Parameter-Efficient Conversational Recommender System as a Language Processing Task

Mathieu Ravaut, Hao Zhang, Lu Xu, Aixin Sun, Yong Liu

mathieuj001@e.ntu.edu.sg

Conversational Recommendation through Language Models

Conversational Recommender Systems (**CRS**) are designed to jointly tackle two sub-tasks: 1) generating natural language responses to interact with the user (**conversation**); and 2) recommending desirable items to user based on dialogue context (**recommendation**). CRS can be classified into *attribute-based* methods: collecting use preference on items attributed to narrow down the item recommendation space ; and *generation-based* methods: acquiring feedback from users through natural language exchanges. We focus on generation-based CRS.

Jointly modelling language generation and item recommendation is not straightforward. Prior work typically use a knowledge graph (**KG**) containing items semantics and a graph neural network (GNN) to learn items representations. Language generation is learned through a language model (**LM**). The GNN and LM being optimized independently, the whole system suffers from inconsistency between items representations and words representations.

Attempts to fix this inconsistency, like UNICRS, require training for multiple stages to unify both semantic spaces. Recent work MESE bypasses the need for a KG, but still fine-tunes several pre-trained LMs (two DistilBERT and a GPT-2).

In this work, ***we propose the first truly unified CRS: fine-tuning a single LM (GPT-2) through a single training stage.***

Model

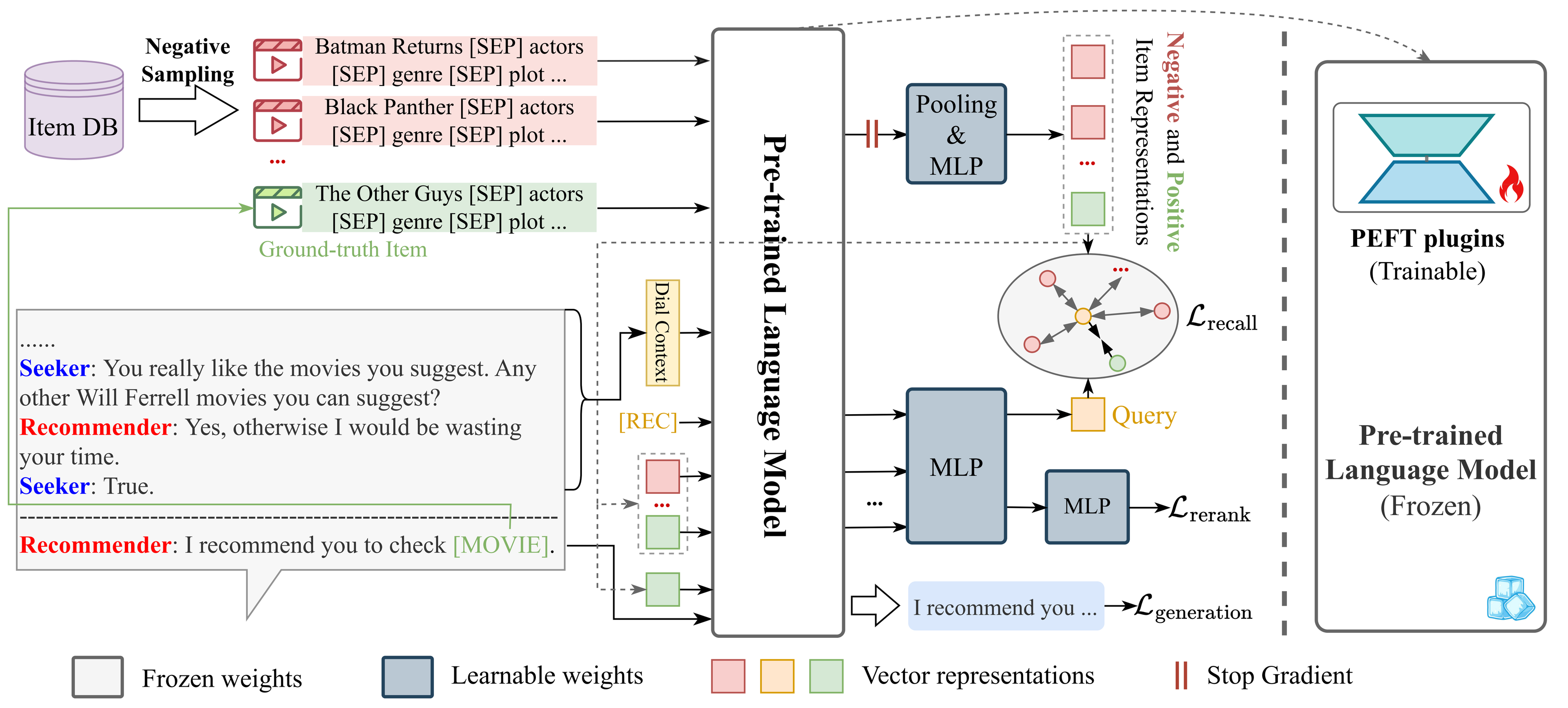

Fig 1. Overall architecture of our system PECRS.

Each movie item is associated with a **textual description** in the template:
Movie Title [SEP] Actors [SEP] Director(s) [SEP] Genre [SEP] Plot
We use the LM to encode each movie, through an item head and pooling layer:

$$\boldsymbol{v}_j = h_{\text{item}}(\boldsymbol{w}^T \cdot \boldsymbol{I}_j).$$

The LM is enhanced with special tokens: [ITEM], [SEP], [REC], [REC_END].

Training

The dialogue context is represented by the [REC] token appended to the concatenation of utterances: $\boldsymbol{D}_t = [\bar{\boldsymbol{u}}_1, \dots, \bar{\boldsymbol{u}}_{t-1}, \boldsymbol{v}_{\text{rec}}]$,

For each context, we randomly sample M negative items.
We use an **NCE approach** to bring closer the dialogue context representation $\boldsymbol{d}_t$ and the positive item v_p, where f is an MLP:

$$\mathcal{E}_{D_t} = \frac{e^{f(\boldsymbol{d}_t)^\top \odot \boldsymbol{v}_p}}{e^{f(\boldsymbol{d}_t)^\top \odot \boldsymbol{v}_p} + \sum_{(\boldsymbol{d}_t, \boldsymbol{v}'_j) \sim \boldsymbol{\mathcal{N}}} e^{f(\boldsymbol{d}_t)^\top \odot \boldsymbol{v}'_j}},$$

The recall loss learns to **retrieve the positive item**:

$$\mathcal{L}_{\text{recall}} = -\frac{1}{|\boldsymbol{\mathcal{D}}|} \sum_{D_t \in \boldsymbol{\mathcal{D}}} \log(\mathcal{E}_{D_t}).$$

We then **re-rank** items (following MESE) with a score predicting if the item is the positive one, conditioning on the context + item representations:

$$\mathcal{L}_{\text{rerank}} = \frac{1}{|\boldsymbol{\mathcal{D}}|} \sum_{D_t \in \boldsymbol{\mathcal{D}}} f_{\text{XE}}(\boldsymbol{r}, \boldsymbol{Y}),$$

The dialogue generation is trained with a standard **next-token prediction**:

$$\mathcal{L}_{\text{gen}} = -\frac{1}{|\boldsymbol{\mathcal{D}}|} \sum_{D_t \in \boldsymbol{\mathcal{D}}} \frac{1}{n} \sum_{j=1}^{n} \log(p_\theta(w_j | w_{1:(j-1)}, \tilde{\boldsymbol{D}}_t)).$$

The final loss is a linear combination of all 3 losses:

$$\mathcal{L} = \alpha \times \mathcal{L}_{\text{recall}} + \beta \times \mathcal{L}_{\text{rerank}} + \gamma \times \mathcal{L}_{\text{gen}},$$

We optimize the LM layers through **LoRA modules.**

During training, we append the ground truth item for response generation.
We **share negative items for recall and re-rank losses**. At inference, we append the top re-ranked item to the context to prompt response generation.

Experimental Setup

- We experiment on ReDial and INSPIRED datasets.
- We train with AdamW and learning rate 3.10^{-5}, warming up one epoch.
- M_{train} is set to 150, $M_{\text{inference}}$ to 700.
- Losses are balanced with $\alpha = 0.15$, $\beta = 0.85$ and $\gamma = 1.00$.

Recommendation Results

Model	Metadata			Model Properties			ReDial				INSPIRED			
	KG	Reviews	Description	Extra Model	PEFT	Rounds	R@1	R@10	R@50	Unique	R@1	R@10	R@50	Unique
ReDial (Li et al., 2018)	✗	✗	✗	✓	✗	3	2.4	14.0	32.0	_	_	_	_	_
KBRD (Chen et al., 2019)	✓	✗	✗	✓	✗	2	3.0	16.3	33.8	_	_	_	_	_
KGSF (Zhou et al., 2020a)	✓	✗	✗	✓	✗	3	3.9	18.3	37.8	_	_	_	_	_
KECRS (Zhang et al., 2022)	✓	✗	✗	✓	✗	2	2.3	15.7	36.6	_	_	_	_	_
BARCOR (Wang et al., 2022b)	✓	✗	✗	✓	✗	1	2.5	16.2	35.0	_	_	_	_	_
UniCRS (Wang et al., 2022c)	✓	✗	✗	✓	✓	3	5.1	22.4	42.8	_	**9.4**	**25.0**	**41.0**	_
RecInDial (Wang et al., 2022a)	✓	✗	✗	✓	✗	1	3.1	14.0	27.0	_	_	_	_	_
VRICR (Zhang et al., 2023b)	✓	✗	✗	✓	✗	3	5.7	25.1	41.6	_	_	_	_	_
RevCore (Lu et al., 2021)	✓	✓	✗	✓	✗	2	**6.1**	23.6	45.4	_	_	_	_	_
C^2-CRS (Zhou et al., 2022)	✓	✓	✗	✓	✗	2	5.3	23.3	40.7	_	_	_	_	_
MESE (Yang et al., 2022)	✗	✗	✓	✓	✗	1	5.6	**25.6**	**45.5**	_	4.8	13.5	30.1	_
PECRS-small	✗	✗	✓	✗	✓	1	4.7	20.8	40.5	463	5.4	16.1	33.3	34
PECRS-medium	✗	✗	✓	✗	✓	1	5.8	22.5	41.6	**634**	5.7	17.9	33.7	**72**

Table 1. Recommendation results on ReDial and INSPIRED.

Model	Time/ batch (s)	Rec.		Conv.	
		R@50	Unique	RG-1	Dist@2
PECRS-small	6.1	40.5	463	36.28	0.745
w/o Recall loss	6.1	19.3	21	**37.67**	0.678
w/o Rerank loss	6.1	12.2	87	36.50	0.745
w/o Generation loss	6.1	39.2	451	7.76	**11.907**
w/o Neg. sharing (batch)	8.6	39.8	291	36.40	0.747
w/o Neg. sharing (tasks)	9.1	**40.8**	434	35.98	0.727
w/o Item pooling	6.1	39.6	**530**	36.60	0.748
w/o Item head	6.1	37.9	453	36.33	0.726
w/o Metadata (just title)	**4.2**	35.8	384	36.38	0.765

Table 2. Ablation on ReDial.

Removed	None	Title	Actor(s)	Director(s)	Genre(s)	Plot
R@50	**33.3**	29.8	26.9	32.5	30.5	20.7

Table 3. Ablation on the textual description fields on INSPIRED with PECRS-small.

PECRS-medium reaches comparable Recall to SOTA fine-tuned LMs.

Conversation Results

Model	Reference-based			Reference-free			
	RG-1	RG-2	F-1	PPL	Dist@2	Dist@3	Dist@4
C^2-CRS	_	_	_	_	0.163	0.291	0.417
UniCRS	_	_	_	_	0.492	0.648	0.832
RecInDial	_	_	_	_	0.518	0.624	0.598
MESE	_	_	_	12.9	**0.822**	1.152	1.313
PECRS-small	36.28	14.77	86.04	9.89	0.745	1.462	2.132
PECRS-medium	**36.86**	**15.27**	**86.36**	**8.98**	0.820	**1.552**	**2.154**

Table 4. Conversation results on ReDial.

Aspect	MESE	PECRS-small	Tie
Fluency	28.00 (1.63)	**46.67** (5.91)	25.33 (6.24)
Relevancy	26.33 (2.62)	**46.00** (0.82)	27.67 (2.87)

Table 5. Human evaluation of generated responses on ReDial, with 3 human volunteers.

PECRS-medium reaches SOTA conversation performance.

Analysis

Model	Rec.				Conv.	
	R@1	R@10	R@50	Unique	RG-1	RG-2
PECRS-small	5.4	**16.1**	**33.3**	**34**	**29.72**	**8.26**
Llama-2-7B-chat	**9.3**	*9.3*	*9.3*	26	19.88	2.88
Vicuna-1.5-7B	8.2	*8.2*	*8.2*	23	21.18	3.50

Table 6. Comparison to zero-shot instruction-tuned LLMs on INSPIRED.

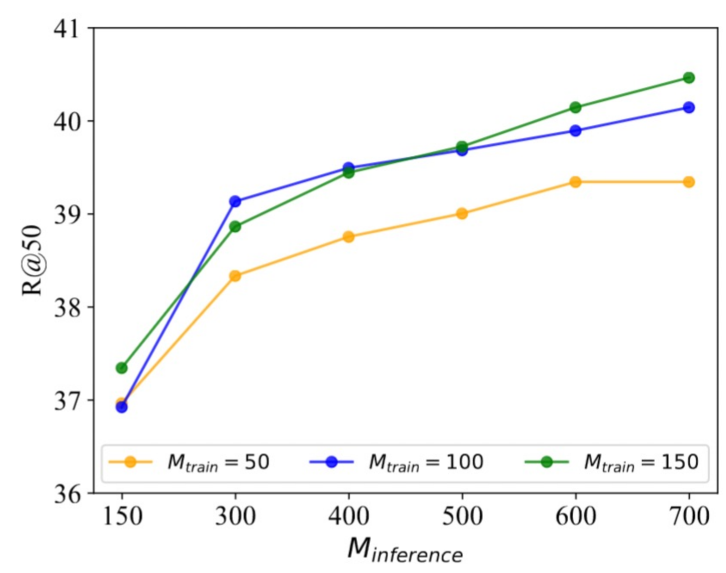

Fig 2. Influence of the M parameter (number of negatives). We decouple M between training and inference. Higher M value performs better.

Decoding Strategy	Reference-based		Reference-free		
	RG-1	RG-2	Dist@2	Dist@3	Dist@4
Greedy decoding	38.54	16.25	0.208	0.311	0.390
Beam search	38.23	16.83	0.235	0.353	0.444
Diverse beam search (diversity=0.5)	39.94	17.30	0.190	0.287	0.361
Diverse beam search (diversity=1.0)	**40.29**	**17.40**	0.179	0.264	0.320
Diverse beam search (diversity=1.5)	40.07	17.23	0.172	0.246	0.290
Top-k sampling (k=25)	33.54	14.40	0.593	1.177	1.806
Top-k sampling (k=50)	33.37	14.17	**0.647**	1.300	1.989
Top-k sampling (k=75)	33.48	14.15	0.644	**1.303**	**1.992**
Nucleus sampling (p=0.90)	36.35	16.04	0.329	0.555	0.760
Nucleus sampling (p=0.95)	36.44	16.02	0.351	0.594	0.804
Nucleus sampling (p=0.99)	36.60	16.07	0.352	0.593	0.809

Table 7. Conversation performance on ReDial for several decoding strategies.

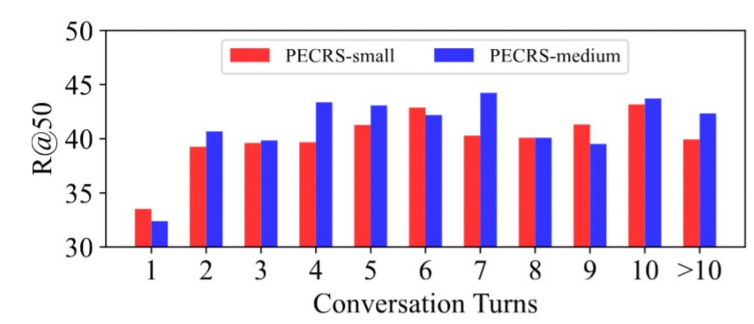

Fig 3. Performance per number of conversation turns in the context.

We advocate for reference-based conversation evaluation (Table 7).