A Focus of Attention Neural Conversation Model

Chuwei Luo
Wuhan University,
Department of Computing,
The Hong Kong Polytechnic University,
luochuwei@whu.edu.cn

Abstract

When we use sequence-to-sequence neural network-based models to generate conversational responses, models may perform not very well on a long response. We believe that the traditional way of decoding a sentence from the first word to last word in sequence-to-sequence models is not suitable for producing a response sentence. We propose ABCDEFG Neural Responding Model, a novel sequence-to-sequence neural network-based model for response generation. Given a post sentence as an input, our model encodes it by a recurrent neural network with an attention mechanism. Then, our model decodes the response by firstly producing the root-word of the response instead of generating the first word of the response. Experiments demonstrate the effectiveness of our model.

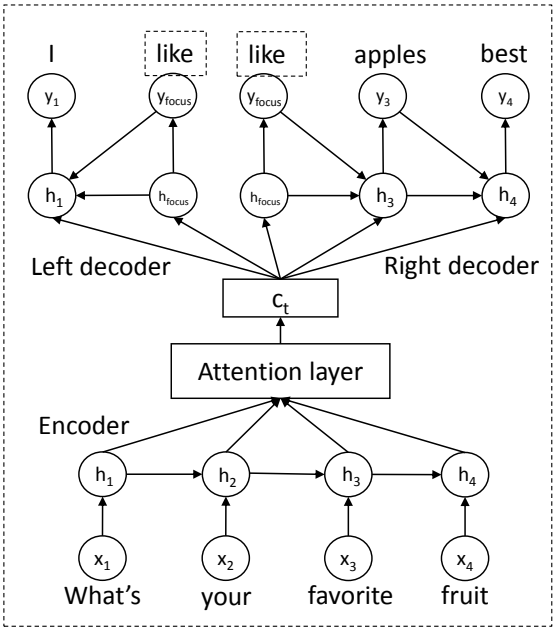

Figure 1: Focus of Attention Neural Responding Model. The model consists of three neural networks. Suppose the focus of attention in the response is "*apples*".

1 Introduction

Conversational response generation is a challenging task in natural language processing. As the amount of conversation data on social media websites such as Twitter and Weibo have tremendously increased and becomes available, many researchers focus on investigating data-driven conversational response generation. Neural conversation models which is in the form of sequence-to-sequence models have been successfully showed the ability of generating response to a given post. Sordoni (2015) first uses sequence-to-sequence neural network model which is integrated contextual information to generate conversational responses. Other works of conversational response generation (Vinyals and Le, 2015; Shang et al., 2015; Serban et al., 2015; Wen et al., 2015) also use the sequence-to-sequence models and have shown better performances than other response generation framework like statistical machine translation (Ritter et al., 2015) and Information retrieval (Ji et al., 2014). The work of Li (2015) uses a new objective function in neural conversation models and gets more appropriate responses.

In theory, recurrent neural networks(RNN) in sequence-to-sequence models are capable of handling long-term dependencies and generating a long sentence as long as models are well trained. Unfortunately, in practice, even though we use LSTM (Hochreiter and Schmidhuber, 1997; Li et al., 2015) or GRU (Cho et al., 2014; Chung et al., 2014; Shang et al., 2015) in sequence-to-sequence models, we still have problems in generating a

Response	"*I like apples best*"
Dependency	root(ROOT-0, like-2)
Left part	*I like*
Right part	*like apples best*

Table 1: Sentences produces by a sequence-to-sequence model using LSTM.

long response. For example in Table 2, if the response sentence is too long, the model may suffer from the error propagation and lead to produce a bad sequence. A good response generation system should be able to produce coherent, meaningful responses. To this end, it is important to reduce the influence that is from the lack of ability of RNN on generating long sentences. However, the study of this problem on sequence-to-sequence models has been missing so far. In order to fill this gap, we propose ABCDEFG Model, a novel encoder-decoder neural network-based model for response generation. The framework of the proposed model is shown in Figure 1. Our model belongs to the family of sequence-to-sequence neural networks. The difference between our model and its siblings is that, rather than generating a response in a single pass, our model splits the decoder into two recurrent neural networks to reduce the length of the response generated by a RNN. For example in Table 2, we do dependency parsing on the response, then we choose the word that is connected to the ROOT node (Marneffe and Manning, 2013) and denotes it as *rootword*, then we split the response into two parts by this *rootword*. Finally, our model generates the left part using left decoder that starts from the *rootword* and the same way that the right decoder using on producing the right part.

The paper makes the following contributions. We present a novel sequence-to-sequence neural network model for response generation. We take the syntactic structure of response into account. Our model changes the decoding way of previous sequence-to-sequence conversation model. It is a more natural way to generate a sentence that starts from the *rootword* of a sentence dependency-based parse tree. Theoretically, this model would generate a long, coherent and meaningful response. The left and right two decoder networks of our model output sentence starting from *rootword* of a response. As such, in our model's decoding way, we reduce the decoding step of a single RNN to almost a half and can mitigate the impact of error propagation which may lead us to a ridiculous focus word. It is a more natural way to generate a sentence that starts from the root of a sentence dependency-based parse tree. Experiments show that our model's ability to generate it and demonstrate that our model outperforms the baselines statistical machine translation(SMT) based methods as well as information retrieval based methods.

2 Focus of Attention Neural Responding Model

As illustrated in Figure 1, the basic idea of our model is to construct a hidden representation of a post sentence with an attention mechanism, then generate the left half of the response by firstly producing the *rootword*, and finally, decoding the right half of the response by the *rootword* and the left half response's hidden representation.

In more details, to one pair of the post and response, we consider a post sentence as a sequence $X = (x_1, x_2, ..., x_n)$ with length n and the corresponding response as a sequence $Y = (y_1, y_2, ..., y_r, ..., y_m)$ with length m, where x and y denotes the word in the sentence. y_r denotes the focus of attention word in the response sentence. Since we pay special attention to y_r, we divide Y into two sequences. The left part of the response sentence $Y_{left} = (y_r, ..., y_2, y_1)$ and the right part $Y_{right} = (y_r, ..., y_{m-1}, y_m)$. And our model is a condition model of the response given the post. We estimate:

$$\begin{aligned} P(Y|X) &= P(Y_{left}|X) + P(Y_{right}|Y_{left}, X) \\ &= \prod_{t=[r,1]} p(y_t|X, y_r, y_{r-1}, ..., y_t) + \\ &\prod_{t'=(r,m]} p(y_{t'}|X, Y_r, y_{r+1}, ..., y_{t'-1}) \end{aligned} \tag{1}$$

2.1 Encoder network

The encoder network receives the input sequence X and covers it into a context vector which can be treated as a representation of the post. There are many ways to encode the post sentence. The long short-term memory(LSTM) (Hochreiter and Schmidhuber, 1997; Li et al., 2015) and the gated recurrent unit (Cho et al., 2014; Chung et al., 2014; Shang et al., 2015) all have good performance on sequence modeling. We choose GRU because it has less parameters and is easier to train.

So in encoder network, the hidden state of time t is calculated by

$$\begin{aligned} h_t(enc) &= f_1(x_t, h_{t-1}(enc)) \\ &= GRU_{enc}(x_t, h_{t-1}(enc)) \end{aligned} \quad (2)$$

We use an attention mechanism to get the context vector c_t:

$$c_t = A([h_1(enc), h_2(enc), ..., h_n(enc)], h_t(dec)) \quad (3)$$

Where $A(\cdot)$ can be a linear network or a nonlinear network. What we use here is also a GRU recurrent neural network. $h_t(dec)$ is the hidden state $h_t(dec_l)$ in left decoder or $h_t(dec_r)$ in right decoder.

2.2 Decoder network

As with encoding, the left decoder is also a recurrent neural network with GRU. First, generate the focus*rootword* in response. Then, construct the left half of the response sentence. The procedure of it can be summarized as follows, for $t = r, r-1, ..., 1$:

$$h_t(dec_l) = f_2(y_t, h_{t-1}(dec_l), c_t) \quad (4)$$

$$p(y_r|\cdot) = Softmax(h_r(dec_l)) \quad (5)$$

$$p(y_t|\cdot) = Softmax(h_t(dec_l), y_r) \quad (6)$$

Where $f_2(\cdot)$ here is also a GRU, y_t is a one-hot representation,the initial state of $h_t dec_l$ is for generate the *rootword* word in response sentence, and so t is from $r, r-1$ down to 1. We use the last hidden state $h_1 dec_l$ as the representation of the left half of the sentence.

The right decoder is similar to the left. It first generates the *rootword* in response. Then the model feeds the *rootword*, the context vector and left half of the response sentence representation to generate the right half of the response sentence. Formally,(f_3 still GRU), for $t = r, r+1, ..., m$:

$$h_t(dec_r) = f_3(y_t, h_{t-1}(dec_r), c_t, h_1(dec_l)) \quad (7)$$

$$p(y_r|\cdot) = Softmax(h_r(dec_r)) \quad (8)$$

$$p(y_t|\cdot) = Softmax(h_t(dec_r), y_r) \quad (9)$$

At last, we combine the outputs of left($[y_r, ..., y_2, y_1]$) and right($[y_r, ..., y_{m-1}, y_m]$) to get the response sequence $(y_1, y_2, ..., y_r, ..., y_m)$. Since our model generates responses not only on the context vector like other sequence-to-sequence models, but also condition on the focus of attention in response, our model is called focus of attention neural responding model.

Models	$BLEU$
SMT-based	2.22
IR-based	2.22
Our model	3.33

Table 2: Performance on STC dataset for SMT-based, IR-based baselines and our model.

Post	**Response**
XXXXXX	XXXXXXX
XXXXXXXXXXXXX	XXXXXXXX
XXXXXXXXXXXXXX	XXXXXXX

Table 3: Sample responses generated by our model

3 Experiments

3.1 Dataset

We implement our model on short-text-conversation dataset (Wang et al., 2013; Shang et al., 2015), a large amount of short text conversation data available on social media Weibo from the NTCIR-12 STC task[1]. This dataset is for Weibo conversations and contains roughly 190 thousand posts and 5.6 million responses constructing almost 5.6 million post-response pairs.

To find the focus of attention in responses, we use do dependency parsing on every response by LTP (Che et al., 2010). As we know, the word which the root node points to is most responsible for determining the distribution of that sentence. It is the core of the whole sentence. So we treat this word as the focus of attention of a response.

3.2 Evaluation Method

For model evaluation, we used $BLEU$ (Papineni et al., 2002) following Sordoni (2015), Wen (2015) and Li (2016).

3.3 Results

The evaluation results using $BLEU$ are presented in Table 1. We compare our model with the Statistical machine translation(SMT) based method baseline and the information retrieval(IR) based method baseline. The results show a lot gains over these baselines. Table 2 provides examples of responses generated on the STC corpus. Our model produces natural responses to the input posts. The focus of attention in responses is very clear.

[1] http://ntcir12.noahlab.com.hk/stc.htm

4 Conclusion and Future Work

This paper proposes a novel response generation model called focus of attention neural responding model that generates responses by first produce the focus in it instead of outputting the first word in a response like other sequence-to-sequence models. Experiments on the STC dataset not only demonstrate that our proposed model outperforms the SMT and IR baselines but also show that this model generates coherent, natural and meaningful response to given inputs. In future work, we will incorporate our model to multi-turn conversations.

References

Wanxiang Che, Zhenghua Li, and Ting Liu. 2010. Ltp: A chinese language technology platform. In *Proceedings of the Coling 2010:Demonstrations*, pages 13–16.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *arXiv preprint arXiv:1406.1078*.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *arXiv preprint arXiv:1412.3555*.

Nomi Erteschik-Shir. 1997. In *The Dynamics of Focus Structure*. Cambridge University Press, Cambridge.

Alex Graves. 2013. Generating sequences with recurrent neural networks. In *arXiv preprint arXiv:1308.0850*.

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

Zongcheng Ji, Zhengdong Lu, and Hang Li. 2014. An information retrieval approach to short text conversation. In *arXiv preprint arXiv:1408.6988*.

Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of ACL*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *arXiv preprint arXiv:1510.03055v2*.

Marie-Catherine De Marneffe and Christopher D Manning. 2013. In *Technical report*. Stanford University.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.

Alan Ritter, Colin Cherry, and William Dolan. 2015. Data-driven response generation in social media. In *Proceedings of EMNLP*, pages 583–593.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2015. Building end-to-end dialogue systems using generative hierarchical neural network models. In *arXiv preprint arXiv:1507.04808*.

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

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Meg Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of NAACL-HLT*.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. In *Proceedings of ICML Deep Learning Workshop*.

Hao Wang, Zhengdong Lu, Hang Li, and Enhong Chen. 2013. A dataset for research on short-text conversations. In *Proceedings of EMNLP*.

Tsung-Hsien Wen, Milica Gasic, NikolaMrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of EMNLP*, pages 1711–1721.