Paper review

You Impress Me: Dialogue Generation via Mutual Persona Perception (ACL 2020)

Presentation: **Jeiyoon Park**
W AI Automation

1/12/2021

https://arxiv.org/pdf/2004.05388.pdf

Outline

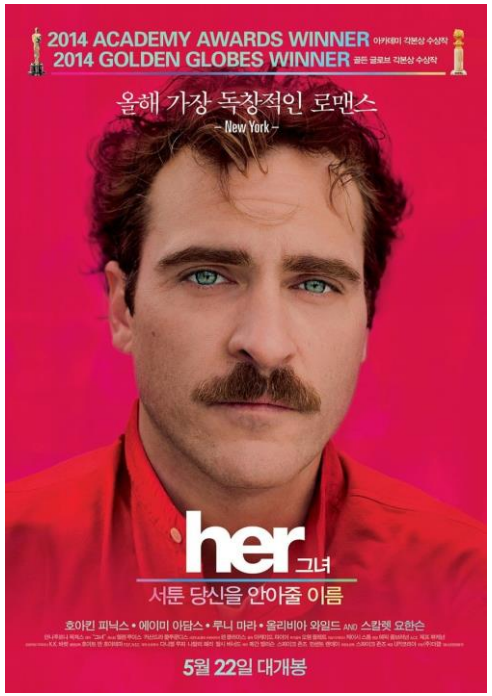
1. Contribution
2. Method
3. Experiments
4. Conclusion

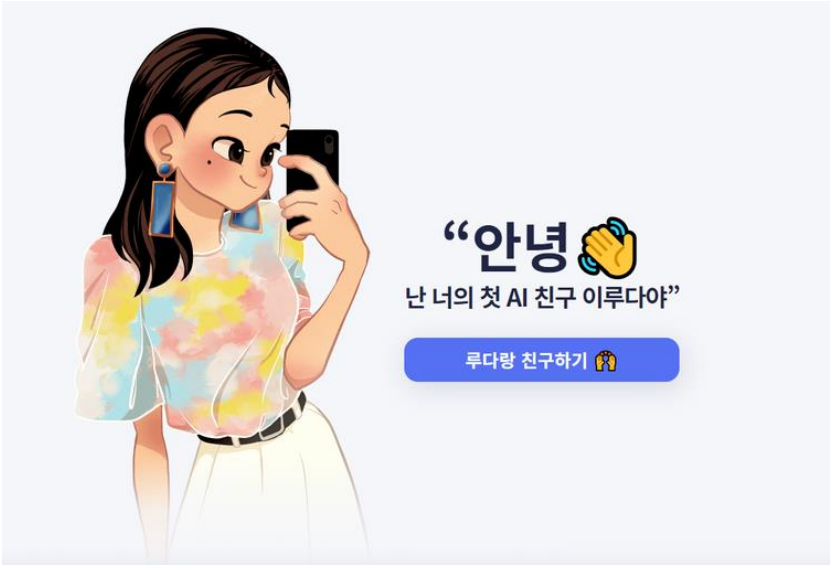
Outline

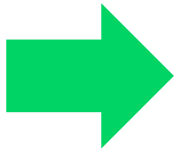
1. Contribution
2. Method
3. Experiments
4. Conclusion

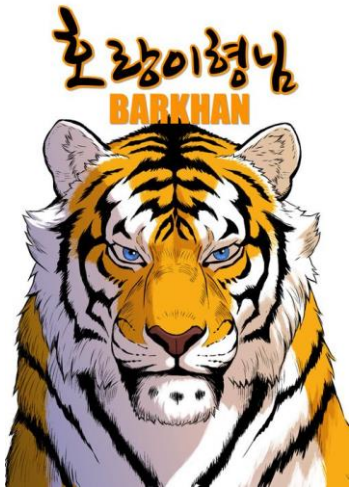
Contribution

0. Why Persona-grounded Conversation?









Contribution

1. Persona-grounded Conversation

1) Current chit-chat systems tend to generate uninformative responses

A: Where are you going? (1)
B: I'm going to the restroom. (2)
A: See you later. (3)
B: See you later. (4)
A: See you later. (5)
B: See you later. (6)
...
...

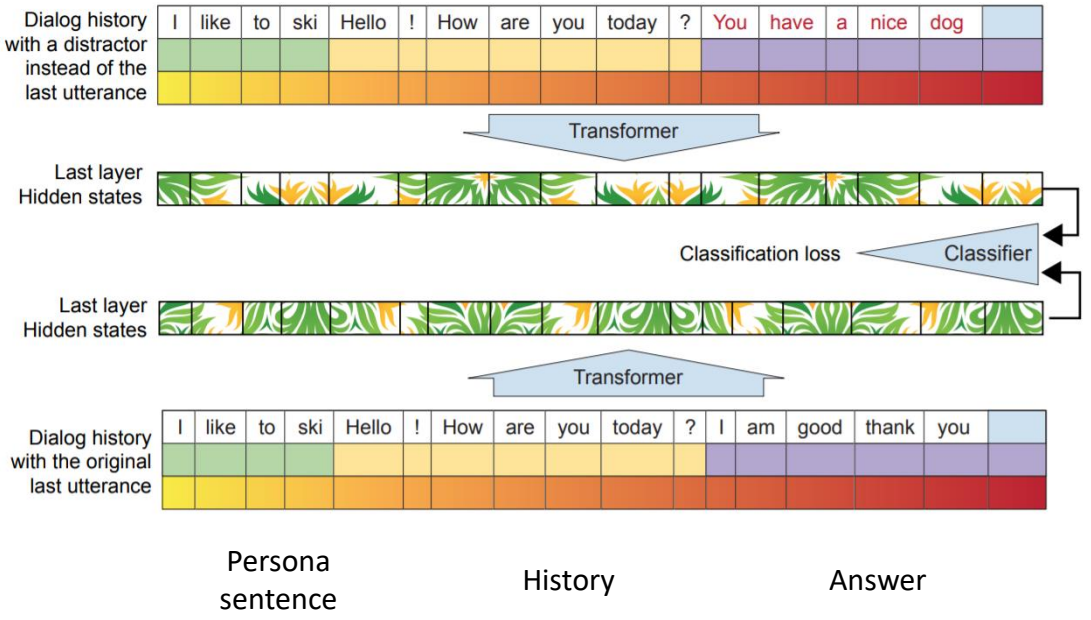
A: how old are you? (1)
B: I'm 16. (2)
A: 16? (3)
B: I don't know what you are talking about. (4)
A: You don't know what you are saying. (5)
B: I don't know what you are talking about . (6)
A: You don't know what you are saying. (7)
...

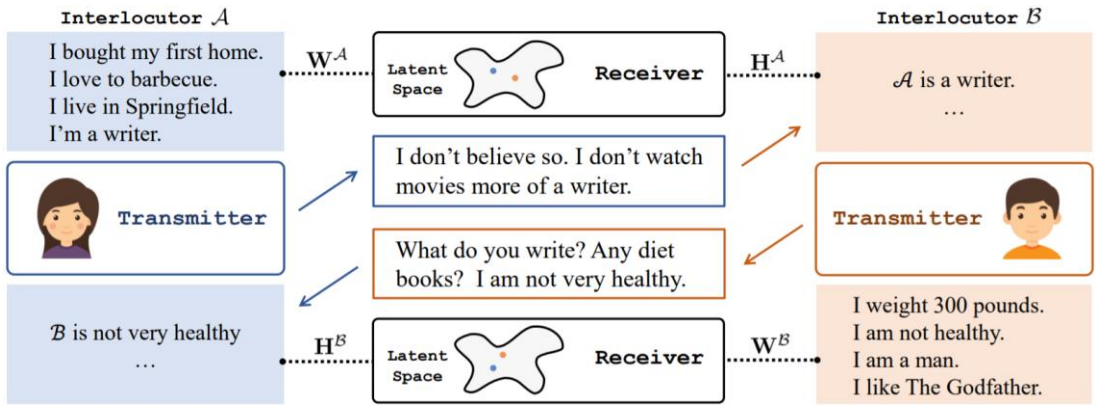
2) They focus more on mimicking the style of human-like responses, leaving understudied the aspects of explicitly modeling understanding between interlocutors

Contribution

2. Why This Paper?

1) Cognitive science-based approach: Understanding between interlocutors is an essential signal for high quality chit-chat conversation





Contribution

2. Why This Paper?

2) To this end,

- Supervised training and self-play fine-tuning

- Transmitter and Receiver

- Reward shaping

Contribution

3. Summary

1) This paper tries to solve

- Uninformative responses
- Understanding between interlocutors

2) Contribution

- Self-play fine-tuning
- Transmitter and Receiver framework

Outline

1. Contribution
2. Method
3. Experiments
4. Conclusion

Method

1. Transmitter

1) Transmitter is Transformer decoder (Radford et al., 2018), a.k.a., GPT-1

2) Transmitter treats dialog generation as a sequence generation problem

3) Training Procedure

- Supervised dialog generation
- Self-play fine-tuning

4) W^A is persona, h_n^A is dialog history, x_n^A is utterance at n-th turn

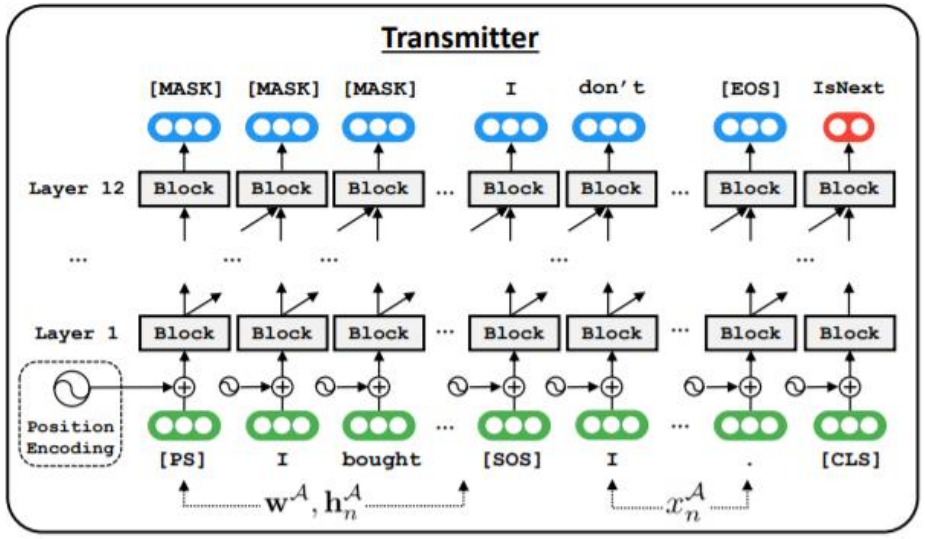


Figure 3: The overall architecture of Transmitter. "Block" is short for "Transformer Block". Arrows ↗ bridge the current block to subsequent blocks of its following layer. Position encoding is to incorporate position information into block by assigning an embedding for each absolute position in the sequence. Here we omit the architecture inside the block, and refer the readers to Vaswani et al. (2017) for more details. [MASK] tokens are ignored in the training objective.

Method

1. Transmitter

5) **Supervised dialog generation**

- Given a training instance (W^A, h_n^A, x_n^A):

$$\mathcal{L}_{\text{mle}} = \sum_t \log p_\theta(x_{n,t}^{\mathcal{A}} \mid \mathbf{w}^{\mathcal{A}}, \mathbf{h}_n^{\mathcal{A}}, x_{n,<t}^{\mathcal{A}}) \quad \text{[Maximize]}$$

,where θ is parameter of Transmitter

- During inference, *Beam Searched* (size 2) candidates are chosen:

$$x_n^{\mathcal{A}*} = \arg\max_{\hat{x}_n^{\mathcal{A}}} \frac{\log p_\theta(\hat{x}_n^{\mathcal{A}} \mid \mathbf{w}^{\mathcal{A}}, \mathbf{h}_n^{\mathcal{A}})}{|\hat{x}_n^{\mathcal{A}}|} \quad \text{[Simplified]}$$

Method

1. Transmitter

5) **Supervised dialog generation**

- Next utterance prediction

$$x_n^{\mathcal{A}^*} = \arg\max_{\hat{x}_n^{\mathcal{A}}} \left(\alpha \cdot \frac{\log p_\theta(\hat{x}_n^{\mathcal{A}} \mid \mathbf{w}^{\mathcal{A}}, \mathbf{h}_n^{\mathcal{A}})}{|\hat{x}_n^{\mathcal{A}}|} + (1-\alpha) \cdot \log p_\theta(y_n = 1 | \mathbf{w}^{\mathcal{A}}, \mathbf{h}_n^{\mathcal{A}}, \hat{x}_n^{\mathcal{A}}) \right)$$

,where $y_n = 1$ is the signal indicating the generated response $\hat{x}_n^A$ is predicted as the next utterance, and α is hyper-parameter (0.1)

Method

1. Transmitter

6) **Self-play model fine-tuning**

- Although supervised dialog generation alone can be used to mimic human-like responses, it doesn't inherently target understanding.

- This paper applies Self-play to simulate the communication between two Transmitters A *(user, env)* and B *(agent).*

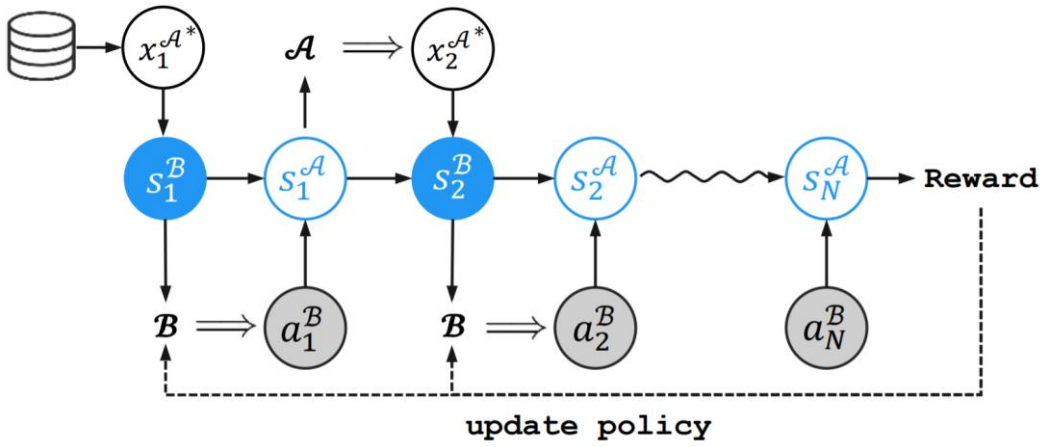
- Markov Decision Process (MDP)

state: $s_n^B = \{W^B,\ h_n^B\}$
action: a_n^B is the response to be generated
reward: R_1, R_2, R_3

Method

1. Transmitter

6) Self-play model fine-tuning



$$\mathcal{L}_{\mathrm{rl}} = \mathbb{E}_{a_n^{\mathcal{B}} \sim p_\theta(a_n^{\mathcal{B}} | s_n^{\mathcal{B}})}[R(a_n^{\mathcal{B}})]$$

, where x_n^{A*} is directly taken from the dataset

- Using policy gradient: *REINFORCE (Sutton et al., 1999)*,

$$\nabla_\theta \mathcal{L}_{\mathrm{rl}} = \mathbb{E}_{a_n^{\mathcal{B}} \sim p_\theta(a_n^{\mathcal{B}} | s_n^{\mathcal{B}})} \nabla_\theta \log p_\theta(a_n^{\mathcal{B}} | s_n^{\mathcal{B}}) R(a_n^{\mathcal{B}})$$

Method

where $\boldsymbol{a_n^{\mathcal{B}}}$ is the response to be generated at n-th turn, t indicates token index, $\boldsymbol{s_n^{\mathcal{B}}} = \{\boldsymbol{W^{\mathcal{B}}}, \boldsymbol{h_n^{\mathcal{B}}}\}$, $\boldsymbol{W^{\mathcal{B}}}$ is persona, $\boldsymbol{h_n^{\mathcal{B}}}$ is the dialogue history up to n-th turn, $\boldsymbol{y_n} = \boldsymbol{1}$ is the signal that generated response is predicted as the next utterance,
$\boldsymbol{\gamma}$ is discount factor, $\boldsymbol{r}(\boldsymbol{a_n^{\mathcal{B}}}) = score(a_n^{\mathcal{B}}, W^{\mathcal{B}})$, and λ is hyper-parameter (0.4, 0.1, 0.5).

1. Transmitter

7) Reward Shaping

$$R_1(a_n^{\mathcal{B}}) = \frac{1}{|a_n^{\mathcal{B}}|} \sum_t \log p_{\text{lm}}(a_{n,t}^{\mathcal{B}} \mid a_{n,<t}^{\mathcal{B}})$$ [RS.1 Language Style]

$$R_2(a_n^{\mathcal{B}}) = \log p_\theta(y_n = 1 \mid a_n^{\mathcal{B}}, s_n^{\mathcal{B}})$$ [RS.2 Discourse Coherence]

$$R_3(a_n^{\mathcal{B}}) = r(a_n^{\mathcal{B}}) + \sum_{k=n+1}^{N} \Big(\gamma^{2(k-n)-1} r(x_k^{\mathcal{A}^*}) + \gamma^{2(k-n)} r(a_k^{\mathcal{B}}) \Big),$$ [RS.3 Mutual Persona Perception]

$$R = \lambda_1 R_1 + \lambda_2 R_2 + \lambda_3 R_3$$

Method

2. Receiver

1) Training

- Receiver measures the proximity between the built impressions and the actual personas

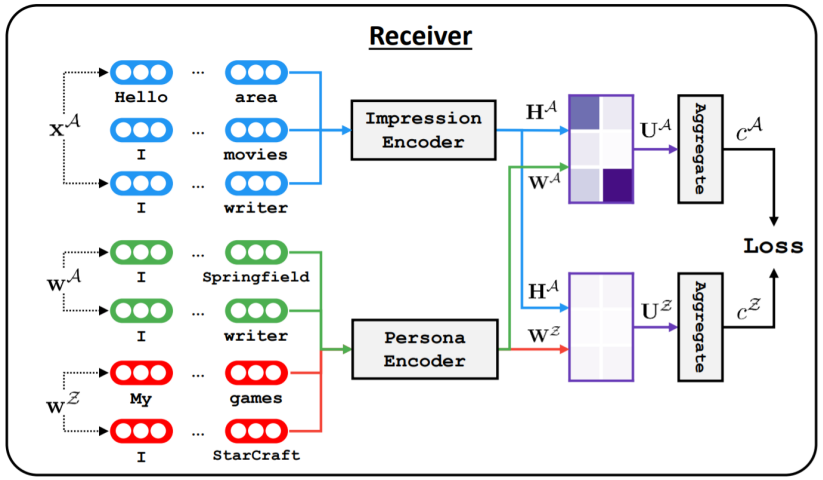
- Receiver is trained to identify the real persona W^A from $\{W^A, W^Z\}$, where W^Z is randomly sampled distractors

$$r(a_n^{\mathcal{B}}) = \text{score}(a_n^{\mathcal{B}}, \mathbf{w}^{\mathcal{B}})$$

Method

2. Receiver

1) Training



$$\mathbf{U}^{\Delta} = \frac{\mathbf{H}^{\mathcal{A}}(\mathbf{W}^{\Delta})^{\top}}{\sqrt{d}}, \in \mathbb{R}^{N \times L}$$

$$, \text{where } \Delta \in \{\mathcal{A}, \mathcal{Z}\}$$

- Initialized by Transformer encoder, BERT (Devlin et al., 2019)

- Training loss:

$$\mathcal{L}_{\text{rec}} = \max(0, m + c^{\mathcal{Z}} - c^{\mathcal{A}}) + \beta \cdot |\mathbf{U}^{\Delta}|_1$$

Method

2. Receiver

1) Training

$$\mathcal{L}_{\text{rec}} = \max(0, m + c^{\mathcal{Z}} - c^{\mathcal{A}}) + \beta \cdot |\mathbf{U}^{\Delta}|_1$$

- c^{Δ} is cumulative score:

$$c^{\Delta} = \frac{1}{N} \sum_{n=1}^{N} Agg(\mathbf{U}^{\Delta}_{n,:})$$

$$Agg(\mathbf{U}^{\Delta}_{n,:}) = \frac{\sum_{k=1}^{L} \exp(\mathbf{U}^{\Delta}_{n,k}/\tau) \cdot \mathbf{U}^{\Delta}_{n,k}}{\sum_{k=1}^{L} \exp(\mathbf{U}^{\Delta}_{n,k}/\tau)} \quad : \tau \; anneals \; 10 \; to \; 0.5$$

where *temperature* $\tau > 0$ is a tunable parameter

Method

2. Receiver

2) Inference

$$r(a_n^{\mathcal{B}}) = \text{score}(a_n^{\mathcal{B}}, \mathbf{w}^{\mathcal{B}})$$

$$\text{score}(x_n^{\mathcal{A}}, \mathbf{w}^{\mathcal{A}}) = \frac{Agg\big(\mathbf{H}_{n,:}^{\mathcal{A}}(\mathbf{W}^{\mathcal{A}})^{\top}\big)}{\sqrt{d}}$$

$$R_3(a_n^{\mathcal{B}}) = r(a_n^{\mathcal{B}}) + \sum_{k=n+1}^{N} \left(\gamma^{2(k-n)-1} r(x_k^{\mathcal{A}^*}) + \gamma^{2(k-n)} r(a_k^{\mathcal{B}}) \right)$$

Outline

1. Contribution
2. Method
3. Experiments
4. Conclusion

Experiments

1. Data

- ConvAI2 (Dinan et al., NeurIPS 2018) aims at finding approaches to creating high quality dialogue agents capable of meaningful open domain conversation.

- (PERSONA-CHAT): 8,939/1,000 multi-turn dialogs and 1,155/100 personas

[PERSON 1:] Hi
[PERSON 2:] Hello ! How are you today ?
[PERSON 1:] I am good thank you , how are you.
[PERSON 2:] Great, thanks ! My children and I were just about to watch Game of Thrones.
[PERSON 1:] Nice ! How old are your children?
[PERSON 2:] I have four that range in age from 10 to 21. You?
[PERSON 1:] I do not have children at the moment.
[PERSON 2:] That just means you get to keep all the popcorn for yourself.
[PERSON 1:] And Cheetos at the moment!
[PERSON 2:] Good choice. Do you watch Game of Thrones?
[PERSON 1:] No, I do not have much time for TV.
[PERSON 2:] I usually spend my time painting: but, I love the show.

Table 1: A clipped conversation from the dataset

Persona 1
I like to ski
My wife does not like me anymore
I have went to Mexico 4 times this year
I hate Mexican food
I like to eat cheetos

Persona 2
I am an artist
I have four children
I recently got a cat
I enjoy walking for exercise
I love watching Game of Thrones

Experiments

2. Metrics

1) Automatic metric

- Hit@1/20
- Perplexity (PPL)
- F1 (word-level precision and recall)
- BLEU

2) Human evaluation

- 1 to 4 (higher is better)

Experiments

3. Results

Category	Model	Original			Revised		
		Hits@1(%) ↑	ppl ↓	F1(%) ↑	Hits@1(%) ↑	ppl ↓	F1(%) ↑
Retrieval	KV Profile Memory	54.8	-	14.25	38.1	-	13.65
	Dually Interactive Matching	78.8	-	-	**70.7**	-	-
Generative	Generative Profile Memory	10.2	35.01	16.29	9.9	34.94	15.71
	Language Model	-	50.67	16.30	-	51.61	13.59
	SEQ2SEQ-ATTN	12.5	35.07	16.82	9.8	39.54	15.52
Pretrain Fintune	Lost In Conversation	17.3	-	17.79	16.2	-	16.83
	Transfertransfo	**82.1**	17.51	19.09	-	-	-
	$\mathcal{P}^2$ BOT (Our)	81.9 [0.1]	**15.12** [0.16]	**19.77** [0.08]	68.6 [0.2]	**18.89** [0.11]	**19.08** [0.07]

Table 1: Automatic evaluation results of different methods on the PERSONA-CHAT dataset. The standard deviation [σ] (across 5 runs) of $\mathcal{P}^2$ BOT is also reported. All the results were evaluated on the dev set since the test set was not publicly available.

Experiments

3. Results

Model	1 (%)	2 (%)	3 (%)	4 (%)	Avg
Lost In Conversation	26.3	**48.7**	22.0	3.0	2.017
Transfertransfo	**41.7**	25.3	**28.7**	4.3	1.956
$\mathcal{P}^2$ BOT (Our)	18.9	26.3	28.6	**26.2**	**2.621**

Table 2: Human evaluation results.

Experiments

3. Results

Variant	Hits@1(%)↑	F1(%)↑	BLEU(%)↑
$\mathcal{P}^2$ BOT-S	68.7	18.14	0.56
- Persona	65.5	17.77 (-2.0%)	0.57 (+ 1.8%)
- Next	17.6	18.11 (-0.1%)	0.55 (- 1.8%)
+ RS.1	68.4	18.32 (+0.9%)	0.60 (+ 7.1%)
↪ + RS.2	68.6	18.41 (+1.5%)	0.61 (+ 8.9%)
↪ + RS.3	68.6	19.08 (+5.2%)	0.75 (+33.9%)

Table 3: Variant analysis results on PERSONA-CHAT revised mode, along with relative improvements (shown inside brackets) compared with $\mathcal{P}^2$ BOT-S. BLEU refers to the cumulative 4-gram BLEU score. "- Persona" means dialogue generation without personas; "- Next" ablates the auxiliary task mentioned in Section 3.1; "+ RS.1" means only using Language Style score as the reward in the self-play fine-tuning phase; "↪ + RS.2" means adding Discourse Coherence to the reward on the basis of RS.1; "↪ + RS.3" is equivalent to our proposed $\mathcal{P}^2$ BOT.

Outline

1. Contribution
2. Method
3. Experiments
4. Conclusion

Conclusion

1. Contribution

1) Response Blandness
2) Speaker Consistency
3) Word Repetition
4) Lack of Grounding
5) Evaluation metric
6) Understanding conversation between interlocutors
7) Lack of maintaining a consistent personality
8) Lack of an explicit long-term memory
9) Too much questions lead to boring and annoying conversations

Conclusion

2. Any problems?

1) Reward Shaping

$$R_1(a_n^{\mathcal{B}}) = \frac{1}{|a_n^{\mathcal{B}}|} \sum_t \log p_{\text{lm}}(a_{n,t}^{\mathcal{B}} \mid a_{n,<t}^{\mathcal{B}})$$

[RS.1 Language Style]

$$R_2(a_n^{\mathcal{B}}) = \log p_\theta(y_n = 1 \mid a_n^{\mathcal{B}}, s_n^{\mathcal{B}})$$

[RS.2 Discourse Coherence]

$$R_3(a_n^{\mathcal{B}}) = r(a_n^{\mathcal{B}}) + \sum_{k=n+1}^{N} \Big(\gamma^{2(k-n)-1} r(x_k^{\mathcal{A}^*}) + \gamma^{2(k-n)} r(a_k^{\mathcal{B}}) \Big),$$

[RS.3 Mutual Persona Perception]

These hand-crafted reward fail to consider all the variables

Conclusion

Team Names	Perplexity	Hits@1	F1
1. Hugging Face	16.28	80.7	19.5
2. ADAPT Centre	31.4	-	18.39
3. Happy Minions	29.01	-	16.01
4. High Five	-	65.9	-
5. Mohd Shadab Alam	29.94	13.8	16.91
6. Lost in Conversation	-	17.1	17.77
7. Little Baby	-	64.8	-
8. Sweet Fish	-	45.7	-
9. 1st-contact	31.98	13.2	16.42
10. NEUROBOTICS	35.47	-	16.68
11. Cats'team	-	35.9	-
12. Sonic	33.46	-	16.67
13. Pinta	32.49	-	16.39
14. Khai Mai Alt	-	34.6	13.03
15. loopAI	-	25.6	-
16. Salty Fish	34.32	-	-
17. Team Pat	-	-	16.11
18. Tensorborne	38.24	12.0	15.94
19. Team Dialog 6	40.35	10.9	7.27
20. Roboy	-	-	15.83
21. IamNotAdele	66.47	-	13.09
22. flooders	-	-	15.47
23. Clova Xiaodong Gu	-	-	14.37
Seq2Seq + Attention Baseline	29.8	12.6	16.18
Language Model Baseline	46.0	-	15.02
KV Profile Memory Baseline	-	55.2	11.9

Table 3: Automatic Metrics Leaderboard.

2. Any problems?

2) Metrics

- Word overlap metrics don't correlate well with human judgement

e.g.,) F1 measure. Always replying "I am you to do and your is like" scores 18 to 20.

- Current metric fails to measure the multi-turn aspects (w.r.t repetition, consistency, and balance of dialog act)

- Heavily rely on Human Evaluation (Absence of solid evaluation criteria)

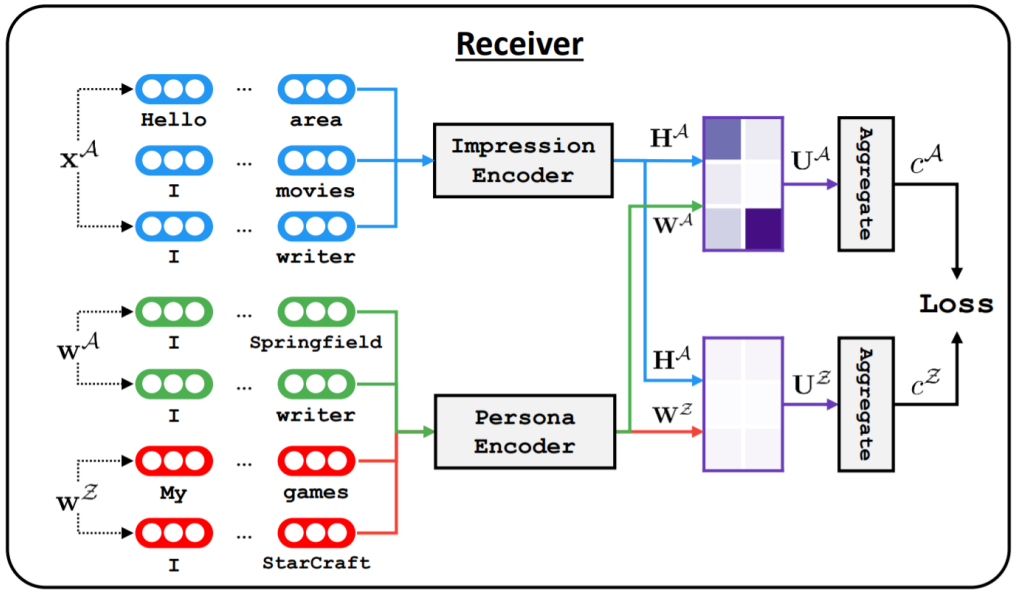
- Optimizing only one of metrics can fail to address important issues.

Conclusion

2. Any problems?

3) Improvements?

- Inverse Reinforcement Learning (IRL)
- Shaped reward -> evaluation metric for conversation model
- Much meaningful encoder



Thank you

https://jeiyoon.github.io/

Jeiyoon Park

I'm a master's student in the Department of Computer Science and Engineering at Korea University, advised by the professor Heuiseok Lim.

My research interests are Dialog systems, Recommendation systems, Meta-learning, and Reinforcement learning.

Email / Github / Google Scholar / LinkedIn / YouTube / Posts

Appendix A

1. Open-domain dialog system

1) Response Blandness: Utterance generated by neural response generation systems are often bland and deflective (uninformative response).

e.g.,) I don't know or I'm Ok

2) Speaker Consistency: Due to training data itself. Conversational datasets feature multiple speakers, which often have difference or conflicting personas and backgrounds.

e.g.,) speaking styles (e.g., British English), topic (e.g., sports), …

Appendix A

1. Open-domain dialog system

3) Word Repetition: It is clear the humans repeat themselves very infrequently

e.g.,) “i like watching horror” followed by “i love watching scary movies”.

4) Lack of Grounding: Previous off-the-shelf models often struggle when it comes to generating names and facts that connect to the real world, due to the lack of grounding.

- Persona
- Speaker
- Addressee
- Textual knowledge sources
- User or Agent's virtual environment
- Emotion of the user

Appendix A

2. Persona-based Conversation Model ⊂ Open Domain Dialog

- Basically, it contains

1) Response Blandness: Utterance generated by neural response generation systems are often bland and deflective (uninformative response).

2) Speaker Consistency: Due to training data itself. Conversational datasets feature multiple speakers, which often have difference or conflicting personas and backgrounds.

3) Word Repetition: It is clear the humans repeat themselves very infrequently

4) Lack of Grounding: Previous off-the-shelf models often struggle when it comes to generating names and facts that connect to the real world, due to the lack of grounding.

Appendix A

2. Persona-based Conversation Model

- Plus, the following challenges:

1) Lack of maintaining a consistent personality

2) Lack of an explicit long-term memory
(It excessively depends on previous dialog history)

3) Too much questions lead to boring and annoying conversations
(e.g., TransferTransfo, Wolf et al., AAAI 2019, Hugging Face)

4) Metrics!!!

Appendix B

Any problem?: Reward shaping & evaluation metric

1) Reward Shaping

- e.g.,)

1) $p(Dull\ Response \mid T_i)$: short response such as “I don’t know”
2) $\log Sigmoid\ [\cos(T_{i-1}, T_i)]$: ensuring a consecutive turns are similar to each other
3) $\log p(T_{i-1}|T_i) + \log p(T_i|T_{i-1})$: encouraging consecutive turns in a dialog session to be related to each other (without this, the model changes topics so frequently)

- Approach:

1) Additional reward engineering
2) Inverse Reinforcement Learning (e.g., Boltzmann distribution)