HeterMPC: A Heterogeneous Graph Neural Network for Response Generation in Multi-Party Conversations

Jia-Chen Gu[1*†], Chao-Hong Tan[1†], Chongyang Tao[2], Zhen-Hua Ling[1], Huang Hu[2], Xiubo Geng[2], Daxin Jiang[2‡]

[1]National Engineering Research Center for Speech and Language Information Processing, University of Science and Technology of China, Hefei, China

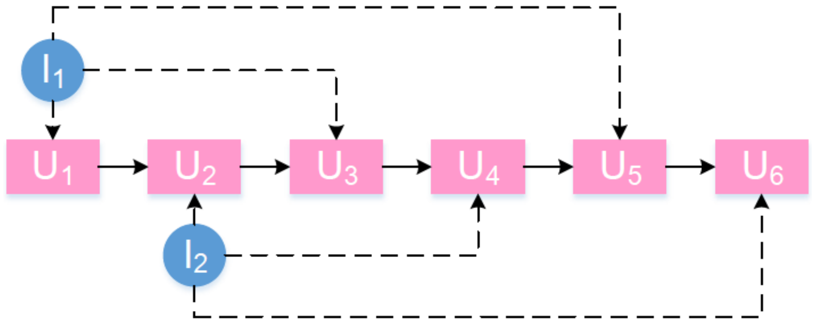
[2]Microsoft, Beijing, China

*Work done during the internship at Microsoft. †Equal contribution. ‡Corresponding author.

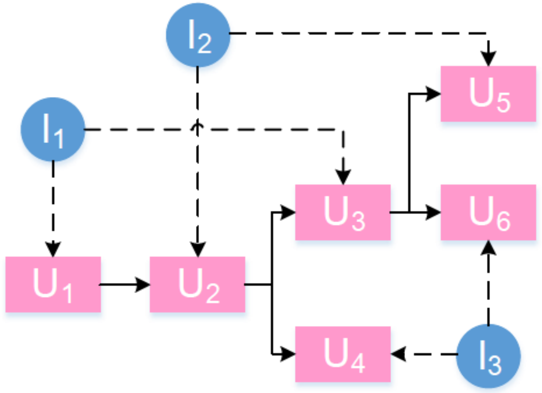
Outline

- **Introduction**
- HeterMPC
- Experiments
- Conclusion

Introduction



Utterances in a two-party conversation are posted one by one between two interlocutors, constituting a sequential information flow.



Utterances in a multi-party conversation can be spoken by anyone and address anyone else, constituting a graphical information flow.

Related Work

- Model a conversation with a homogeneous graph, where nodes represented only utterances while interlocutors are ignored.
- The same model structure and parameters are employed for both the forward and backward flows of a bidirectional message passing algorithm, which cannot distinguish the “reply" or “replied-by" relations between two connected utterance nodes.
- Information flows along both directions are independently propagated, so that a graph node cannot be jointly updated at a single propagation step.

Outline

- Introduction
- **HeterMPC**
- Experiments
- Conclusion

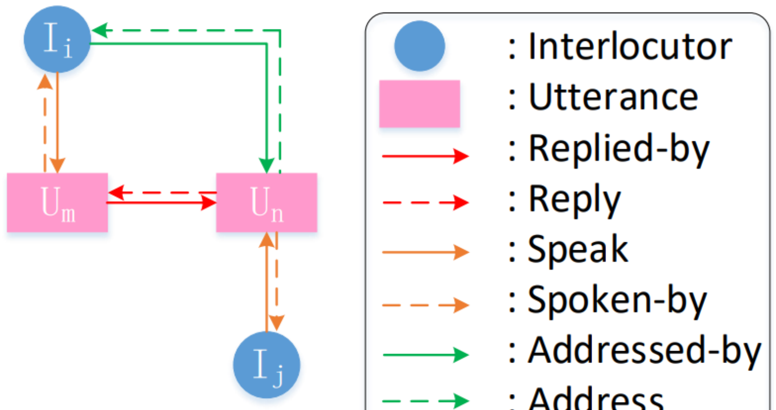
Overview

- Utterances and interlocutors are considered as two types of nodes under a unified heterogeneous graph, to explicitly model the complicated interactions between interlocutors, between utterances, and between an interlocutor and an utterance.

Graph Construction

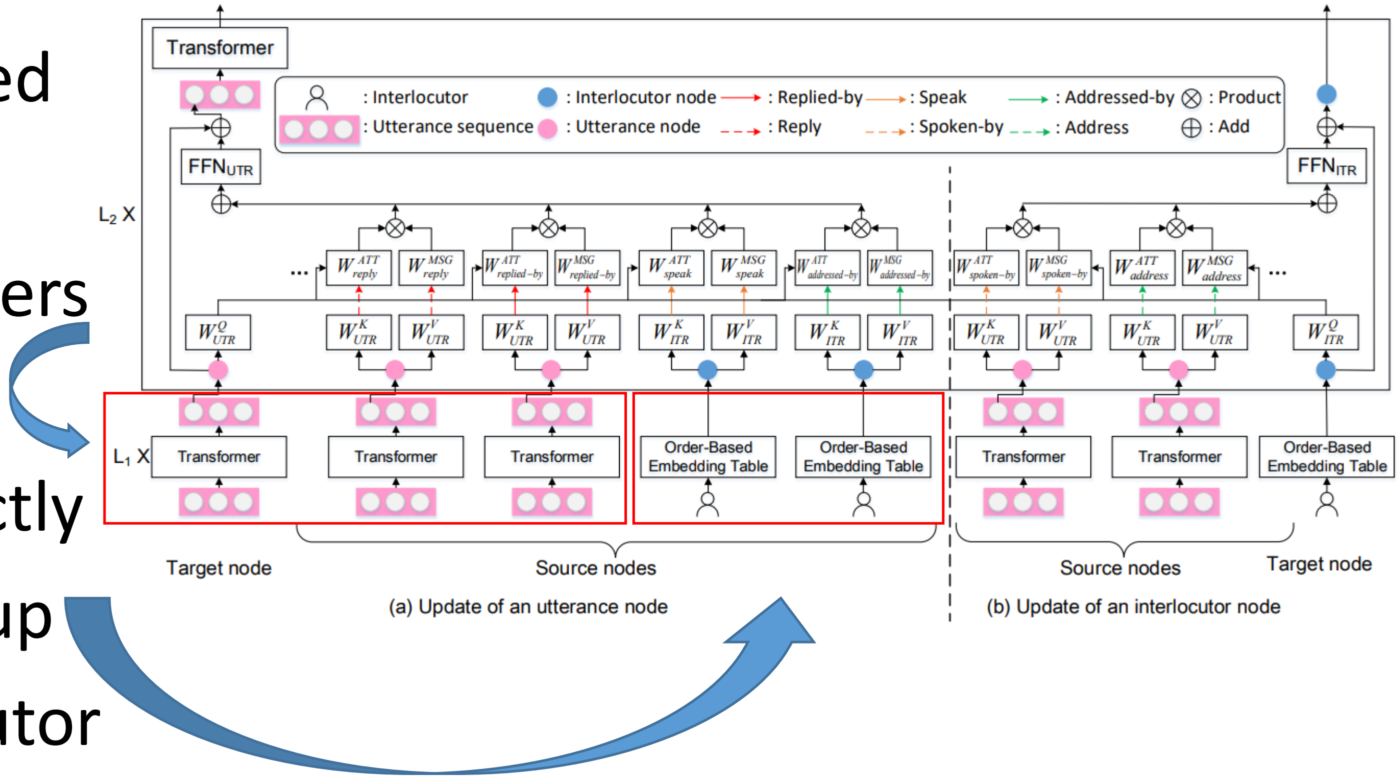
- *M* utterances and *I* interlocutors → a heterogeneous graph *G(V, E)*
- *V* : a set of *M + I* nodes, each denoting an utterance or an interlocutor
- $E = \{e_{p,q}\}_{p,q=1}^{M+I}$: a set of directed edges, each edge $e_{p,q}$ describing the connection from node *p* to node *q*

- Six types of meta relations: {*reply, replied-by, speak, spoken-by, address, addressed-by*} to describe directed edges between two nodes



Node Initialization

- Each utterance is encoded individually by stacked Transformer encoder layers
- Each interlocutor is directly represented by looking up an order-based interlocutor embedding table



Node Updating

- Introduce parameters to model heterogeneity
- Attention weights

$$\boldsymbol{k}^l(s) = \boldsymbol{h}_s^l \boldsymbol{W}_{\tau(s)}^K + \boldsymbol{b}_{\tau(s)}^K,$$

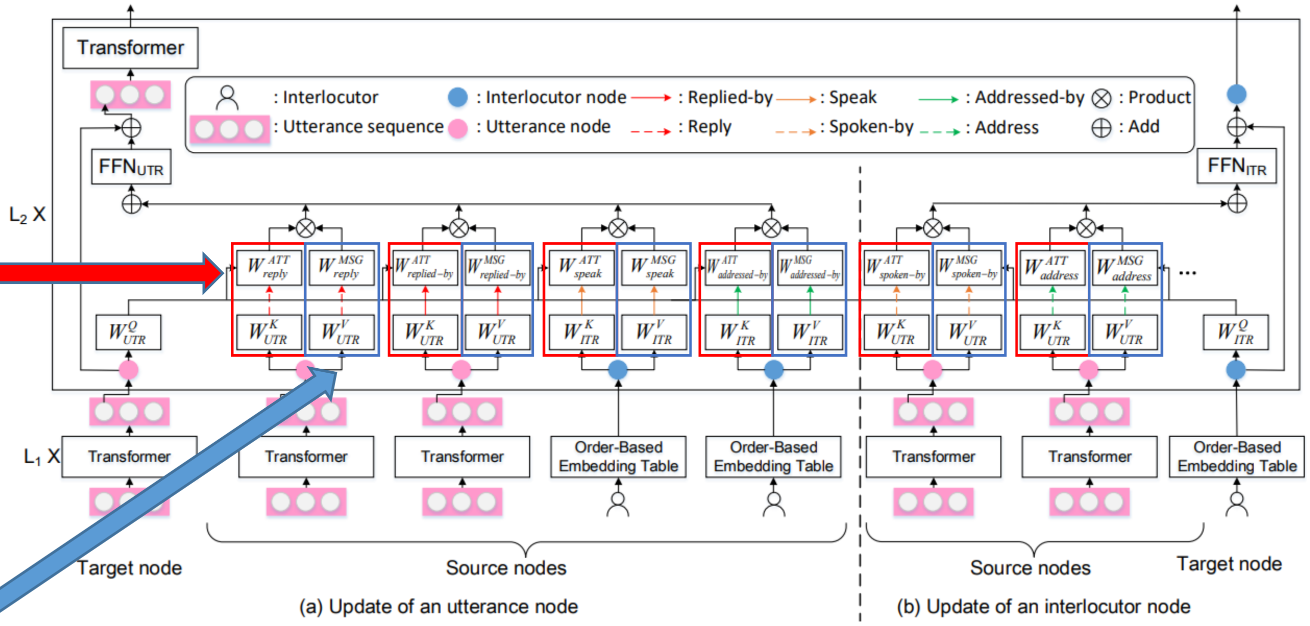
$$\boldsymbol{q}^l(t) = \boldsymbol{h}_t^l \boldsymbol{W}_{\tau(t)}^Q + \boldsymbol{b}_{\tau(t)}^Q,$$

$$w^l(s,e,t) = \boldsymbol{k}^l(s) \boldsymbol{W}_{e_{s,t}}^{ATT} \boldsymbol{q}^l(t)^T \frac{\mu_{e_{s,t}}}{\sqrt{d}}.$$

- Message passing

$$\boldsymbol{v}^l(s) = \boldsymbol{h}_s^l \boldsymbol{W}_{\tau(s)}^V + \boldsymbol{b}_{\tau(s)}^V,$$

$$\bar{\boldsymbol{v}}^l(s) = \boldsymbol{v}^l(s) \boldsymbol{W}_{e_{s,t}}^{MSG},$$



(s, e, t) denotes (source, edge, target)

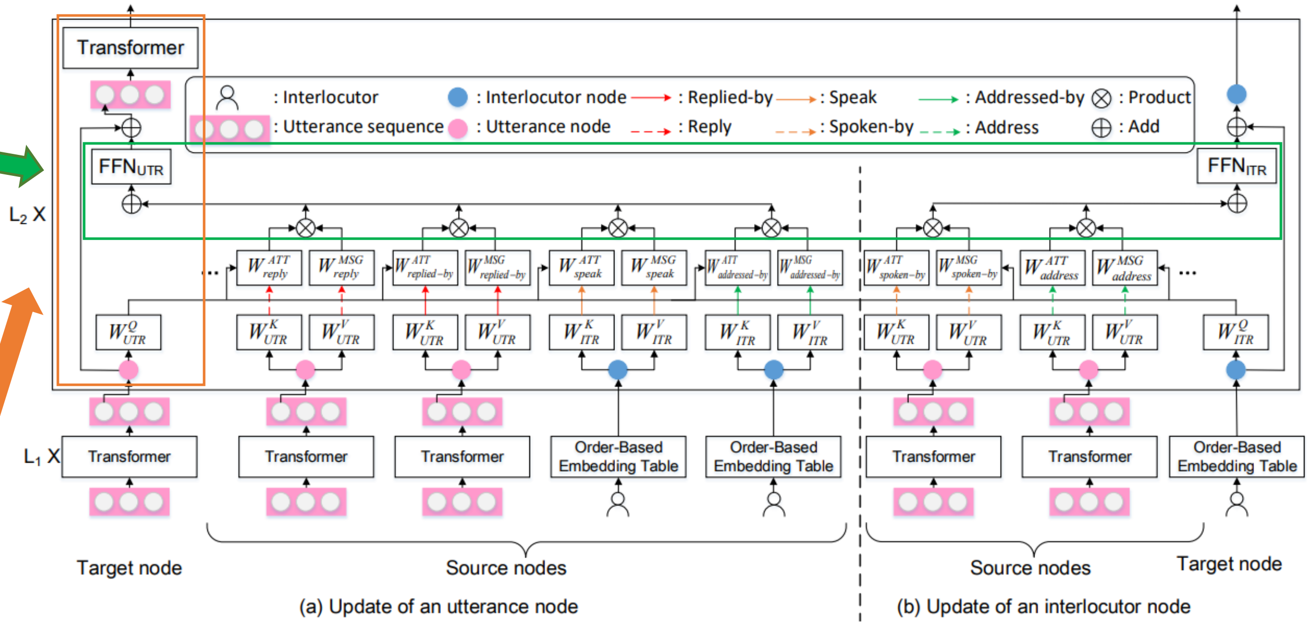
τ(s), τ(t) ∈ {utterance, interlocutor}

Node Updating

- Aggregation

$$\bar{\boldsymbol{h}}_t^l = \sum_{s \in S(t)} \mathrm{softmax}(w^l(s,e,t))\bar{\boldsymbol{v}}^l(s),$$

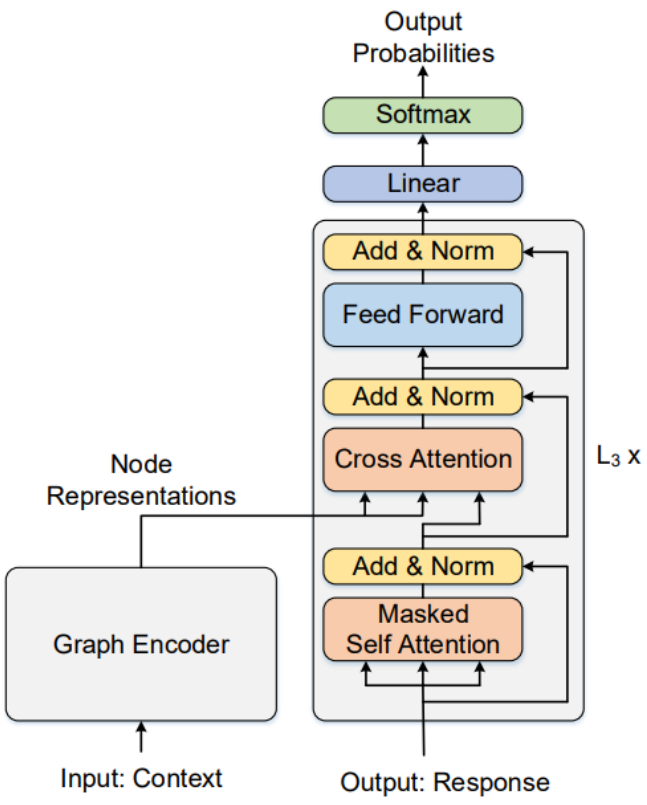
$$\boldsymbol{h}_t^{l+1} = FFN_{\tau(t)}(\bar{\boldsymbol{h}}_t^l) + \boldsymbol{h}_t^l,$$



- Specifically, the context information in an utterance node is shared with other tokens in the utterance through another round of Transformer layer intra-utterance self-attention.

Decoder

- Standard implementation of Transformer decoder
- A cross-attention operation over the node representations of the graph encoder output is performed to incorporate graph information



Outline

- Introduction
- HeterMPC
- **Experiments**
- Conclusion

Setup

- Dataset

  Ubuntu IRC benchmark released by Hu et al., 2019

- Baselines

  RNN-based Seq2Seq, Transformer, GPT-2, BERT, GSN and BART

- Metrics

  Automated: BLEU1 to BLEU-4, METEOR and ROUGEL

  Human: relevance, fluency and informativeness

Results

- BERT or BART was selected to initialize the utterance encoder layers of HeterMPC

Models \ Metrics	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE$_L$
Seq2Seq (LSTM) (Sutskever et al., 2014)	7.71	2.46	1.12	0.64	3.33	8.68
Transformer (Vaswani et al., 2017)	7.89	2.75	1.34	0.74	3.81	9.20
GSN (Hu et al., 2019b)	10.23	3.57	1.70	0.97	4.10	9.91
GPT-2 (Radford et al., 2019)	10.37	3.60	1.66	0.93	4.01	9.53
BERT (Devlin et al., 2019)	10.90	3.85	1.69	0.89	4.18	9.80
HeterMPC$_{BERT}$	**12.61**	**4.55**	**2.25**	**1.41**	**4.79**	**11.20**
HeterMPC$_{BERT}$ w/o. node types	11.76	4.09	1.87	1.12	4.50	10.73
HeterMPC$_{BERT}$ w/o. edge types	12.02	4.27	2.10	1.30	4.74	10.92
HeterMPC$_{BERT}$ w/o. node and edge types	11.60	3.98	1.90	1.18	4.20	10.63
HeterMPC$_{BERT}$ w/o. interlocutor nodes	11.80	3.96	1.75	1.00	4.31	10.53
BART (Lewis et al., 2020)	11.25	4.02	1.78	0.95	4.46	9.90
HeterMPC$_{BART}$	**12.26**	**4.80**	**2.42**	**1.49**	**4.94**	**11.20**
HeterMPC$_{BART}$ w/o. node types	11.22	4.06	1.87	1.04	4.57	10.63
HeterMPC$_{BART}$ w/o. edge types	11.52	4.27	2.05	1.24	4.78	10.90
HeterMPC$_{BART}$ w/o. node and edge types	10.90	3.90	1.79	1.01	4.52	10.79
HeterMPC$_{BART}$ w/o. interlocutor nodes	11.68	4.24	1.91	1.03	4.79	10.65

Table 1: Performance of HeterMPC and ablations on the test set in terms of automated evaluation. Numbers in bold denote that the improvement over the best performing baseline is statistically significant (t-test with p-value < 0.05).

Models \ Metrics	Score	Kappa
Human	2.81	0.55
GSN (Hu et al., 2019b)	2.00	0.50
BERT (Devlin et al., 2019)	2.19	0.42
BART (Lewis et al., 2020)	2.24	0.44
HeterMPC$_{BERT}$	2.39	0.50
HeterMPC$_{BART}$	2.41	0.45

Table 2: Human evaluation results of HeterMPC and some selected systems on a randomly sampled test set.

Analysis

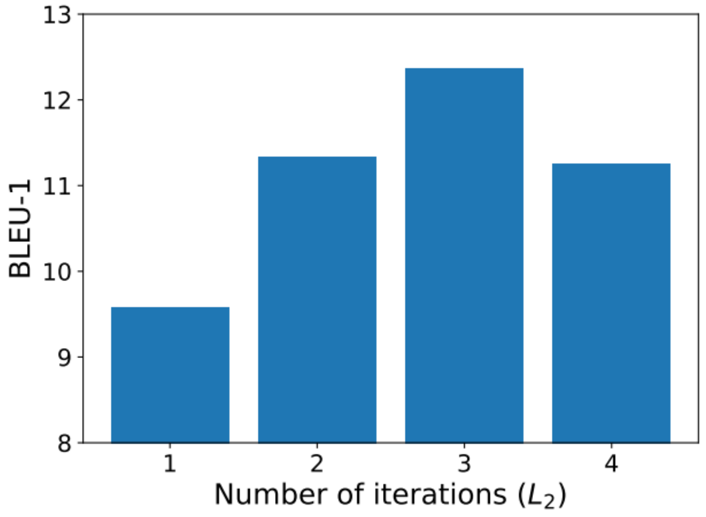


Figure 5: Performance of HeterMPC$_{BERT}$ under different numbers of iterations (L_2) on the test set.

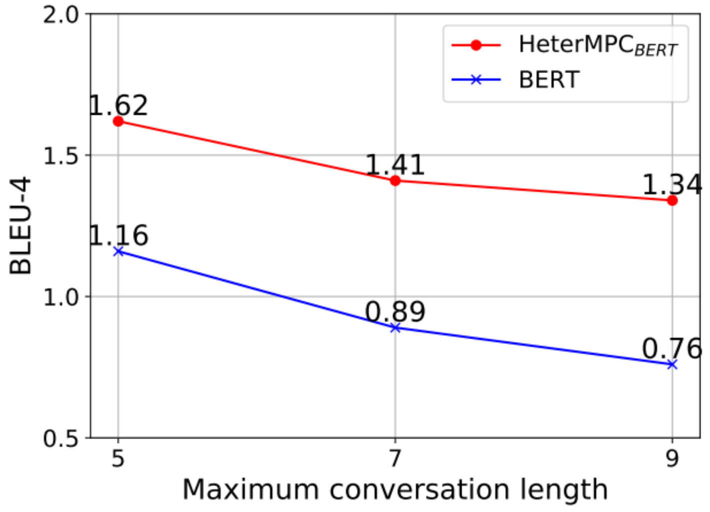


Figure 6: Performance of BERT and HeterMPC$_{BERT}$ on test samples with different session lengths.

The performance of was significantly improved as L2 increased at the beginning. Then, the performance was stable and dropped slightly.

As session lengths increased, the performance dropped less than that of BERT, showing superiority on dealing with longer conversations.

Case Study

- "*gparted*" in the first utterance is two hops away from I.2 (the first utterance is replied by the second utterance which is spoken by I.2), and "*install gparted*" in the third utterance is one hop away from I.2 (this utterance directly addresses I.2).
- These keywords are included in both responses generated by $\text{HeterMPC}_{\text{BERT}}$ and $\text{HeterMPC}_{\text{BART}}$, but are missing in those generated by other models.

Speaker	*Utterance*	*Addressee*
	Case 1	
I.1	geev: in gparted now?	-
I.2	there is no such command in my computer	I.1
I.1	open a terminal, type: sudo apt-get install gparted	I.2
I.1	after: gparted aksks for your pw, opens a gui	I.2
I.2	ok if found fail i will contact you	I.1
I.3	does it say "to install type sudo apt-get install gparted"? **(Human)**	I.2
	do you have remote access to remote desktop? **(GSN)**	
	ok, i will try that. **(BERT)**	
	you can use the command "sudoapt-get install gparted" to install the gparted livecd. **($\text{HeterMPC}_{\textit{BERT}}$)**	
	i'm not sure what you mean by "in gparted" **(BART)**	
	you need to install gparted, it is a gui tool **($\text{HeterMPC}_{\textit{BART}}$)**	

Robustness

- The addressee labels are important for constructing a graph used in HeterMPC. This kind of label is commonly available in real life such as "A@B" labels in group chatting, Twitter, Reddit and various forums that denote speaker A talking to addressee B.

- Addressee labels of a part of utterances are missing in the existing MPC datasets since a speaker may forget to specify an addressee.

- Utterances without addressee labels can be assigned with a general addressee label "*To all interlocutors*". We leave evaluation on other datasets in future work.

Outline

- Introduction
- HeterMPC
- Experiments
- **Conclusion**

Conclusion

- We present HeterMPC to model complicated interactions between utterances and interlocutors in MPCs with a heterogeneous graph.
- Two types of graph nodes and six types of edges are designed for better utilizing the structural knowledge of conversations during node updating.
- Results show that HeterMPC achieves a new state-of-the-art performance for response generation in MPCs on the Ubuntu IRC benchmark.

Jia-Chen Gu

Chao-Hong Tan

Chongyang Tao

Zhen-Hua Ling

Huang Hu

Xiubo Geng

Daxin Jiang

Thanks! Q&A

Paper: https://aclanthology.org/2022.acl-long.349.pdf

Code: https://github.com/lxchtan/HeterMPC