S2SPMN:A Simple and Effective Framework for Response Generation with Relevant Information

Jiaxin Pei
School of Computer Science
Wuhan University
Hubei, China
pedropei@vip.qq.com

Chenliang Li*
School of Cyber Science and Engineering
Wuhan University
Hubei, China
cllee@whu.edu.cn

Abstract

How to generate relevant and informative responses is one of the core topics in response generation area. Following the task formulation of machine translation, previous works mainly consider response generation task as a mapping from a source sentence to a target sentence. To realize this mapping, existing works tend to design intuitive but complex models. However, the relevant information existed in large dialogue corpus is mainly overlooked. In this paper, we propose Sequence to Sequence with Prototype Memory Network (S2SPMN) to exploit the relevant information provided by the large dialogue corpus to enhance response generation. Specifically, we devise two simple approaches in S2SPMN to select the relevant information (named prototypes) from the dialogue corpus. These prototypes are then saved into prototype memory network (PMN). Furthermore, a hierarchical attention mechanism is devised to extract the semantic information from the PMN to assist the response generation process. Empirical studies indicate the advantage of our model over several classical and strong baselines.

1 Introduction

Dialogue systems, or say, chatbots are usually considered as the future of human-computer interaction and extensive works have been done in this area (Wen et al., 2016; Qiu et al., 2017; Wen et al., 2017; Kreyssig et al., 2018).

As one of the main approaches for dialogue system design, response generation has attracted more and more attention from research community. Neural networks based models like Seq2Seq architecture (Vinyals and Le, 2015; Shang et al., 2015) are proven to be effective to generate valid responses for a dialogue system. However, as revealed in many previous works (Li et al., 2016a; Wu et al., 2018), "safe reply" is still an open problem and lots of efforts are made to generate more informative responses (Li et al., 2016a; Mou et al., 2016; Li et al., 2016b; Qiu et al., 2017; Li et al., 2017; Zhao et al.; He et al., 2017; Zhou et al., 2017; Liu et al., 2018; Chen et al., 2018).

Note that in this paper when we say response generation, we focus on single turn chit-chat for that other tasks like multi-turn (Zhang et al., 2018) or goal-oriented (Kan et al., 2018)generation could be partly considered as the extensions of single-turn generation.

Though existing works mentioned above are helpful in some ways, they all follow the task formulation proposed by (Ritter et al., 2011), which considers response generation (RG) task as a mapping from a source sentence to a target sentence like machine translation (MT). This task formulation ignores the natural difference between MT and RG: MT deals with sentence pairs of the same meanings while RG needs to realize the meaning transformation from a source post to the target response. In this sense, the meaning transformation is more difficult than machine translation. Hence, many researchers have designed more and more complex models. However, given a target post, the relevant information covered by the dialogue corpus is usually overlooked. It is intuitive that the responses for a similar post would provide more contextual information to guide the response generation. To this end, we are interested in exploiting the relevant responses in the training set as soft prototypes to assist the response generation.

Specifically, in this paper, we propose Sequence to Sequence with Prototype Memory Network (named S2SPMN). We introduce two Prototype Memory Networks (PMNs) to store the relevant responses extracted from the dialogue corpus: static PMN and dynamic PMN. Tested on a widely used benchmark dataset, the proposed

*Chenliang Li is the Corresponding Author

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 745–750
Brussels, Belgium, October 31 - November 4, 2018.

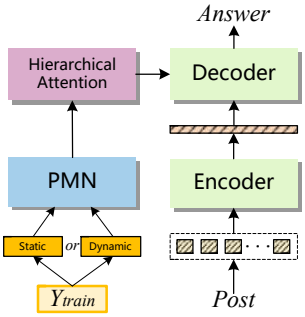

Figure 1: S2SPMN Framework

S2SPMN produces more informative responses than the standard and strong baselines. To the best of our knowledge, it is the first work leveraging prototype information in dialogue corpus in response generation area.

The contributions of this paper could be summarized as follows:

(1) We propose S2SPMN, a simple yet effective response generation model which could leverage relevant information in dialogue corpus to assist response generation.

(2) Empirical studies indicate the superiority of proposed S2SPMN over other methods.

2 Architecture

2.1 Problem Definition

Given a dialogue dataset $\Gamma = \{X_i, Y_i\}_{i=1}^{N}$, where Y_i is the response for a post X_i, we aim to train a model with Γ such that the model can generate an accurate and informative response for a new post X'. Here, we propose to exploit the relevant information provided by Γ. Let $T' = (r_1, r_2, ..., r_m)$ refers to the prototype memory network constructed for post X', where r_i is the i-th relevant response (named prototype) extracted from dialogue dataset Γ. The goal is to derive the model to generate the response Y': $p(Y'|X') = p(Y'|T', X')$.

In following sections, we firstly introduce the generation framework with hierarchical attention mechanism assuming PMN is constructed. Then we will introduce two kinds of PMNs: static PMN and dynamic PMN.

2.2 Sequence-to-Sequence with Prototype Memory Network

S2SPMN is built with a Seq2Seq encoder-decoder framework (Sutskever et al., 2014) with the attention mechanism (Bahdanau et al., 2014). We use LSTM (Hochreiter and Schmidhuber, 1997) to materialize both encoder and decoder. The hidden state at t-th encoding step is generated from previous hidden state h_{t-1} and current input x_t as follows:

$$h_t = lstm(x_t, h_{t-1}) \tag{1}$$

For decoder, at i-th timestep, s_i is the decoder's hidden state and p_i is the probability distribution of candidate words .

$$s_i = lstm(y_{i-1}, s_{i-1}, c_i, o_i) \tag{2}$$
$$p_i = softmax(MLP(s_i, y_{t-1}, o_i, c_i)) \tag{3}$$

where $MLP()$ is a one-layer perception, o_i is the hierarchical attention over entire prototype memory network which will be formalized in following sections. c_i is the summarization for the post regarding to the hidden state s_{i-1}:

$$c_i = \sum_{j=1}^{T} \alpha_{ij} h_j, \quad \alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T} exp(e_{ik})} \tag{4}$$
$$e_{ij} = v_1^T MLP(s_{i-1}, h_j) \tag{5}$$

where v_1 is the attention parameter.

2.3 Prototype Memory Network

Given a post X', a set of responses are selected from training set as prototypes and are then saved into the Prototype Memory Network(PMN). We propose two kinds of Prototype Memory Networks.

Static PMN: For static PMN(SPMN), we randomly select m responses before training starts and the entire PMN remains unchanged during the training process. That is, we use the same prototypes for all the post-response pairs.

Dynamic PMN: In dynamic PMN(DPMN), prototypes are selected by retrieving the most relevant posts. We calculate the cosine similarity with TF-IDF weighting scheme between the given post and all the posts in training set. We consider top-m posts and put the associated responses into DPMN. This means that the prototypes are characteristic for each post-response pair.

In both SPMN and DPMN, m is a predefined hyper-parameter controlling the size of the PMN. Each prototype is represented with the concatenation of word embeddings. We perform zero padding for both SPMN and DPMN with a pseudo

word[1], making the length for the representation of each prototype be the same. Here we denote the prototype memory network as PMN = $\{r_1, r_2, ..., r_m\}$, in which r_m is the representation of m-th prototype and m is the size of the PMN. And $r_m = \{w_{m,1}, w_{m,2}, ..., w_{m,l}\}$ where $w_{m,i}$ is the embedding of i-th word, and l is the maximum allowable length for a prototype.

For both SPMN and DPMN, we select responses rather than posts although sometimes they have similar vocabularies and syntactic structure. We believe that using responses as prototypes could help with the meaning transformation from post to response. In DPMN, all the retrieved prototypes could be considered as responses to the target post. It is intuitive that the generated response would have similar representation to these prototypes.

2.4 Hierarchical Attention Mechanism

We use a two-stage hierarchical attention mechanism to extract useful information in PMN and integrate it into the decoding process. The first stage is a sentence level attention over entire PMN to generate the abstractive prototype $\hat{r}_i$ at each timestep:

$$\hat{r}_i = \sum_{j=1}^{M} \beta_{ij} r_j, \quad \beta_{ij} = \frac{exp(f_{ij})}{\sum_{k=1}^{M} exp(f_{ik})} \tag{6}$$

$$f_{ij} = v_2^T MLP(s_{i-1}, r_j) \tag{7}$$

where v_2 is the attention parameter.

The second stage is a word level attention o_i over the generated $\hat{r}_i = \{\hat{w}_1, \hat{w}_2, ..., \hat{w}_l\}$ and is calculated as follows:

$$o_i = \sum_{j=1}^{l} \gamma_{ij} \hat{w}_j, \quad \gamma_{ij} = \frac{exp(g_{ij})}{\sum_{k=1}^{l} exp(g_{ik})} \tag{8}$$

$$g_{ij} = v_3^T MLP(s_{i-1}, \hat{w}_j) \tag{9}$$

where v_3 is the attention parameter.

3 Experiment

3.1 Experiment Setup

We use a subset of STC dataset (Shang et al., 2015) crawled from Weibo, the largest social media in China. The vocabulary size is set to be $8,000$ for computational efficiency and words out of vocabulary are replaced by the symbol "unk". We remove sentences longer than 25 words or containing more than 2 unk symbols. After preprocessing step, we have $315,980$ post-response pairs in training set, $3,510$ pairs in validation set and 300 in test set.

In our model, we use one-layer LSTM and the hidden size is set to be 600 in both encoder and decoder. For all the words used in our model, the embedding size is 300. Mini-batch learning is used and batch size is set as 64. We use simple SGD for optimization and the initial learning rate is set to be 0.2.

3.2 Evaluation Metrics

We use two automatic evaluation metrics including Perplexity and Distinct. Human evaluation is also conducted as the only gold standard for response generation is human judgement.

Perplexity: Following (Vinyals and Le, 2015) and (Xing et al., 2017), we use perplexity as one of our automatic evaluation metrics. Perplexity could measure the holistic condition of model learning. A lower perplexity score indicates better generalization performance. Perplexity on both validation set (PPL-V) and test set (PPL-T) are presented in table 2.

Distinct-1, Distinct-2: Distinct-1 and distinct-2 calculate the ratios of distinct unigrams and bigrams in the generated responses respectively (Li et al., 2016a; Xing et al., 2017; Wu et al., 2018). The higher score suggests that the generated response is more diverse and informative. Here, we report the distinct-1 and distinct-2 scores on entire test set.

Human Annatation: We further recruit human annotators to judge the quality of the generated answers for all the qa-pairs in test set. Responses generated by all the methods are pooled and randomly shuffled for each annotator. A score between 0 and 2 is assigned to each generated answer based on the following criteria:

+2: the answer is natural and relevant to the question.

+1: the answer can be used as a reply, but is not informative enough (e.g. "我也是" (me too), "不知道" (I don't know)).

+0: the answer is irrelevant and unclear in meaning (e.g. too many grammatical errors to understand).

[1]The embedding of the pseudo word is a zero vector.

Model	PPL-V	PPL-T	distinct-1	distinct-2
S2SA	8.41	9.05	**0.0809**	**0.2110**
TAS2S	7.38	7.84	0.04759	0.1087
SPMN500	7.04	7.93	0.06430	0.1734
SPMN1000	**6.28**	7.72	0.07347	0.1909
DPMN100	6.45	**7.69**	0.04350	0.1048

Table 1: Automatic evaluation

Model	0	1	2	Kappa
S2SA	**76.83**%	16.33%	6.83%	0.6124
TAS2S	69.83%	19.83%	10.33%	0.7425
SPMN500	21.67%	55.00%	23.33%	0.6534
SPMN1000	19.17%	52.50%	28.33%	0.7330
DPMN100	12.08%	**56.67**%	**31.25**%	0.6280

Table 2: Human Annotation

3.3 Results Comparation

We use a standard baseline and a strong baseline for comparison.

S2SA: The standard Seq2Seq model with an attention mechanism (Vinyals and Le, 2015).

TAS2S: One of the existing state-of-the-art neural models based on Seq2Seq architecture. The topical words relevant to the post are considered via an attention mechanism when decoding (Xing et al., 2017).

As for our models, we use SPMN to denote the generating method with static prototype memory networks and DPMN with dynamic prototype memory networks. The numbers following model names are the size of PMN.

Automatic Evaluation: Table 1 shows the automatic evaluation results. We see that both SPMN and DPMN obtain huge improvements over the two baselines in terms of PPL-V and PPL-T. Also, we observe that SPMN1000 outperforms SPMN500 in all the four automatic metrics. Note that each post has the same prototypes provided by SPMN. This is reasonable that the relevant response is more likely to be covered by storing more prototypes in SPMN. As for the DPMN, we can see that DPMN achieves the best performance with only 100 prototypes in terms of PPL-T, compared with the other 4 methods. This suggests that using a retrieval mechanism to incorporate the relevant responses brings more useful information for better response generation. Note that S2SA outperforms the others in terms of distinct-1 and distinct-2. Further human evaluation indicates that many responses generated by S2SA are irrelevant and meaningless, which could inevitably increase the distinct scores.

Human Annotation: Table 2 shows human annotation results. It is clear that our models (SPMN500, SPMN1000, DPMN100) generate much more informative and valid responses and much less meaningless or "safe" responses than baseline models (S2SA, TAS2S). Specifically, SPMN500, SPMN1000 and DPMN100 all outperform S2SA and TAS2S by producing more informative and valid responses. Also, we can find that DPMN still outperforms SPMN500 and SPMN1000 with only 100 relevant responses, which is consistent with the observation made in automatic evaluation (in terms of PPL-V and PPL-T).

Case Study Table 3 shows several cases generated by different models. Note that the size of training set and vocabulary used in our experiments are relatively small compared to millions of qa-pairs used in other works (Xing et al., 2017; Wu et al., 2018), so it's reasonable that bad cases sometimes occur in results of baselines. However, our models, no matter the static one or the dynamic one, could generate amazing responses which are not only grammatical and informative, but also have some emotional expressions like the use of punctuation and repetition.

4 Related Work

4.1 Natural language generation

How to generate grammatical and interesting sentences in different situations is one of the core topics in natural language processing area. Extensive works are proposed to generate poems (Zhang et al., 2017), abstracts (Wang and Ling, 2016), arguments (Hua and Wang, 2018), stories (Peng et al., 2018) and so on. Although existing approaches are useful in some ways, it's still difficult to generate natural sentences from scratch and integrating retrieved results has recently become a new fashion in this area. Hua and Wang (2018) proposed an encoder-decoder style neural network-based argument generation model enriched with externally retrieved evidence from Wikipedia. Li et al. (2018) devised a Retrieve-Rerank-Rewrite model for abstractive summarization which uses retrieved results as soft template to assist the decoding process.

Post 1	**美国加州莫诺湖，奇异的美丽 (Mono Lake in California,the US, fantastically beautiful)**
S2SA	有unk的地方,就是unk (It's unk if there's unk)
TAS2S	，，， (speechless)
SPMN500	这是什么地方？我也去看看 (Where's the place? I'd like to go and see)
SPMN1000	这是在哪？！ (What's the place?!)
DPMN100	我也想去的地方 (That's exactly where I want to go)
Post 2	**看看能让妈妈疯掉的baby (Oh! Look at the baby! She's driving mom mad!)**
S2SA	unk,unk! (speechless)
TAS2S	我也喜欢 (Wow,I like her,too)
SPMN500	好可爱啊！好可爱！ (She's so cute!)
SPMN1000	我也是这样的 (I was like her when I was at her age)
DPMN100	我也想养一个 (I want a baby like her)
Post 3	**装修以后很快后悔的80件事！还没装修的朋友们，一定要借鉴！ (80 things to regret after decorating your house! Look at this article if you haven't started decoration!)**
S2SA	有unk的时候，我会有一个unk的unk! (When I have unk, I will have a unk unk!)
TAS2S	，的 (speechless)
SPMN500	我也要去看看，一定要收藏！ (Ok,I will read and collect it!)
SPMN1000	很实用，很实用，很实用 (very very very useful)
DPMN100	很实用，很实用，很实用，很实用！ (very very very very useful!)

Table 3: The answers generated by different models for the sampled questions.

4.2 Response generation

Hand-craft rules, retrieval and generation are three main solutions for conversational AI and generation is the most interesting one in current research community. Li et al. (2016a; 2016b; 2017) proposed a series of works in solving the "safe reply" problem using different approaches like redefining the objective function or leveraging GAN. Xing et al. (2017) considered topic coherence issue by incorporating topical words. Dynamically restricting the target vocabulary is also an interesting idea and Wu et al. (2018) proposed to filter irrelevant words while achieving better computational efficiency . He et al. (2017) introduced copy mechanism to simulate people's behaviors in real conversations and the proposed model could copy useful words from source sentences. Zhou et al. (2017) indicated that emotion is quite important in real dialogues thus an emotional chatting machine was devised to generate emotional responses. Liu et al. (2018) proposed a neural knowledge diffusion (NKD) model to introduce knowledge into dialogue generation.

5 Conclusion and Future Work

In this paper, we propose S2SPMN, a simple yet effective response generation model by exploiting relevant information contained in large dialogue dataset. Empirical studies indicate that simply selecting responses from training set as prototypes and integrating them into the generation process could dramatically improve the quality of generated responses. Moreover, our model is very flexible and could be adapted to any other Seq2Seq based generation methods. Most importantly, we claim the intrinsic difference between RG and MT and propose a new way to define response generation.

As the first work trying to help with the meaning transformation between source and target, we have obtained the encouraging progress. However, we know that there are still many directions to enrich the proposed framework. In future work, we would like to devise more sophisticated solutions to bridge the semantic gap in RG and explore linguistic patterns in conversations like what has been done in discourse analysis (Lei et al., 2018) .

Acknowledgments

This research was supported by National Natural Science Foundation of China (No. 61502344, No. 61872278), Natural Scientific Research Program of Wuhan University (No. 2042017kf0225). Chenliang Li is the corresponding author.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Hongshen Chen, Zhaochun Ren, Jiliang Tang, Yihong Eric Zhao, and Dawei Yin. 2018. Hierarchical variational memory network for dialogue generation. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web*, pages 1653–1662.

Shizhu He, Cao Liu, Kang Liu, and Jun Zhao. 2017. Generating natural answers by incorporating copying and retrieving mechanisms in sequence-to-sequence learning. In *ACL*, pages 199–208.

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

Xinyu Hua and Lu Wang. 2018. Neural argument generation augmented with externally retrieved evidence. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Min-Yen Kan, Xiangnan He, Wenqiang Lei, Xisen Jin, Zhaochun Ren, and Dawei Yin. 2018. Sequicity: Simplifying task-oriented dialogue systems with single sequence-to-sequence architectures. In *ACL*.

Florian Kreyssig, Iñigo Casanueva, Pawel Budzianowski, and Milica Gasic. 2018. Neural user simulation for corpus-based policy optimisation for spoken dialogue systems. In *SIGDIAL Conference*.

Wenqiang Lei, Yuanxin Xiang, Yuwei Wang, Qian Zhong, Meichun Liu, and Min-Yen Kan. 2018. Linguistic properties matter for implicit discourse relation recognition: Combining semantic interaction, topic continuity and attribution. In *AAAI*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *NAACL*, pages 110–119.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016b. Deep reinforcement learning for dialogue generation. In *EMNLP*, pages 1192–1202.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial learning for neural dialogue generation. In *EMNLP*, pages 2157–2169.

Wenjie Li, Furu Wei, Sujian Li, and Ziqiang Cao. 2018. Retrieve, rerank and rewrite: Soft template based neural summarization. In *ACL*.

Qun Liu, Yang Feng, Hongshen Chen, Zhaochun Ren, Dawei Yin, and Shuman Liu. 2018. Knowledge diffusion for neural dialogue generation. In *ACL*.

Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In *COLING*, pages 3349–3358.

Nanyun Peng, Marjan Ghazvininejad, Jonathan May, and Kevin Knight. 2018. Towards controllable story generation. In *Proceedings of the First Workshop on Storytelling*, pages 43–49.

Minghui Qiu, Feng-Lin Li, Siyu Wang, Xing Gao, Yan Chen, Weipeng Zhao, Haiqing Chen, Jun Huang, and Wei Chu. 2017. Alime chat: A sequence to sequence and rerank based chatbot engine. In *ACL*, pages 498–503.

Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *EMNLP*, pages 583–593.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *ACL*, pages 1577–1586.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112.

Oriol Vinyals and Quoc V. Le. 2015. A neural conversational model. *CoRR*, abs/1506.05869.

Lu Wang and Wang Ling. 2016. Neural network-based abstract generation for opinions and arguments. In *HLT-NAACL*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Pei hao Su, Stefan Ultes, David Vandyke, and Steve J. Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *EACL*.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Pei hao Su, David Vandyke, and Steve J. Young. 2016. Multi-domain neural network language generation for spoken dialogue systems. In *HLT-NAACL*.

Yu Wu, Wei Wu, Dejian Yang, Can Xu, Zhoujun Li, and Ming Zhou. 2018. Neural response generation with dynamic vocabularies. In *AAAI*.

Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic aware neural response generation. In *AAAI*, pages 3351–3357.

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 *ACL*.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling multi-turn conversation with deep utterance aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING 2018)*, pages 3740–3752.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*, pages 654–664.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2017. Emotional chatting machine: emotional conversation generation with internal and external memory. *arXiv preprint arXiv:1704.01074*.