Do Response Selection Models Really Know What's Next? Utterance Manipulation Strategies for Multi-turn Response Selection

Taesun Whang*, Dongyub Lee*, Dongsuk Oh, Chanhee Lee,
Kijong Han, Dong-hun Lee, and Saebyeok Lee

* Equal Contribution.

Multi-turn Response Selection

- Selecting the optimal response given a user and dialog context in multi-turn dialog systems.

[Dialog Context]

Good morning! What can I do for you?

I'm thinking of traveling to California in May.
Could you recommend some tourist programs for that?

With pleasure. We arrange two kinds of tourist programs for
California, a seven-day tour by bus and a five-day flying journey.

How much does a seven-day
tour by bus cost?

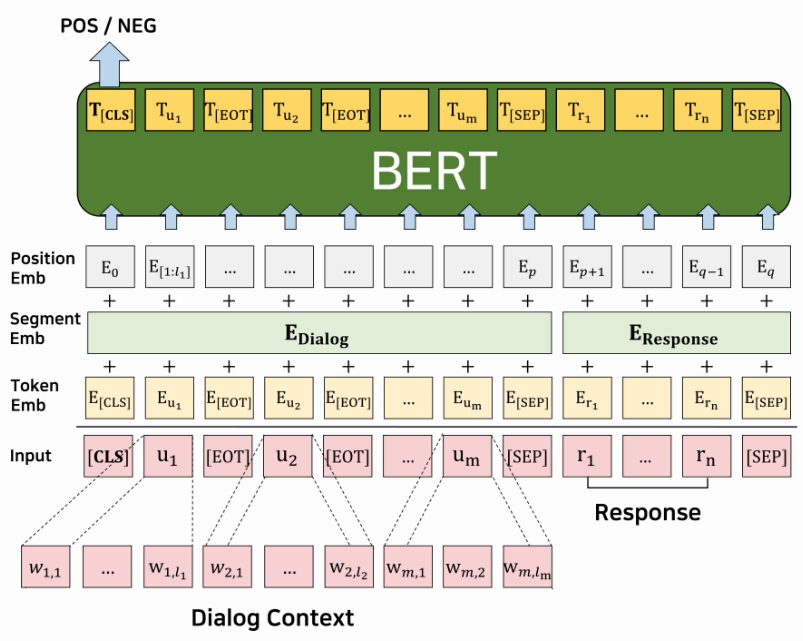
[Response Candidates]

1. Two thousand dollars. ✓
2. Does that include hotels and meals?

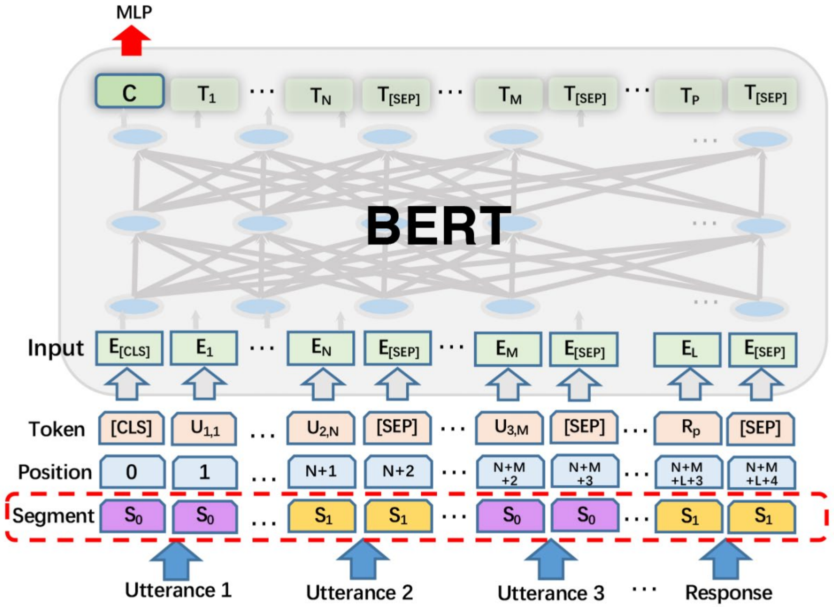
⋮

Recent success of PrLM based models

- BERT-VFT (Whang et al., 2020)



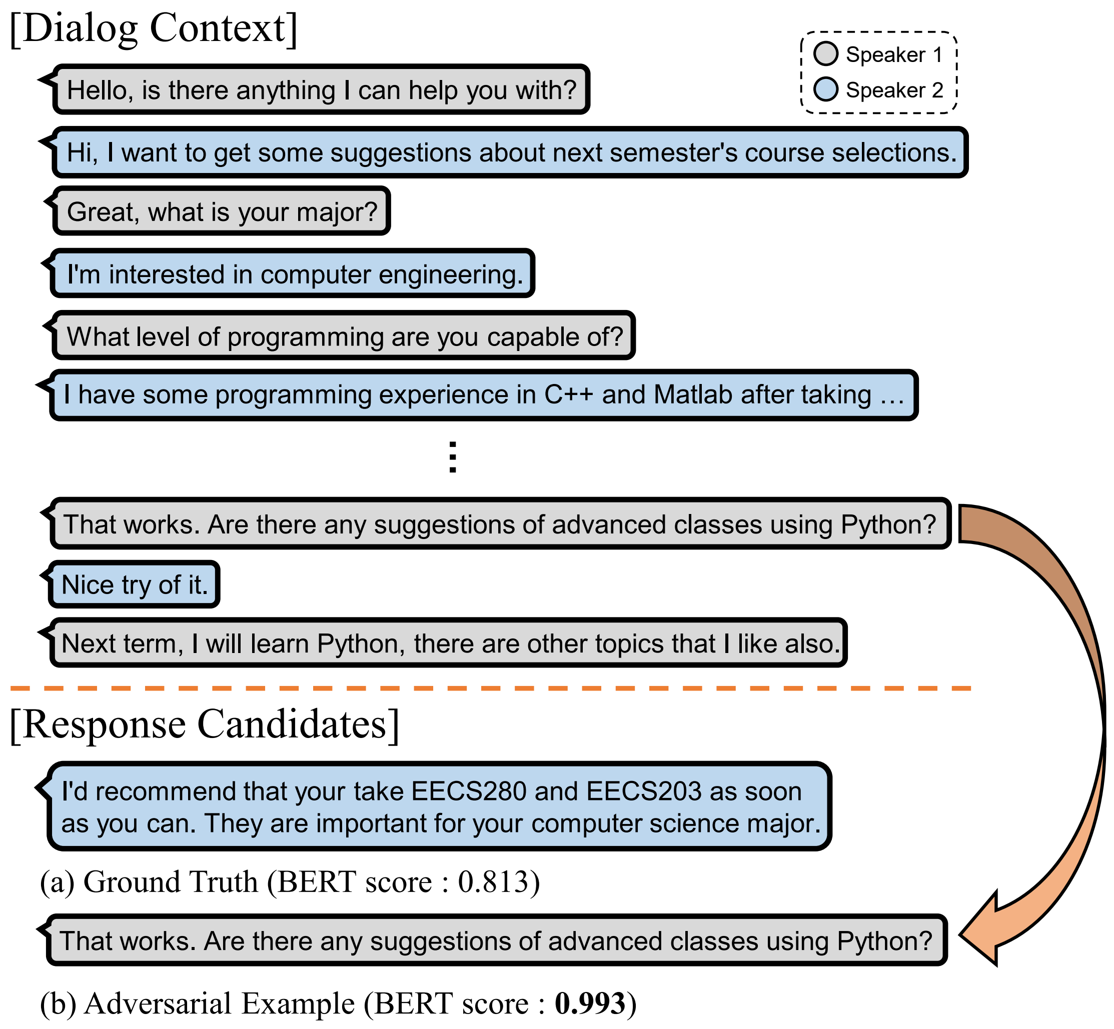
- BERT-SS-DA (Lu et al., 2020)



- Obtained state-of-the-art results.
- Tend to make predictions based on relatedness of history and candidates.
- Limits in adapting the sequential nature of multi-turn dialog.

Whang et al., An Effective Domain Adaptive Post-Training Method for BERT in Response Selection. INTERSPEECH 2020.
Lu et al., Improving Contextual Language Models for Response Retrieval in Multi-Turn Conversation. SIGIR 2020.

Recent success of PrLM based models (Adversarial Experiments)



Multi-turn Response Selection (Challenges)

- Domain adaption based on an additional training on a target corpus is extremely time-consuming and computationally costly.
- Formulating response selection as a dialog-response binary classification task is insufficient to represent intra- and inter-utterance interactions.
- Existing models tend to select the optimal response depending on how semantically similar it is to a given dialog.

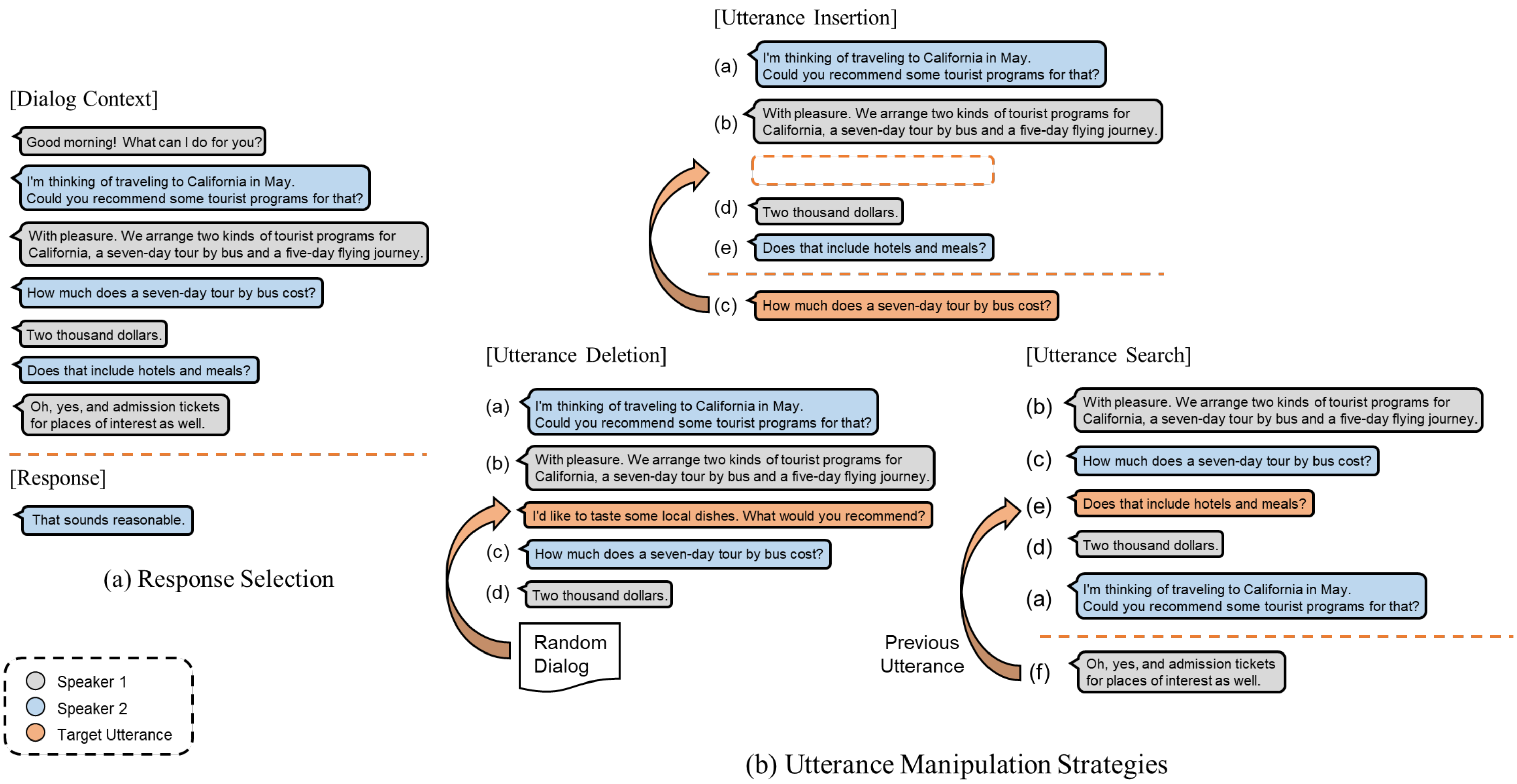
↓

Utterance Manipulation Strategies (UMS)

Contributions

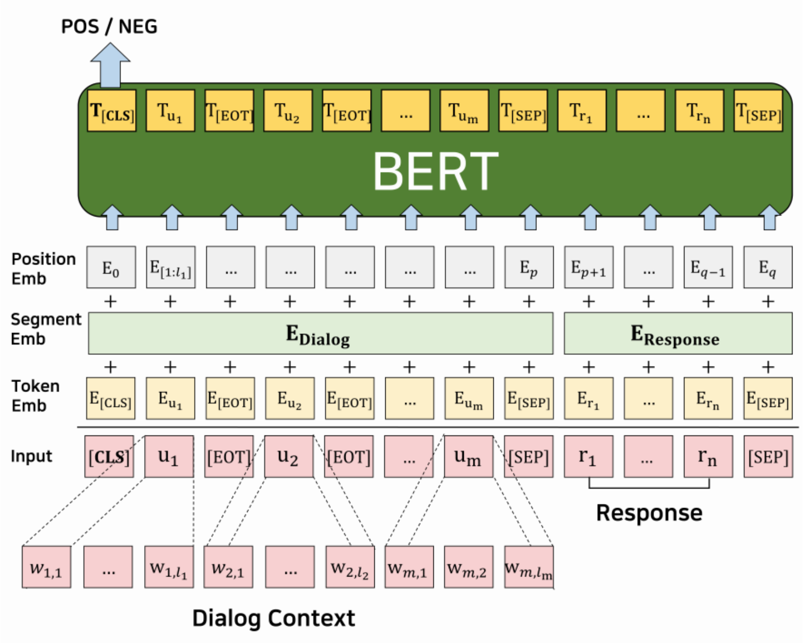
- Show that existing response selection models are more likely to predict a semantically relevant response with its dialog rather than the next utterance.
- Propose highly effective self-supervised learning methods, utterance manipulation strategies (UMS), which aid the model towards maintaining dialog coherence.
- State-of-the-art performance on multiple public benchmarks (i.e., Ubuntu, Douban, and E-commerce).

Proposed Method (Overview)



Proposed Method (Language Models for Response Selection)

- Pre-trained Language Models : BERT (Devlin et al., 2019), ELECTRA (Clark et al., 2020)
- Domain-specific Post-training (Additional training on a target corpus with PrLM objectives)
- Training Response Selection Models : BERT (Whang et al., 2020)



$$\mathbf{X} = [[\mathrm{CLS}]\, u_1\, u_2 \ldots u_{n_u}\, [\mathrm{SEP}]\, r\, [\mathrm{SEP}]]$$

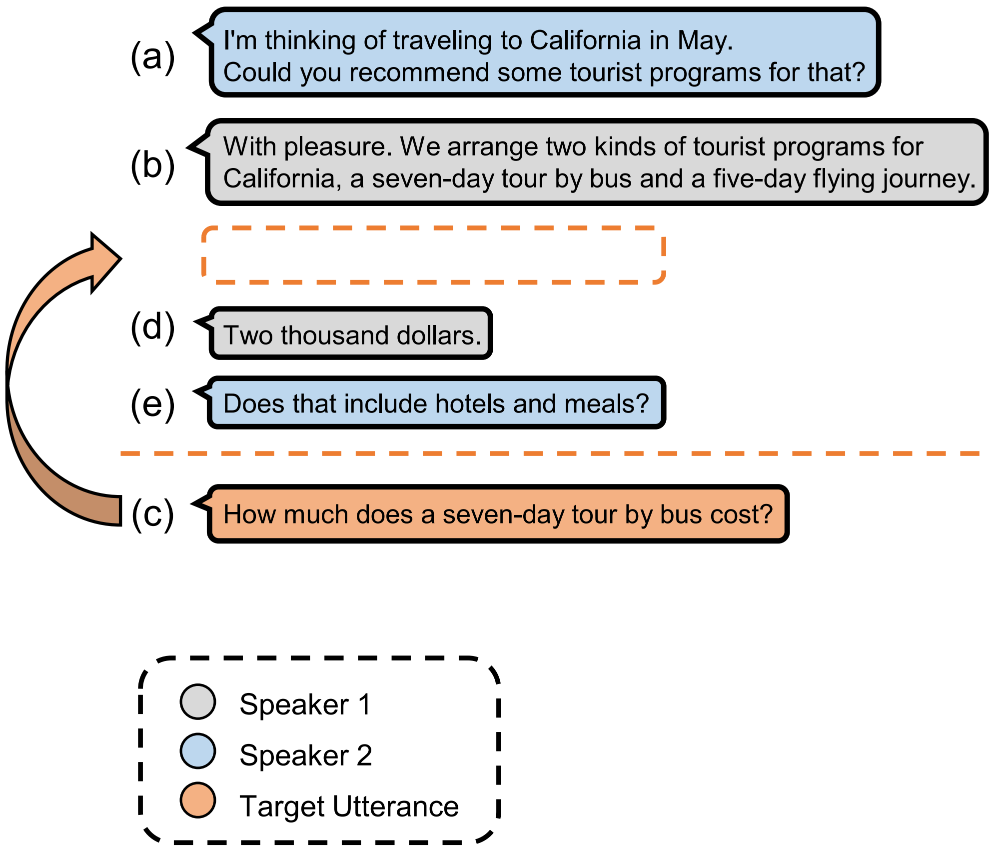
$$g(U, r) = \sigma(\mathbf{w}^\top \mathbf{x}_{[\mathrm{CLS}]} + b)$$

$$\mathcal{L}oss = - \sum_{(U,r,y) \in \mathcal{D}} y\log(g(U,r)) + (1-y)\log(1-g(U,r))$$

Devlin et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL 2019.
Clark et al., ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. ICLR 2020.
Whang et al., An Effective Domain Adaptive Post-Training Method for BERT in Response Selection. INTERSPEECH 2020.

Proposed Method (UMS – Utterance Insertion)

[Utterance Insertion]

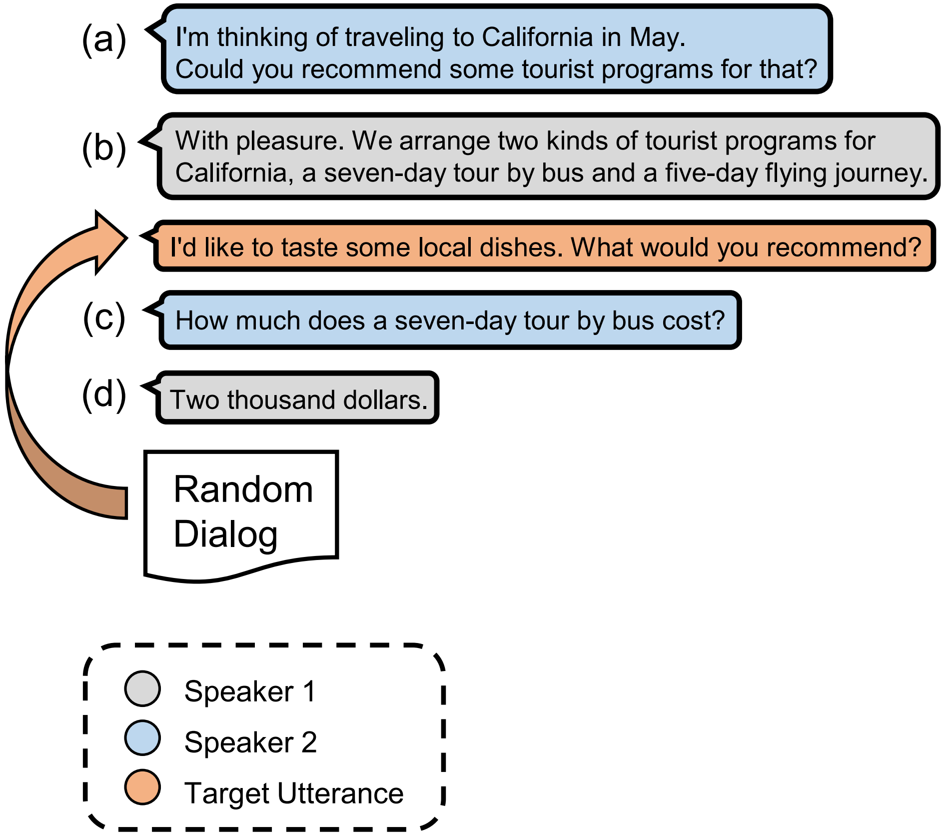


- Find where the selected utterance should be inserted.
- [INS] tokens are positioned before each utterance and after the last utterance.
- $\boldsymbol{u_t}$ is the target utterance and $[\mathbf{INS}]_{\boldsymbol{t}}$ is the target insertion token.

$$\mathbf{X}_{\mathrm{INS}} = [[\mathrm{CLS}]\ [\mathrm{INS}]_1\, u_1 [\mathrm{INS}]_2\, u_2 \ldots u_{t-1}$$
$$[\mathrm{INS}]_t\, u_{t+1} \ldots u_k\ [\mathrm{INS}]_k\ [\mathrm{SEP}]\ u_t\ [\mathrm{SEP}]]$$

Proposed Method (UMS – Utterance Deletion)

[Utterance Deletion]

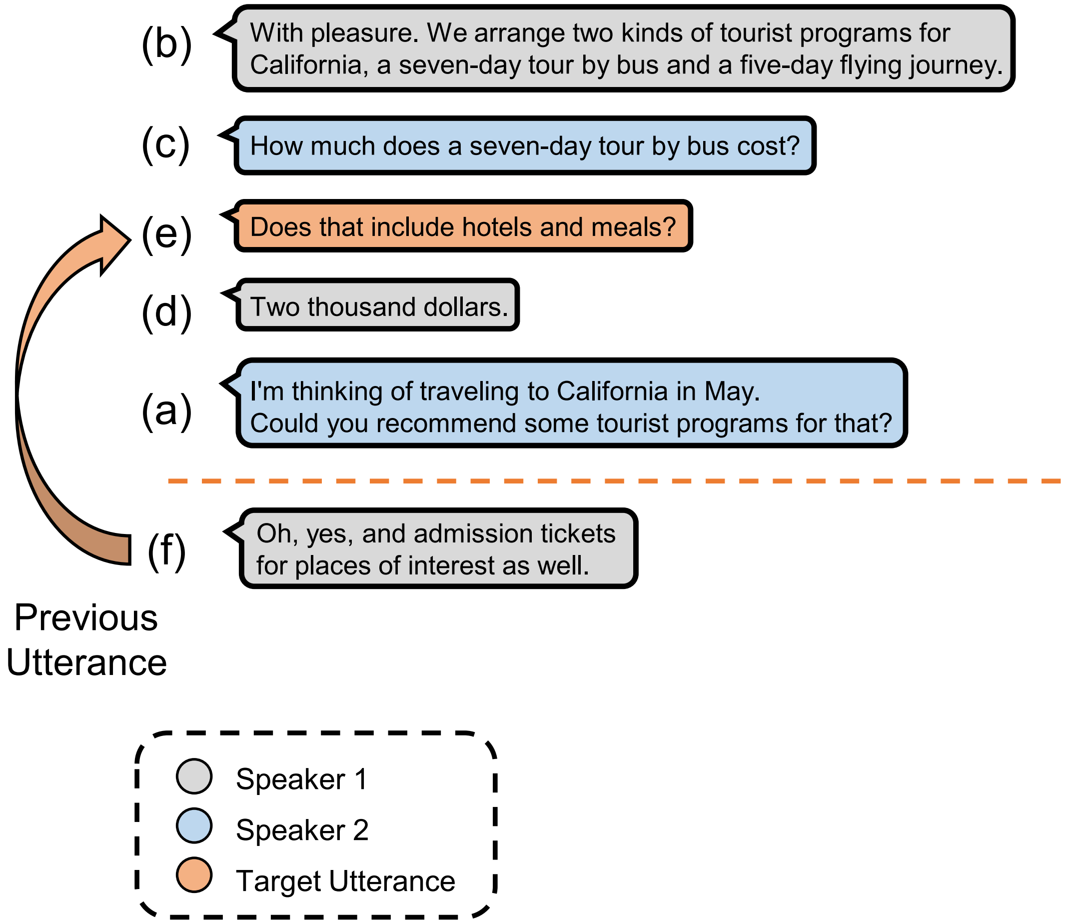


- Find an unrelated utterance to the dialog.
- The unrelated utterance is sampled from the random dialog.
- [DEL] tokens are positioned before each utterance.
- $\boldsymbol{u}^{\boldsymbol{rand}}$ is the utterance from the random dialog and $[\mathbf{DEL}]_{\boldsymbol{t}}$ is the target deletion token.

$$\mathbf{X}_{\text{DEL}} = [[\text{CLS}]\ [\text{DEL}]_1\ u_1\ [\text{DEL}]_2\ u_2 \ldots [\text{DEL}]_t$$
$$u^{rand}\ [\text{DEL}]_{t+1}\ u_t \ldots\ [\text{DEL}]_{k+1}\ u_k\ [\text{SEP}]]$$

Proposed Method (UMS – Utterance Search)

[Utterance Search]



- Find the previous utterance of the last utterance from the jumbled utterances.
- Shuffle utterances except for the last utterance.
- [SRCH] tokens are positioned before each utterance.
- $\boldsymbol{u}_t'$ ($\boldsymbol{u}_{k-1}$) is the previous utterance of the last utterance $\boldsymbol{u}_k$ and $[\mathbf{SRCH}]_t$ is the target search token.

$$\mathbf{X}_{\text{SRCH}} = [[\text{CLS}]\ [\text{SRCH}]_1\ u_1'\ [\text{SRCH}]_2\ u_2' \ldots \\ [\text{SRCH}]_t\ u_t' \ldots u_{k-1}'\ [\text{SEP}]\ u_k\ [\text{SEP}]]$$

Proposed Method (Multi-task Learning Setup)

- The output representations of special tokens ([INS], [DEL], and [SRCH]) are used to classify whether each toke is in a correct position to be inserted, deleted, and searched.
- Target tokens for each task ($[\mathbf{INS}]_t$, $[\mathbf{DEL}]_t$, and $[\mathbf{SRCH}]_t$) are labeled as 1, otherwise 0.

  $p(y_{\text{TASK}} = 1|\mathbf{X}_{\text{TASK}}) = \sigma(\mathbf{w}^\top \mathbf{x}_{\text{TASK}} + b)$, where $\text{TASK} \in \{\text{INS}, \text{DEL}, \text{SRCH}\}$

- Binary cross-entropy loss for all auxiliary tasks to optimize the model.
- Response Selection loss and UMS losses are summed with the same ratio.

Experimental Setup

- Dataset
  - Ubuntu : Ubuntu internet relay chats (troubleshooting the Ubuntu OS).
  - Douban : Chinese open-domain dialogs (web-crawled from Douban Group).
  - E-Commerce : Chinese customer consultation dialogs (Taobao).
  - Kakao : Korean open-domain (Twitter and Reddit) constructed by Kakao Corporation.
- Evaluation Metrics
  - R_n@k (k={1,2,5}), P@1, Mean Average Precision (MAP), Mean Reciprocal Rank (MRR)

Dataset	Ubuntu			Douban			E-Commerce			Kakao			
	Train	Val	Test	Train	Val	Test	Train	Val	Test	Train	Val	Test (Web)	Test (Clean)
# pairs	1M	500K	500K	1M	50K	6670	1M	10K	10K	1M	50K	5139	7164
pos:neg	1:1	1:9	1:9	1:1	1:1	1.2:8.8	1:1	1:1	1:9	1:1	1:1	1.6:7.4	2:7
# avg turns	10.13	10.11	10.11	6.69	6.75	6.45	5.51	5.48	5.64	3.00	3.00	3.49	3.25

Baselines

- Single-turn Matching Models
  - CNN, LSTM, BiLSTM
  - MV-LSTM, Match-LSTM
- Multi-turn Matching Models
  - Multi-View, DL2R
  - SMN, DUA, DAM, IoI
  - MSN
- BERT-based Models
  - Vanilla BERT, BERT-SS-DA, SA-BERT

Quantitative Results (Ubuntu, Douban, and E-Commerce Corpus)

Models	Ubuntu			Douban						E-commerce		
	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	MAP	MRR	P@1	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$
CNN (Kadlec, Schmid, and Kleindienst 2015)	0.549	0.684	0.896	0.417	0.440	0.226	0.121	0.252	0.647	0.328	0.515	0.792
LSTM (Kadlec, Schmid, and Kleindienst 2015)	0.638	0.784	0.949	0.485	0.537	0.320	0.187	0.343	0.720	0.365	0.536	0.828
BiLSTM (Kadlec, Schmid, and Kleindienst 2015)	0.630	0.780	0.944	0.479	0.514	0.313	0.184	0.330	0.716	0.365	0.536	0.825
MV-LSTM (Wan et al. 2016)	0.653	0.804	0.946	0.498	0.538	0.348	0.202	0.351	0.710	0.412	0.591	0.857
Match-LSTM(Wang and Jiang 2016)	0.653	0.799	0.944	0.500	0.537	0.345	0.202	0.348	0.720	0.410	0.590	0.858
Multi-View (Zhou et al. 2016)	0.662	0.801	0.951	0.505	0.543	0.342	0.202	0.350	0.729	0.421	0.601	0.861
DL2R (Yan, Song, and Wu 2016)	0.626	0.783	0.944	0.488	0.527	0.330	0.193	0.342	0.705	0.399	0.571	0.842
SMN (Wu et al. 2017)	0.726	0.847	0.961	0.529	0.569	0.397	0.233	0.396	0.724	0.453	0.654	0.886
DUA (Zhang et al. 2018)	0.752	0.868	0.962	0.551	0.599	0.421	0.243	0.421	0.780	0.501	0.700	0.921
DAM (Zhou et al. 2018)	0.767	0.874	0.969	0.550	0.601	0.427	0.254	0.410	0.757	0.526	0.727	0.933
IoI (Tao et al. 2019b)	0.796	0.894	0.974	0.573	0.621	0.444	0.269	0.451	0.786	0.563	0.768	0.950
MSN (Yuan et al. 2019)	0.800	0.899	0.978	0.587	0.632	0.470	0.295	0.452	0.788	0.606	0.770	0.937
BERT (Gu et al. 2020)	0.808	0.897	0.975	0.591	0.633	0.454	0.280	0.470	0.828	0.610	0.814	0.973
BERT-SS-DA (Lu et al. 2020)	0.813	0.901	0.977	0.602	0.643	0.458	0.280	0.491	0.843	0.648	0.843	0.980
SA-BERT (Gu et al. 2020)	0.855	0.928	0.983	0.619	0.659	0.496	0.313	0.481	0.847	0.704	0.879	0.985
BERT (ours)	0.820	0.906	0.978	0.597	0.634	0.448	0.279	<u>0.489</u>	0.823	0.641	0.824	0.973
ELECTRA	0.826	0.908	0.978	0.602	0.642	0.465	0.287	0.483	0.839	0.609	0.804	0.965
UMS_{BERT}	0.843	0.920	0.982	0.597	0.639	0.466	0.285	0.471	0.829	<u>0.674</u>	<u>0.861</u>	<u>0.980</u>
$\text{UMS}_{\text{ELECTRA}}$	<u>0.854</u>	<u>0.929</u>	<u>0.984</u>	<u>0.608</u>	<u>0.650</u>	<u>0.472</u>	<u>0.291</u>	0.488	<u>0.845</u>	0.648	0.831	0.974
BERT+	0.862	0.935	0.987	0.609	0.645	0.463	0.290	**0.505**	0.838	0.725	0.890	0.984
ELECTRA+	0.861	0.932	0.985	0.612	0.655	0.480	0.301	0.499	0.836	0.673	0.835	0.974
$\text{UMS}_{\text{BERT+}}$	**0.875**†	**0.942**†	**0.988**†	**0.625**	**0.664**	**0.499**	**0.318**	0.482	**0.858**	**0.762**	**0.905**	**0.986**
$\text{UMS}_{\text{ELECTRA+}}$	**0.875**	0.941	**0.988**	0.623	0.663	0.492	0.307	0.501	0.851	0.707	0.853	0.974

Quantitative Results (Ubuntu, Douban, and E-Commerce Corpus)

- Two different PrLMs (BERT and ELECTRA)
- Domain-specific post-training (denoted as BERT+ and ELECTRA+)
- ELECTRA vs $\text{UMS}_{\text{ELECTRA}}$
  - R_{10}@1: + 2.8% (Ubuntu), + 3.9% (E-Commerce)
  - P@1 : + 0.7% (Douban)
- BERT+ vs $\text{UMS}_{\text{BERT+}}$
  - R_{10}@1: + 1.3% (Ubuntu), + 3.7% (E-Commerce)
  - P@1 : + 3.3% (Douban)

Models	Ubuntu			Douban						E-commerce		
	R_{10}@1	R_{10}@2	R_{10}@5	MAP	MRR	P@1	R_{10}@1	R_{10}@2	R_{10}@5	R_{10}@1	R_{10}@2	R_{10}@5
BERT (ours)	0.820	0.906	0.978	0.597	0.634	0.448	0.279	0.489	0.823	0.641	0.824	0.973
ELECTRA	0.826	0.908	0.978	0.602	0.642	0.465	0.287	0.483	0.839	0.609	0.804	0.965
UMS_{BERT}	0.843	0.920	0.982	0.597	0.639	0.466	0.285	0.471	0.829	0.674	0.861	0.980
$\text{UMS}_{\text{ELECTRA}}$	0.854	0.929	0.984	0.608	0.650	0.472	0.291	0.488	0.845	0.648	0.831	0.974
BERT+	0.862	0.935	0.987	0.609	0.645	0.463	0.290	**0.505**	0.838	0.725	0.890	0.984
ELECTRA+	0.861	0.932	0.985	0.612	0.655	0.480	0.301	0.499	0.836	0.673	0.835	0.974
$\text{UMS}_{\text{BERT+}}$	**0.875**†	**0.942**†	**0.988**†	**0.625**	**0.664**	**0.499**	**0.318**	0.482	**0.858**	**0.762**	**0.905**	**0.986**
$\text{UMS}_{\text{ELECTRA+}}$	**0.875**	0.941	**0.988**	0.623	0.663	0.492	0.307	0.501	0.851	0.707	0.853	0.974

Quantitative Results (Kakao Corpus)

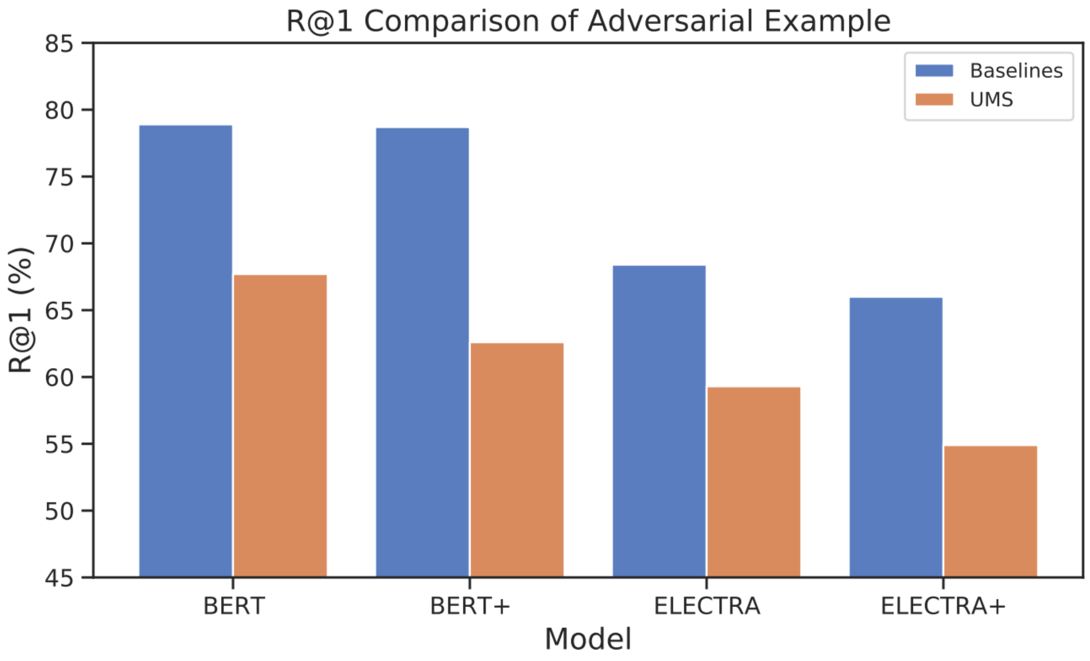
- BERT vs UMS_{BERT}
- UMS_{BERT} improves performance compared to the baseline for both Web and Clean.
- Absolute improvement of 5.1% (Web) and 6.8% (Clean) in P@1.

Test Split	Approach	MAP	MRR	$P@1$	$R_{10}@1$	$R_{10}@2$	$R_{10}@5$	P@1
Web	BERT	0.671	0.720	0.555	0.391	0.599	0.890	
	UMS_{BERT}	**0.699**	**0.751**	**0.606**	**0.428**	**0.623**	**0.911**	+ 5.1%
Clean	BERT	0.726	0.792	0.648	0.395	0.612	0.888	
	UMS_{BERT}	**0.761**	**0.834**	**0.716**	**0.431**	**0.663**	**0.903**	+ 6.8%

Adversarial Experiment

- Investigate whether language models for response selection are trained properly.
- Randomly extract an utterance from the dialog context and replace it with one of negative responses.
- R_{10}@1 score decreases by 58% (baselines) and 48 % (UMS) on average.

Approach	Model	Original		Adversarial	
		R_{10}@1	MRR	R_{10}@1	MRR
Baselines	BERT	0.820	0.887	0.199	0.561
	BERT+	**0.862**	**0.915**	0.203	0.573
	ELECTRA	0.826	0.890	0.304	0.614
	ELECTRA+	0.861	0.914	**0.329**	**0.636**
	Avg	0.842	0.902	0.259	0.596
UMS	BERT	0.843	0.902	0.310	0.622
	BERT+	**0.875**	**0.923**	0.363	0.656
	ELECTRA	0.854	0.910	0.397	0.668
	ELECTRA+	**0.875**	0.922	**0.437**	**0.692**
	Avg	0.862	0.914	0.377	0.660



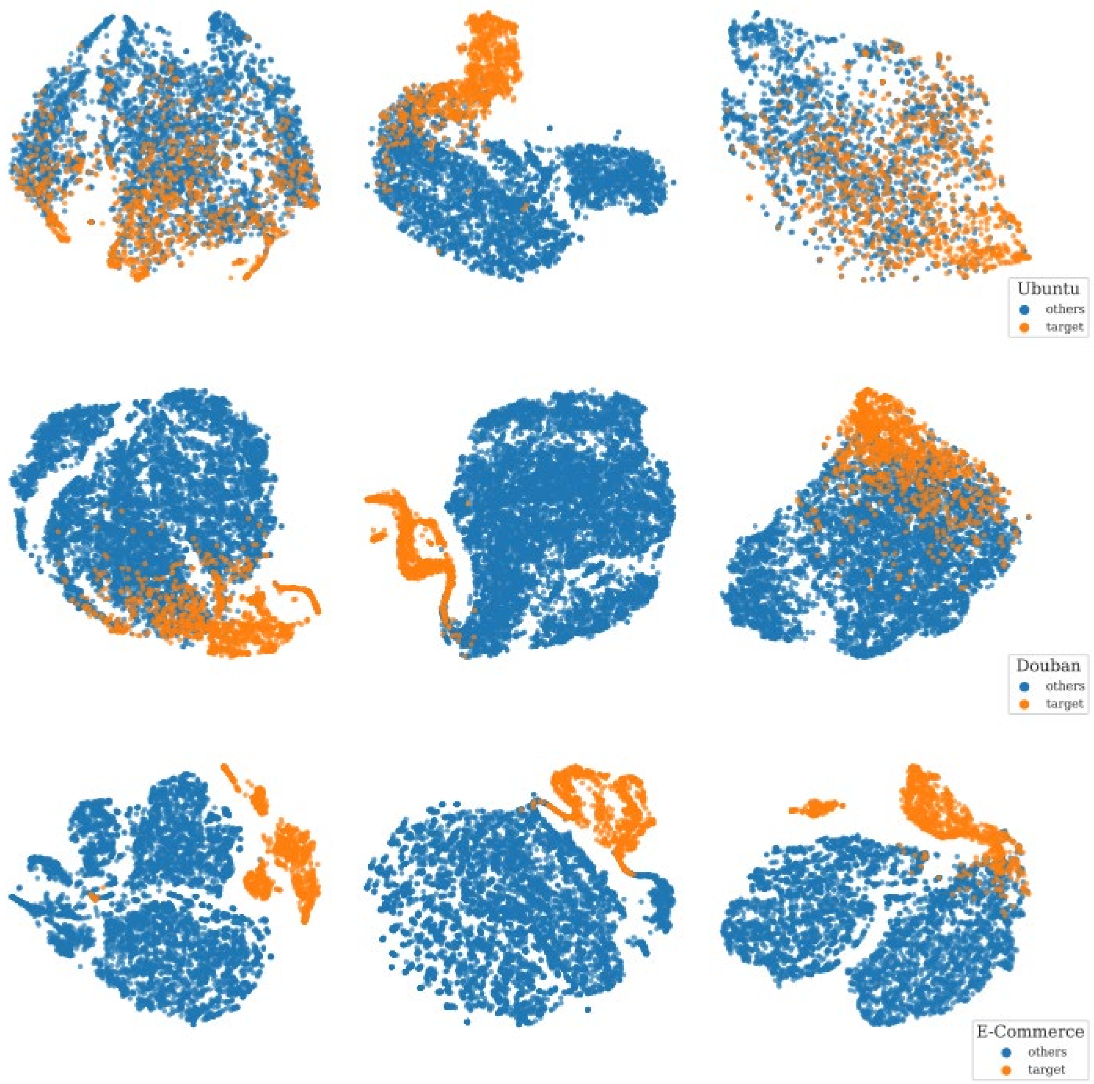
*Lower is better

Ablation Study

- One auxiliary task (i.e., 3 > 2 ≈ 4)
- Two auxiliary tasks (i.e., 5 ≈ 7 > 6)
- Overall, DEL > INS ≈ SRCH
- Improvement of 2.8% w.r.t. R_{10}@1

	Auxiliary Tasks	R_{10}@1	R_{10}@2	R_{10}@5	MRR	R_{10}@1
1	None	0.826	0.908	0.978	0.890	
2	INS	0.836	0.917	0.980	0.897	+ 1.0%
3	DEL	0.848	0.924	0.983	0.905	+ 2.2%
4	SRCH	0.834	0.915	0.981	0.896	+ 0.8%
5	INS + DEL	0.853	0.927	0.984	0.909	+ 2.7%
6	INS + SRCH	0.841	0.920	0.982	0.901	+ 1.5%
7	DEL + SRCH	0.852	0.927	0.983	0.908	+ 2.6%
8	INS + DEL + SRCH	**0.854**	**0.929**	**0.984**	**0.910**	+ 2.8%

Visualization



(a) Insertion (b) Deletion (c) Search

Conclusion

- Pointed out the limitations of existing works based on PrLMs, such as BERT in retrieval-based multi-turn dialog systems.
- Proposed highly effective utterance manipulation strategies (UMS) for multi-turn response selection.
- UMS are fully applied in self-supervised manner and can be easily incorporated into existing models.
- New state-of-the-art results on multiple public benchmark datasets.

Thank you

Our code is publicly available at