{bmatrix} \boldsymbol{u}'' \\ log(\boldsymbol{\sigma}''^2) \end{bmatrix} = \tanh(\boldsymbol{W_y} f(\boldsymbol{y}) + \boldsymbol{b_y}) \tag{21}$$

$\boldsymbol{z}_{\boldsymbol{\beta}^g_{ti}}$ and $\boldsymbol{z}_{\boldsymbol{\beta}'^g_{ti}}$ are isotropic Gaussian distributions. $p_\theta(\boldsymbol{z}_{\boldsymbol{\alpha}^g_{ti}}|\boldsymbol{x},\overline{\boldsymbol{x}})/q_\phi(\boldsymbol{z}_{\boldsymbol{\alpha}^g_{ti}}|\boldsymbol{x},\overline{\boldsymbol{x}},\boldsymbol{y})$ is prior/posterior distributions of α^g_{ti}, which is attention weight on knowledge graphs g_i. $\boldsymbol{z}_{\boldsymbol{\beta}^g_{ti}}/\boldsymbol{z}_{\boldsymbol{\beta}'^g_{ti}}$ is prior/posterior distributions of β^g_{ti}, which is attention score on knowledge graph g_i. $\boldsymbol{u}$ is calculated as Eq(8), and $log(\boldsymbol{\sigma}^2)$ is calculated in the same way with new parameters $\boldsymbol{V'_b}$, $\boldsymbol{W'_b}$, $\boldsymbol{U'_b}$ corresponding to $\boldsymbol{V_b}$, $\boldsymbol{W_b}$, $\boldsymbol{U_b}$ respectively. Every element in $\boldsymbol{u}$ or $log(\boldsymbol{\sigma}^2)$ corresponds a β^g_{ti}. $\boldsymbol{W_y}$ and $\boldsymbol{b_y}$ are parameters to involve output $\boldsymbol{Y}$. $f(\boldsymbol{y})$ is the final state of $\boldsymbol{GRU}$ function for $\boldsymbol{Y}$, which contains the information of the complete output. Posterior attention score distributions $\boldsymbol{z}'_{\boldsymbol{\beta}^g_{ti}}$ involve complete output information via an addition operation on distribution.

Assumed attention distributions must be ones that can ensure the sum is one, such as Dirichlet. In our paper, we simplify this process. We fit and optimize the

attention distributions of scores β_{ti}^g rather than direct weight or probability α_{ti}^g. The attention scores will be transformed into probability from $0-1$, and ensure the sum is one after feeded into $softmax$ function. We use the reparametrization trick [10] to obtain samples of scores distributions.

Concretely, based on an auxiliary noise variable $\boldsymbol{\epsilon} \sim \mathcal{N}_p(0,1)$ of prior scores distributions and the other $\boldsymbol{\epsilon'} \sim \mathcal{N}_q(0,1)$ of posterior scores distributions, we can sample as follows:

$$\boldsymbol{z}_{\boldsymbol{\beta}_{\boldsymbol{ti}}^{\boldsymbol{g}}} = \boldsymbol{u} + \exp(\log(\boldsymbol{\sigma}^2)) \circ \boldsymbol{\epsilon} \tag{22}$$

$$\boldsymbol{z'}_{\boldsymbol{\beta}_{\boldsymbol{ti}}^{\boldsymbol{g}}} = \boldsymbol{u'} + \exp(\log(\boldsymbol{\sigma'}^2)) \circ \boldsymbol{\epsilon'} \tag{23}$$

$\circ$ is element-wise product. To avoid randomness, we only use prior distributions of $\boldsymbol{u}$ without any output information as condition during test. In fact, we find that the final $\exp(\log(\boldsymbol{\sigma}^2))$ is very small after trained in the experiments.

In this way, we incorporate output information to train α_{ti}^g, which is the attention of knowledge graph $\boldsymbol{g_i}$. Thus we improve the ability of capturing clean and suitable attention on knowledge graphs. The process is show as Figure 2. We call the variational attention based commonsense knowledge aware conversational model VACCM.

4 Experiments

4.1 Data

Our dataset is the same as [29]. For commonsense knowledge base, we use ConceptNet[3] [23] as the commonsense knowledge base. Conversation dataset is from the site[4]. There are four different sets: high-frequency pairs where each post has all top 25% frequent words, medium-frequency pairs within the range of 25%-75%, low-frequency pairs within the range of 75%-100%, and OOV pairs where each post contains out-of-vocabulary words. There are 5,000 pairs randomly sampled from the dataset in each test set. Besides, there are around 5.8 graphs per pair, 106.4 entities per pair and 18.3 triples per graph.

4.2 Settings

The two encoders for the posts and output words have 2-layer GRU structures with 512 hidden cells for each layer. The decoder has the same settings. Cells don't share parameters. The size of word embedding is set to 300 and the size of vocabulary is set to 30,000. The embedding size of entities and relations is set to 100, and we adopted TransE [3] to obtain entity and relation representations. The mini-batch size is set to 100. The weight of KL loss increases linearly from 0 to 1 in the first 5000 batches. We used the Adam optimizer to tain, and the learning rate is set to 0.0001. We ran the models at most 20 epoches.

[3] https://conceptnet.io

[4] https://www.reddit.com/r/datasets/comments/3bxlg7/i_have_every_publicly_available_reddit_comment/

	Overall	High Freq	Medium Freq	Low Freq	OOV
app.	0.554	0.556	0.520	0.534	0.605
inf.	0.473	0.462	0.477	0.492	0.459

Table 1. Manual evaluation results. The metrics are *appropriateness* (app.) and *informativeness* (inf.) respectively.

	Overall		High Freq		Medium Freq		Low Freq		OOV	
	ppx.	ent.	ppx.	ent.	ppx.	ent.	ppx.	ent.	ppx.	ent.
Seq2Seq	47.02	0.717	42.41	0.713	47.25	0.740	48.61	0.721	49.96	0.669
MemNet	46.85	0.761	41.93	0.764	47.32	0.788	48.86	0.760	49.52	0.706
CopyNet	40.27	0.96	36.26	0.91	40.99	0.97	42.09	0.96	42.24	0.96
CCM	39.18	**1.180**	35.36	**1.156**	39.64	**1.191**	40.67	**1.196**	40.87	**1.162**
VACCM	**38.49**	1.158	**34.74**	1.141	**38.90**	1.163	**40.36**	1.179	**40.25**	1.149

Table 2. Automatic evaluation results. The metrics are *perplexity* (ppx.) and *entity score* (ent.) respectively.

4.3 Baselines

Other models as baselines are as follows:

– Seq2Seq. A seq2seq model [24], which is commonly used in open-domain conversational systems.
– MemNet. A knowledge-grounded adapted from [7], where the memory units use the TransE [3] embeddings of knowledge triples.
– CopyNet. A copy network model [30], which can copy a word from knowledge triples besides generating a word from the vocabulary.
– CCM. A commonsense knowledge aware conversational model with graph attention [29]. The difference from our model is that we use variational attention on knowledge graphs.

4.4 Automatic Evaluation

Metrics:
There are two metrics for automatic evaluation. The first is *perplexity* (ppx.) [19]. It is adopted to evaluate the model at the content level, which is the same with [29]. So we can know whether the content is grammatical and relevant in topic by *perplexity*; The *entity score* (ent.) is another metric, which calculates the number of entities per response. Its aim is to measure the model's ability to select the concepts from the commonsense knowledge base in generation.
Results:
The results are shown in Table 2. VACCM has the lowest *perplexity* over all the test sets. It indicates that VACCM can better understand the posts of users and generate more grammatical responses. For *entity score*, VACCM is obviously higher than Seq2Seq, MemNet and CopyNet. But it's slightly lower than CCM, it indicates that variational attention incorporate more clean and suitable knowledge. We can also know commonsense knowledge is more used in low-frequency posts than high-frequency posts. It can be explained that rare

	Overall		High Freq		Medium Freq		Low Freq		OOV	
	aut.	man.	aut.	man.	aut.	man.	aut.	man.	aut.	man.
VACCM vs. CCM	+3.8%	0.552	+8.3%	0.583	0.3%	0.529	+3.0%	0.550	+3.7%	0.548

Table 3. Evaluation results on metric *precision*. The results are in automatic evaluation (aut.) and manual evaluation (man.) respectively.

Post	US vs Algeria 2010 world cup Donovan's goal. Amazing!	I think theon will live . The old gods aren't done with him after all.	Random question, how long of a drive is it to Chicago from Detroit?
Knowledge	(Algeria, IsA, country), (world, AtLocation, thought), (play, RelatedTo, goal)	(home, RelatedTo, live), (die, RelatedTo, live), (gods, FormOf, god)	(hour, RelatedTo, long), (road, RelatedTo, drive), (Detroit, PartOf, Michigan)
CCM	I thought that was the goal.	I think he's going to be a god.	I'm in Michigan.
VACCM	I was so excited to see him play in the second half.	I think he's going to die.	I think it's a 5 hour drive from Detroit.

Table 4. Generation samples between VACCM (variational attention) and CCM (standard attention). Colored words are also entities in knowledge base.

concepts need more shared background to understand and reply. The *perplexity* for high-frequency posts is still lower than low-frequency posts, it's because that the frequent words can be more sufficiently trained.

4.5 Manual Evaluation

Metrics:
The metrics of manual evaluation is the same with [29]. There are two metrics for manual evaluation: One is *appropriateness* (app.), which aims at the content level. It tests the response whether appropriate or not in grammar, topic, and logic; and the other is $informativeness$ (inf.), which aims at the knowledge level. It tests the response whether can provide new information and knowledge connected with the post).
Statistics:
Considering the high cost of manual annotation and focusing on the effects of variational attention, we only compared our model with the state-of-the-art model CCM [29]. For manual annotation, there are 50 posts randomly sampled from different frequency based test sets. In total, we have 400 pairs since we have four test sets and two metrics. A pair-wise comparison is conducted between the responses generated by VACCM and CCM for the same post. For each response pair, three judges were hired to give a preference between the two responses, in terms of the above two metrics. The Kappa of annotation consistency is 0.41 and 0.61 for *appropriateness* and $informativeness$ respectively. "Tie" was also allowed.
Results:
The results are shown in Table 1. The score is the percentage that VACCM wins the state-of-the-art method CCM after removing "Tie" pairs. It shows that VACCM outperforms CCM in metrics *appropriateness*. We can know variational attention based model can incorporate knowledge into generation more

properly, and thus generate more appropriate responses. Specially, we can see the improvement is more obvious on the OOV part. Maybe it's because variational attention have better ability to utilize commonsense knowledge to understand out of vocabulary words. VACCM is lower than CCM in $informativeness$. This indicates that VACCM may reduce noisy and meaningless entities when incorporating knowledge into generation, especially in the situations that need more shared background to understand rare concepts.

4.6 Extra Accurate Incorporation Evaluation

Metrics:
The *entity score* shows how much knowledge can be recalled in responses, but can't evaluate the precision of incorporation. In fact, models may incorporate unsuitable knowledge. In addition, some knowledge is just about different forms of words. These forms do not contain real semantic knowledge, such as (gods, FormOf, god) in Table 4. Thus we propose another metric *precision* in both automatic evaluation (aut.) and manual evaluation (man.), and then compare our model with CCM. The manual annotation is to decide whether the incorporated knowledge in generated response is needful and suitable.
Statistics:
In automatic evaluation, we calculate the number of matched entities between predicted response and golden response. In manual evaluation, there are also 50 posts randomly sampled from different frequency based test sets. In total, we have 200 pairs since we have four test sets and one metric. A pair-wise comparison is conducted between the responses generated by VACCM and CCM for the same post. We also hired three judges to give a preference between the two responses. The Kappa of annotation consistency is 0.53. "Tie" was also allowed.
Results:
The results are shown in Table 3. The score *precision* in automatic evaluation indicates how much VACCM wins CCM on the number of matched entities between predicted response and golden response. The score *precision* in manual evaluation is the percentage that VACCM wins CCM after removing "Tie" pairs. In above experiments, we can know VACCM tends to incorporate less but more clean and suitable knowledge than CCM. Especially for high-frequency posts, the improvement is most obvious.

4.7 Study Case

Post-Response Pairs:
Some samples of generation are shown in Table 4. They prove VACCM can do better in incorporating clean and suitable knowledge. Like the third example, the cared information in the post is the cost of time. CCM focuses on the entity word "Detroit" in the post, and thus generate "I'm in Michigan." from relevant knowledge (Detroit, PartOf, Michigan). The response is grammatical with knowledge but not suitable. Contrarily, the VACCM focuses on the entity word "long" and generate corresponding entity word "hour" from relevant knowledge

(hour, RelatedTo, long), which is more suitable. In addition, some incorporated knowledge may be meaningless such as form transformation (gods, FormOf, god) in the second example.

5 Conclusion

In this paper, we present a model (VACCM) for commonsense knowledge aware conversational generation, which uses variational attention on knowledge graphs. Automatic and manual evaluation as well as sampled examples show that VACCM can model better attention on knowledge graphs and generate appropriate responses with more clean and suitable knowledge.

6 Acknowledgement

This work is supported by the National Natural Science Foundation of China (No.61533018), the Natural Key R&D Program of China (No.2018YFC0830101), the National Natural Science Foundation of China (No.61702512, No.61806201) and the independent research project of National Laboratory of Pattern Recognition. This work was also supported by CCF-DiDi BigData Joint Lab and CCF-Tencent Open Research Fund.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
2. Bahuleyan, H., Mou, L., Vechtomova, O., Poupart, P.: Variational attention for sequence-to-sequence models. arXiv preprint arXiv:1712.08207 (2017)
3. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Advances in neural information processing systems. pp. 2787–2795 (2013)
4. Bowman, S.R., Vilnis, L., Vinyals, O., Dai, A.M., Jozefowicz, R., Bengio, S.: Generating sentences from a continuous space. arXiv preprint arXiv:1511.06349 (2015)
5. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
6. Deng, Y., Kim, Y., Chiu, J., Guo, D., Rush, A.M.: Latent alignment and variational attention. arXiv preprint arXiv:1807.03756 (2018)
7. Ghazvininejad, M., Brockett, C., Chang, M.W., Dolan, B., Gao, J., Yih, W.t., Galley, M.: A knowledge-grounded neural conversation model. arXiv preprint arXiv:1702.01932 (2017)
8. Gu, J., Lu, Z., Li, H., Li, V.O.: Incorporating copying mechanism in sequence-to-sequence learning. arXiv preprint arXiv:1603.06393 (2016)
9. He, S., Liu, C., Liu, K., Zhao, J.: Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. In: ACL. vol. 1, pp. 199–208 (2017)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2013)

11. Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. In: NAACL. pp. 110–119 (2016)
12. Li, J., Monroe, W., Ritter, A., Jurafsky, D., Galley, M., Gao, J.: Deep reinforcement learning for dialogue generation. In: EMNLP. pp. 1192–1202 (2016)
13. Long, Y., Wang, J., Xu, Z., Wang, Z., Wang, B., Wang, Z.: A knowledge enhanced generative conversational service agent. In: Proceedings of the 6th Dialog System Technology Challenges (DSTC6) Workshop (2017)
14. Lyu, C., Titov, I.: Amr parsing as graph prediction with latent alignment. In: ACL. pp. 397–407 (2018)
15. Marková, I., Linell, P., Grossen, M., Salazar Orvig, A.: Dialogue in focus groups: Exploring socially shared knowledge. Equinox publishing (2007)
16. Minsky, M.: Society of mind : A response to four reviews. Artificial Intelligence **48**(3), 371–396 (1991)
17. Mou, L., Song, Y., Yan, R., Li, G., Zhang, L., Jin, Z.: Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. arXiv preprint arXiv:1607.00970 (2016)
18. do Nascimento Souto, P.C.: Creating knowledge with and from the differences: the required dialogicality and dialogical competences. RAI-Revista de Administração e Inovação **12**(2), 60–89 (2015)
19. Serban, I.V., Sordoni, A., Bengio, Y., Courville, A.C., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: AAAI. vol. 16, pp. 3776–3784 (2016)
20. Serban, I.V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A.C., Bengio, Y.: A hierarchical latent variable encoder-decoder model for generating dialogues. In: AAAI. pp. 3295–3301 (2017)
21. Shang, L., Lu, Z., Li, H.: Neural responding machine for short-text conversation. arXiv preprint arXiv:1503.02364 (2015)
22. Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.Y., Gao, J., Dolan, B.: A neural network approach to context-sensitive generation of conversational responses. In: NAACL. pp. 196–205 (2015)
23. Speer, R., Havasi, C.: Representing general relational knowledge in conceptnet 5. In: LREC. pp. 3679–3686 (2012)
24. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks. In: Advances in neural information processing systems. pp. 3104–3112 (2014)
25. Xing, C., Wu, W., Wu, Y., Liu, J., Huang, Y., Zhou, M., Ma, W.Y.: Topic aware neural response generation. In: AAAI. vol. 17, pp. 3351–3357 (2017)
26. Xu, Z., Liu, B., Wang, B., Sun, C., Wang, X.: Incorporating loose-structured knowledge into conversation modeling via recall-gate lstm. In: Neural Networks (IJCNN), 2017 International Joint Conference on. pp. 3506–3513. IEEE (2017)
27. Zhang, R., Guo, J., Fan, Y., Lan, Y., Xu, J., Cheng, X.: Learning to control the specificity in neural response generation. In: ACL. pp. 1108–1117 (2018)
28. Zhao, T., Zhao, R., Eskenazi, M.: Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In: ACL. pp. 654–664 (2017)
29. Zhou, H., Young, T., Huang, M., Zhao, H., Xu, J., Zhu, X.: Commonsense knowledge aware conversation generation with graph attention. In: IJCAI. pp. 4623–4629 (2018)
30. Zhu, W., Mo, K., Zhang, Y., Zhu, Z., Peng, X., Yang, Q.: Flexible end-to-end dialogue system for knowledge grounded conversation. arXiv preprint arXiv:1709.04264 (2017)