Cluster-based Beam Search for Pointer-Generator Chatbot Grounded by Knowledge

Yik-Cheung Tam*
WeChat AI - Pattern Recognition Center
Tencent Inc.
wilsontam@tencent.com

Jiachen Ding*†
School of Electronic Information and Electrical Engineering
Shanghai Jiaotong University
ding222@sjtu.edu.cn

Cheng Niu and **Jie Zhou**
WeChat AI - Pattern Recognition Center
Tencent Inc.
niucheng@tencent.com
withtomzhou@tencent.com

Abstract

We present an end-to-end approach for knowledge-grounded response generation in Dialog System Technology Challenges 7 (DSTC7). Our system is trained by a pointer generator model, so that an output token in a response can either be generated or copied from conversation history or facts according to a trainable action probability distribution. Furthermore, to minimize generating meaningless responses, we propose using K-means to dynamically cluster and prune semantically similar partial hypotheses at each decoding step under a fixed beam budget. Moreover, we employ a language model to filter meaningless responses. Official evaluation results show that our proposed system achieved the first place in all primary automatic evaluation metrics and the overall human evaluation score.

Introduction

Recently, neural response generation has attracted a lot of attention (Vinyals and Le 2015; Shang, Lu, and Li 2015; Sordoni et al. 2015; Wen et al. 2016). The task is mainly based on single-turn question-response pairs ignoring conversational context and external knowledge. One of the DSTC7 challenge focuses on conversational chatbot grounded by external knowledge going beyond simple chitchat (Ghazvininejad et al. 2018). For instance, a news article about certain topic and human post histories are given. The goal of a chatbot is to generate relevant and interesting responses under long multi-turn conversation history and facts. To build a knowledge-grounded conversational chatbot, there are several issues that requires further attention, including (1) conversational history modeling, (2) integration of unstructured external knowledge, (3) diversity in response generation that is relevant and interesting.

In this paper, we attempt to address these issues with a proposed system as shown in Figure 1, including data preprocessing and fact retrieval, end-to-end encoder-decoder attention modeling of conversational history and facts, and improved decoding strategies. Our contributions are highlighted as follows: First, conversation history and facts are incorporated into an end-to-end model that enables copying of out-of-vocabulary words in the conversation history and facts. Second, we propose a beam search strategy to improve response diversity driven by K-means clustering and pruning of partially decoded hypotheses that are semantically similar under a fixed beam budget. Third, we employ a language modeling approach to filter out safe responses such as "this is the best thing i have ever seen", "i don't know what you mean" and etc.

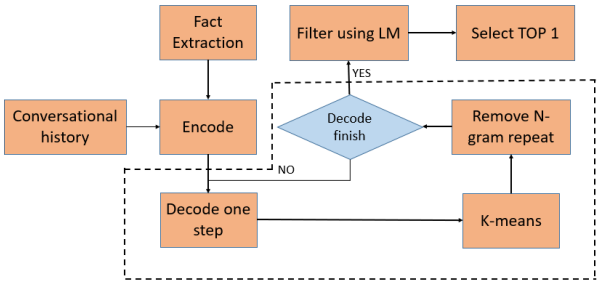

Figure 1: The proposed system flowchart for knowledge-grounded conversational chatbot. The flow inside the dashed box shows the process of decoding one step during beam search. The process will run until the ending condition is met.

Related Work

Sequence-to-sequence model (Sutskever, Vinyals, and Le 2014) has been a popular approach in many domains including neural response generation (Vinyals and Le 2015; Shang, Lu, and Li 2015; Wen et al. 2016; Serban et al. 2016), to name just a few. Attention mechanism has been crucial in a lot of NLP tasks such as machine translation (Bahdanau, Cho, and Bengio 2015; Wu et al. 2016; Vaswani et al. 2017), machine reading comprehension (Seo et al. 2017; Wang et al. 2017; Wang and Jiang 2017) and natural language inference (Wang and Jiang 2016). To deal with OOV generation, (Gu et al. 2016; See, Liu, and Manning 2017) proposed a sequence-to-sequence model with copying mechanism for neural response generation and neural summarization. (Eric and Manning 2017) further extended this

*Equal contribution
†This work was done when Jiachen Ding was an intern at WeChat AI - Pattern Recognition Center, Tencent Inc.

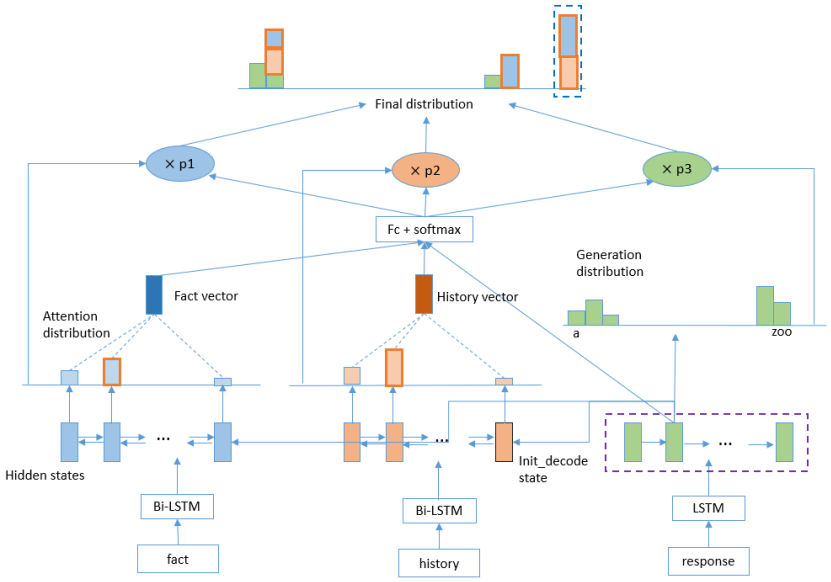

Figure 2: A pointer generator approach that enables copying mechanism for conversation history (H) and facts (F). For each decoding time step, three action probabilities are calculated, namely copying a token from H, copying a token from F, and generating a token. The final word probability distribution is the linear interpolation of these three probability distributions. Out-of-vocabulary (OOV) token receives probability mass from the attention distributions of history and facts.

idea to copying entities of a knowledge base mentioned in the dialog context.

To improve diverse in response generation, (Li et al. 2016) proposed a maximum mutual information (MMI) approach for training and decoding. Empirically, we found that the MMI approach can alleviate the diversity issue but may hurt fluency of the generated response. (Zhou et al. 2017) proposed an encoder-diverter-decoder framework to address the 1-to-n relationship between a post and multiple responses so that diverse responses are generated with latent factors. (Su et al. 2018) searched for a latent code space that maximizes the mutual information between the previous, current, and next sentences to improve diversity. (Zhang et al. 2018) introduced Adversarial Information Maximization and optimization of variational lower bound on pairwise mutual information between query and response to boost informativeness and diversity. (Baheti et al. 2018) added distributional constraints such as latent topics into the model to generate more interesting responses. (Liu et al. 2018) down-weighted frequent response patterns to avoid safe responses. On the beam search improvement, (Li, Monroe, and Jurafsky 2016) proposed a hypothesis rank penalty in beam search to improve diversity. (Shao et al. 2016) introduced a stochastic beam-search algorithm with segment-by-segment reranking to inject diversity earlier in the decoding process. (Vijayakumar et al. 2018) improved beam search that decoded a list of diverse outputs via optimizing a diversity-augmented objective.

To integrate external knowledge, there has been research in the DSTC6 challenge (Hori and Hori 2017; Galley et al. 2017). (Ghazvininejad et al. 2018) employed memory network for encoding facts with great progress. (Madotto, Wu, and Fung 2018) applied memory network to store dialog history and structured knowledge base (KB) and used pointer generator to copy token from dialog history or KB token for task-oriented dialog systems. A hierarchical version was proposed in (Raghu, Gupta, and Mausam 2018).

Our contributions are two-folded: First, we generalize pointer generator (See, Liu, and Manning 2017) that enables copying mechanism over conversation history and facts (and potentially other sources) with multiple attention distributions. Second, our proposed K-means beam search clusters hypotheses dynamically at each decoding step so that semantically similar hypotheses are grouped and pruned. This differs from the approach (Vijayakumar et al. 2018) that assumes fixing previous hypotheses groups when performing beam search in the current hypothesis group. In addition, we employ an N-gram language model to filter away safe responses.

Data Cleaning and Fact Retrieval

As illustrated in Figure 1, one crucial step is to perform data cleaning on raw facts. In the DSTC7 challenge, web pages corresponding to the conversation posts in Reddit are crawled in HTML format. Web pages contain a lot of irrelevant contents such as advertisements, navigation bars, footers and etc. On average, the length of an article is about 4000 words which are too large to fit into a GPU. Since a conversation history usually focuses on a specific part of an article, it may not be necessary to use the full content. Similar to (Ghazvininejad et al. 2018), we retrieve facts (sentences

in an article) that are most relevant to a conversational history with a maximum of 500 word tokens. Each sentence in an article is treated as a unit. Similarity between a conversational history H and a fact sentence F in an article is calculated as follows:

$$sim(H, F) = \sum_{w \in H} idf(w) \times count(w \; in \; F) \quad (1)$$

where w is a unique word in H. $idf(w)$ is the inverse document frequency for w. $count(\cdot)$ calculates the number of times w occurs in F. Relevant fact sentences are selected and concatenated in the order of appearance in the original article. Finally, we want to generate a response R:

$$Pr(R|H, F) \quad (2)$$

Proposed End-To-End System

The overall structure of our end-to-end model is shown in Figure 2 that illustrates an encoder and a decoder in one decoding time step.

Encoder

We encode H and F separately using a single-layer bi-directional LSTM, giving hidden state sequences $H^H = \{h_1^H, h_2^H, ..., h_i^H, ..., h_L^H\}$ and $H^F = \{h_1^F, h_2^F, ..., h_j^F, ..., h_T^F\}$ respectively. Then we take the concatenation of the first and last hidden states of conversational history and project them linearly as an initial state of a decoder.

Decoder

Our decoder consists of three parts: (1) attention mechanism over conversation history and facts, (2) mode prediction, and (3) output word probability estimation.

Attention Mechanism At each time step t, the decoder focuses on different parts of encoder inputs via the attention mechanism (Bahdanau, Cho, and Bengio 2015). Let h_t^R denote the hidden state of an output response at time step t. We apply attention over H^H as follows:

$$e_{ti}^H = v_H^T \cdot \tanh(W_h^H \cdot h_i^H + W_r^H \cdot h_t^R + b^H) \quad (3)$$

$$\alpha_{ti}^H = Softmax(e_{ti}^H) \quad (4)$$

$$h_t^{H*} = \sum_{i=1}^{L} \alpha_{ti}^H h_i^H \quad (5)$$

where v_H, W_h^H, W_r^H and b^H are learnable parameters. The attention distribution α_t^H serves as a probability distribution over word tokens in conversation history, allowing OOV tokens in conversation history to be copied to response outputs. Likewise, we apply attention over H^F as follows:

$$e_{tj}^F = v_H^F \cdot \tanh(W_h^F \cdot h_j^F + W_r^F \cdot h_t^R + b^F) \quad (6)$$

$$\alpha_{tj}^F = Softmax(e_{tj}^F) \quad (7)$$

$$h_t^{F*} = \sum_{j=1}^{T} \alpha_{tj}^F h_j^F \quad (8)$$

where v_F, W_h^F, W_r^F and b^F are learnable parameters for facts.

Mode Prediction Traditional sequence-to-sequence model only generates according to a fixed vocabulary. This suffers from the OOV problem that a lot of low-frequent but informative words are forced to map to the unknown token. Generating the unknown token in output response is not a good idea as the response is incomprehensible and may require recovery of unknown token which may be difficult. Therefore, we extend the pointer-generator approach to enable the copying mechanism. In the original pointer generator (See, Liu, and Manning 2017), there are two modes: (1) Generate a token; (2) Copy a token. In our version, we extend pointer generator that supports three modes: (1) Generate a token; (2) Copy a token from conversational history; (3) Copy a token from facts. We formulate mode prediction at each decoding time step t as multi-class classification using Softmax:

$$\begin{aligned} &Pr(mode = m|t, H, F) = \\ &Softmax(FF(h_t^{F*} \oplus h_t^{H*} \oplus h_t^R \oplus x_t)) \end{aligned} \quad (9)$$

where we feed available hidden vectors from attention and decoder into a feed-forward neural network followed by Softmax over three modes. x_t denotes the input vector to the decoder at time t.

Word prediction We compute the final word distribution as a linear interpolation of vocabulary distributions from three modes:

$$Pr(w|t, H, F) = \sum_{m=1}^{3} Pr(m|t, H, F) \cdot Pr_m(w|t, H, F) \quad (10)$$

where $Pr_m(w|t, H, F) \propto \sum_{i=1}^{N_m} \alpha_{ti}^{(m)} \cdot \delta(w_i, w)$ and m is the mode index corresponding to copying mechanism from conversation history or facts. Attention weights $\alpha_{ti}^{(m)}$ serve as fractional counts for word tokens occurred in conversation history and facts. For generation mode, the in-vocabulary tokens are generated via $Softmax(W_g \cdot h_t^R)$ and OOV tokens are assigned with zero probabilities. In addition, OOV tokens in an output response that do not occur in conversation history or facts will be mapped into the unknown token (UNK) since these tokens can never be copied from conversation history or facts.

Cluster-based Beam Search

Traditional beam search attempts to find the best hypothesis that maximizes $Pr(R|H, F)$. It is well known that this approach tends to generate safe answers. We conjecture that as beam search progresses at every decoding step, partial hypotheses are formed and some of them are semantically similar. It hurts diversity when keeping all semantically similar hypotheses under a fixed beam budget. Therefore, we propose using K-means to cluster hypotheses into K groups, followed by hypothesis pruning per group.

K-means Traditional beam search is a special case of our proposed approach with one cluster. As outlined in Algorithm 1, in each decoding step, we extend each candidate and choose top $BS \times 2$ according to the log probability

Algorithm 1: Beam search with K-means

Input: Beam size BS, Candidates C initialized with start symbol
Output: Final response rsp
Data: Language model threshold lm_{th}

while *Number of completed hypothesis does not reach BS or maximum decoding step is not reached* **do**
  for *i in BS* **do**
    $tmpHyps$=Top-N(Extend($C[i]$), $BS \times 2$);
    Remove hyps in $tmpHyps$ with repeated N-grams or UNK;
    Save $tmpHyps$ to extended candidates;
  end
  Perform K-means over extended candidates;
  for *candidates in each cluster* **do**
    Sort candidates by partial log-prob scores;
    Choose top BS/K candidates;
    Put candidates with end symbol in R;
    Put incomplete candidates in C_{new};
  end
  $C \leftarrow C_{new}$
end
Sort R according to log-prob scores;
for hyp *in* R **do**
  if $score_{lm}(hyp) < lm_{th}$ **then**
    $rsp \leftarrow hyp$;
    break;
  end
end

scores from the model. Then we employ K-means to cluster all extended candidates, forming K clusters. We only choose top BS/K candidates in each cluster for next decoding step. Under a fixed beam search budget, there is a tradeoff between the number of clusters K and the number of top candidates to keep in each cluster. We use the average word embedding of partially decoded hypotheses as features for K-means clustering. Ideally, semantically similar candidates such as safe answers would gather in one cluster, while other clusters capture other meanings. Our approach tries to maximize diversity in the hypothesis space without opening the search beam further to retain the potentially low-scored but informative hypotheses.

Remove Repeated N-grams Since we use attention mechanism in decoder, it is likely that the model pays attention to the previously attended tokens in conversation history or facts. Therefore, repeated N-grams may be generated. To improve fluency, hypotheses with repeated bigrams are removed from further consideration during beam search.

Filter Meaningless Response using LM After obtaining the top BS candidates from beam search, it is likely that safe responses exist as top candidates. We train a trigram language model using KenLM (Heafield 2011) on meaningless responses. Then we filter the candidates when their language model scores are above a preset threshold. The first candidate with score below the threshold is chosen as the final response. For building the language model, we decode the official validation set with the trained pointer generator model using the regular beam search. We expect that most of the top-N responses from the regular beam search are meaningless responses and can be used to train the language model.

Table 1: Dataset statistics for DSTC7 Track 2.

	Train set	Validation set	Test set
Samples	1,408,951	4,542	13,108

Table 2: Hyper-parameter settings.

Name	Value
Vocabulary size	100,000
Word embedding size	300
LSTM hidden size	150
Batch size	128
Beam size (BS)	50
K-means clusters	10
N-gram repeat order	2
LM threshold (lm_{th})	-35
Learning rate for Adam	0.0005
Maximum gradient norm	2

Experiments

Setup

We generated our dataset using the crawling script provided by the DSTC7 organizers [1]. Since the quality of training data is crucial for model training, we further kept the data by restricting the response length within 8 and 20 tokens. Short responses may not be informative while it is very hard for pointer generator to generate very long responses anyway. We used spaCy[2] to tokenize our dataset. Table 1 summarizes the final data statistics for experiments.

We implemented our model by modifying the pointer-generator code [3] to incorporate attention from conversation history and facts. We set the minimum count of 5 to select the top 100,000 words as vocabulary for generation. We used the pretrained 300-dimension Glove embeddings (Pennington, Socher, and Manning 2014) to initialize word embeddings and kept fixed. Embeddings for OOV tokens, if found in Glove, were used. Otherwise, their embeddings were randomly initialized. We employed Adam (Kingma and Ba 2015) for optimization with initial learning rate of 0.0005. Other hyper-parameters are shown in Table 2. We also implemented the beam search decoder with K-means and other hypothesis filtering strategies as described in the previous section.

[1] https://github.com/DSTC-MSR-NLP/DSTC7-End-to-End-Conversation-Modeling

[2] https://spacy.io

[3] https://github.com/abisee/pointer-generator

Table 3: Automatic evaluation results. A total of 2208 test samples were evaluated. Best non-baseline results are marked in **bold**.

Name	NIST-4	BLEU-4	Meteor	Entropy-4	Div-1	Div-2	Avg len
Baseline (constant)	0.184	2.87 %	7.48 %	1.609	0.000	0.000	8
Baseline (random)	1.637	0.86%	5.91%	10.467	0.160	0.647	19.192
Baseline (seq2seq)	0.916	1.82%	6.96%	5.962	0.014	0.048	10.604
Team C/E	1.515	1.32%	6.43%	7.639	0.053	0.171	12.674
Team G	2.040	1.05%	7.48%	**10.057**	0.108	**0.449**	22.336
Our system w/ K-means	**2.523**	1.83%	**8.07**%	9.030	0.109	0.325	15.133
Our system w/o K-means	1.771	**1.94**%	7.64%	8.194	0.094	0.267	12.770
Human	2.650	3.13%	8.31%	10.445	0.167	0.670	18.757

Table 4: Human evaluation results. Best non-baseline results are marked in **bold**.

	Relevance		Interest		Overall	
Model	Mean Score	95% CI	Mean Score	95% CI	Mean Score	95% CI
Baseline (constant)	2.60	(2.560, 2.644)	2.32	(2.281, 2.364)	2.46	(2.424, 2.500)
Baseline (random)	2.32	(2.269, 2.371)	2.35	(2.303, 2.401)	2.34	(2.288, 2.384)
Baseline (seq2seq)	2.91	(2.858, 2.963)	2.68	(2.632, 2.730)	2.80	(2.748, 2.844)
Team C/E (2nd place)	**3.05**	**(3.009, 3.093)**	2.77	(2.735, 2.812)	2.91	(2.875, 2.950)
Team G (3rd place)	2.82	(2.771, 2.870)	2.57	(2.525, 2.619)	2.70	(2.650, 2.742)
Our system (1st place)	2.99	(2.938, 3.042)	**2.87**	**(2.822, 2.922)**	**2.93**	**(2.882, 2.979)**
Human	3.61	(3.554, 3.658)	3.49	(3.434, 3.539)	3.55	(3.497, 3.596)

For official evaluation, we submitted two systems, one with K-means beam search as the primary system and the other without it as the secondary system. Otherwise, both systems used the same trained pointer generator model.

Evaluation Results

There were seven competing teams participated in the DSTC7 evaluation. All response outputs were scored with primary automatic evaluation metrics namely NIST (Doddington 2002), BLEU (Papineni et al. 2002) and Meteor (Denkowski and Lavie 2014). Other metrics such as DIV-1 and DIV-2 (also known as distinct-1 and distinct-2) (Li et al. 2016) and Entropy1-4 (Zhang et al. 2018) were also employed to measure diversity. Human evaluation was performed and human evaluators rated system responses in terms of "relevance and appropriateness" and "interest and informativeness" with five levels: strongly agree, agree, neutral, disagree, strongly disagree. Each response was scored by three judges. 1k test samples were carefully chosen by the DSTC7 organizers for human evaluation.

The DSTC7 organizers provided three baselines: (1) constant: always responds: "i don't know what you mean ."; (2) random: randomly picks a response from the training data; (3) seq2seq: trained with vanilla Keras sequence-to-sequence model [4].

Automatic Evaluation As shown in Table 3, our system achieved the best results on all primary metrics using NIST-4, BLEU-4 and Meteor. Moreover, using K-means beam search improved performance on almost all primary metrics and all diversity metrics effectively. On the other hand, BLEU metric alone may not be reliable since the constant baseline obtained the best BLEU-4 even though the response was totally meaningless. This may show that there were many N-grams in the reference responses that overlapped with this meaningless response. On the other hand, the constant baseline got a bad NIST score because the meaningless N-grams carried less information and were down-weighted accordingly. In terms of the average response length, our systems produced longer responses compared to the sequence-to-sequence baseline. In addition, K-means beam search generated longer responses compared to not using it. On average, Human responses were longer than our systems while Team G generated even longer responses with 22 tokens on average. In terms of the ability of outputing OOVs that were not covered by the top 100k vocabulary, our systems produced 97 and 57 unique OOVs over the submitted test responses using K-means beam search and traditional beam search respectively. K-means beam search appeared to copy more OOVs compared to traditional beam search.

Human Evaluation Table 4 shows the human evaluation results. Compared to the vanilla sequence-to-sequence baseline, our system surpassed the baseline significantly at 95% level of confidence. Moreover, our system achieved the best result in the "interest and informativeness" category with 95% level of confidence compared to the second-place team. Overall, our system achieved the first place in the competition.

Discussion

Our system performed the best in interest and informativeness. Moreover, safe responses were rarely appeared in our

[4] https://github.com/DSTC-MSR-NLP/DSTC7-End-to-End-Conversation-Modeling/tree/master/baseline

Table 5: Case Study

Conversational history	til : chinese philanthropist , run run shaw (1907-2014) gave $ 1 + / - billion to fund education , aiding over 6000 construction projects . because so many buildings (5000 +) bear his name , yifu , it is often confused as a generic name .
Relevance Interest	1.33 3.00
w/ K-means	james cameron is the best name ever . i'm pretty sure he voiced the tianyi of tesla .
w/o K-means	he's a chinese restaurant . it's his name !
Conversational history	til in 1950 , myxoma virus (a type of pox virus) was deliberately released into the rabbit population in australia as a means of population control ,causing the death of over 500 million rabbits
Relevance Interest	2.67 1.00
w/ K-means	if you have a virus infested population , i'd be scared of mice .
w/o K-means	if i remember correctly , population mice are poisonous .
Conversational history	til that in greek mythology zeus thought that the earth is overpopulated and trojan war was instigated to depopulate the earth .
Relevance Interest	4.67 4.67
w/ K-means	james cameron wrote a book about the origins of war on this .
w/o K-means	i've never heard of this , but i didn't know it was aliens .

results. However, according to human evaluation results, our responses were worse than the second-place team in terms of relevance and appropriateness. Therefore, we picked some examples with low scores for analysis. In Table 5, the first case talks about a Chinese philanthropist who has various different names such as "run run shaw" and "yifu". Our model focused on the name part which was reasonable according to the human posts [5]. However, our model generated a sentence about another person "James Cameron" and "Tianyi of Tesla". "Tianyi" turns out to be a brand for automated bus, but it is not manufactured by Tesla. Furthermore, "James Cameron" has nothing related to "Tianyi of Tesla". This example shows that our model has flaws in relevance and fact consistency issues. The second case [6] has a low interest score. Our model captures the word "virus" and generates something about it. The response seems fluent but logically unreasonable due to the generated phrase "of mice" in the response. In the third case, our model picked the war topic and generated a relevant and interesting response using K-means beam search. The article [7] indeed contains the phrase "origins of the war". James Cameron, who is famous for sci-fi and blockbuster film-making, did not occur in conversation history or facts and thus the name was generated totally. In comparison, traditional beam search generates a meaningless response, showing the effectiveness of our K-means beam search.

[5] https://www.reddit.com/r/todayilearned/comments/7ezity/til_chinese_philanthropist_run_run_shaw_19072014/

[6] https://www.reddit.com/r/todayilearned/comments/76elb2/til_in_1950_myxoma_virus_a_type_of_pox_virus_was/

[7] https://en.wikipedia.org/wiki/Trojan_War#Origins_of_the_war

Conclusion

The DSTC7 challenge has provided a valuable opportunity to research in the knowledge-grounded chatbot on conversational dialogs. We have made improvement in the pointer generator model that copies OOV words in conversational history and facts, and new decoding strategies to produce more diverse and informative responses. In automatic and human evaluation, our proposed system has achieved the first place overall among all competing teams. There are still unresolved issues. Apart from relevance and informativeness, response consistency with external knowledge remains a big challenge for future exploration.

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*.

Baheti, A.; Ritter, A.; Li, J.; and Dolan, B. 2018. Generating more interesting responses in neural conversation models with distributional constraints. In *Proceedings of the 2018 conference on empirical methods in natural language processing (EMNLP)*.

Denkowski, M., and Lavie, A. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, 376–380.

Doddington, G. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the second international conference on Human Language Technology Research*, 138–145. Morgan Kaufmann Publishers Inc.

Eric, M., and Manning, C. D. 2017. A copy-augmented sequence-to-sequence architecture gives good performance on task-oriented dialogue. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*.

Galley, M.; Brockett, C.; Dolan, B.; and Gao, J. 2017. The msr-nlp system at dialog system technology challenges 6. In *Proceedings of the 6th Dialog System Technology Challenges (DSTC6) Workshop*.

Ghazvininejad, M.; Brockett, C.; Chang, M.-W.; Dolan, B.; Gao, J.; Yih, W.-t.; and Galley, M. 2018. A knowledge-grounded neural conversation model. In *AAAI*.

Gu, J.; Lu, Z.; Li, H.; and Li, V. O. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of ACL*.

Heafield, K. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, WMT '11, 187–197.

Hori, C., and Hori, T. 2017. End-to-end conversation modeling track in DSTC6. In *Dialog System Technology Challenges*.

Kingma, D., and Ba, J. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, B. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of NAACL-HLT*.

Li, J.; Monroe, W.; and Jurafsky, D. 2016. A simple, fast diverse decoding algorithm for neural generation. *arXiv preprint arXiv:1611.08562*.

Liu, Y.; Bi, W.; Gao, J.; Liu, X.; Yao, J.; and Shi, S. 2018. Towards less generic responses in neural conversation models: A statistical re-weighting method. In *Proceedings of the 2018 conference on empirical methods in natural language processing (EMNLP)*.

Madotto, A.; Wu, C.-S.; and Fung, P. 2018. Mem2Seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 1468–1478. Association for Computational Linguistics.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, 311–318. Association for Computational Linguistics.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Raghu, D.; Gupta, N.; and Mausam. 2018. Hierarchical-pointer generator memory network for task oriented dialog. *arXiv preprint arXiv:1805.01216*.

See, A.; Liu, P. J.; and Manning, C. D. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of ACL*.

Seo, M.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2017. Bidirectional attention flow for machine comprehension. In *Proceedings of ICLR*.

Serban, I. V.; Sordoni, A.; Bengio, Y.; Courville, A. C.; and Pineau, J. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, volume 16, 3776–3784.

Shang, L.; Lu, Z.; and Li, H. 2015. Neural responding machine for short-text conversation. In *Proceedings of ACL*.

Shao, L.; Gouws, S.; Britz, D.; Goldie, A.; Strope, B.; and Kurzweil, R. 2016. Generating long and diverse responses with neural conversation models. `https://openreview.net/forum?id=HJDdiT9gl`.

Sordoni, A.; Galley, M.; Auli, M.; Brockett, C.; Ji, Y.; Mitchell, M.; Nie, J.-Y.; Gao, J.; and Dolan, B. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of NAACL-HLT*.

Su, H.; Shen, X.; Li, W.; and Klakow, D. 2018. NEXUS network: Connecting the preceding and the following in dialogue generation. In *Proceedings of the 2018 conference on empirical methods in natural language processing (EMNLP)*.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Vijayakumar, A. K.; Cogswell, M.; Selvaraju, R. R.; Sun, Q.; Lee, S.; Crandall, D.; and Batra, D. 2018. Diverse beam search: Decoding diverse solutions from neural sequence models. In *AAAI*, 7371–7379.

Vinyals, O., and Le, Q. V. 2015. A neural conversational model. In *ICML*.

Wang, S., and Jiang, J. 2016. Learning natural language inference with lstm. In *Proceedings of NAACL-HLT*.

Wang, S., and Jiang, J. 2017. Machine comprehension using match-lstm and answer pointer. In *Proceedings of ICLR*.

Wang, W.; Yang, N.; Wei, F.; Chang, B.; and Zhou, M. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, 189–198.

Wen, T.-H.; Gasic, M.; Mrksic, N.; Rojas-Barahona, L. M.; Su, P.-H.; Ultes, S.; Vandyke, D.; and Young, S. 2016. Conditional generation and snapshot learning in neural dialogue systems. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*.

Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zhang, Y.; Galley, M.; Gao, J.; Gan, Z.; Li, X.; Brockett, C.; and Dolan, B. 2018. Generating informative and diverse conversational responses via adversarial information maximization. In *Advances in Neural Information Processing Systems*.

Zhou, G.; Luo, P.; Cao, R.; Lin, F.; Chen, B.; and He, Q. 2017. Mechanism-aware neural machine for dialogue response generation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*.