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

Taesun Whang[1]*, Dongyub Lee[2]*, Dongsuk Oh[3], Chanhee Lee[3], Kijong Han[4], Dong-hun Lee[4], and Saebyeok Lee[1,3]†

[1]Wisenut Inc.
[2]Kakao Corp.
[3]Korea University
[4]Kakao Enterprise Corp.
{taesunwhang, saebyeok}@wisenut.co.kr, jude.lee@kakaocorp.com
{inow3555, chanhee0222}@korea.ac.kr, {mat.h, hubert.std}@kakaoenterprise.com

Abstract

In this paper, we study the task of selecting the optimal response given a user and system utterance history in retrieval-based multi-turn dialog systems. Recently, pre-trained language models (*e.g.,* BERT, RoBERTa, and ELECTRA) showed significant improvements in various natural language processing tasks. This and similar response selection tasks can also be solved using such language models by formulating the tasks as dialog–response binary classification tasks. Although existing works using this approach successfully obtained state-of-the-art results, we observe that language models trained in this manner tend to make predictions based on the relatedness of history and candidates, ignoring the sequential nature of multi-turn dialog systems. This suggests that the response selection task alone is insufficient for learning temporal dependencies between utterances. To this end, we propose utterance manipulation strategies (UMS) to address this problem. Specifically, UMS consist of several strategies (*i.e.,* insertion, deletion, and search), which aid the response selection model towards maintaining dialog coherence. Further, UMS are self-supervised methods that do not require additional annotation and thus can be easily incorporated into existing approaches. Extensive evaluation across multiple languages and models shows that UMS are highly effective in teaching dialog consistency, which leads to models pushing the state-of-the-art with significant margins on multiple public benchmark datasets.

Introduction

In recent years, building intelligent conversational agents has gained considerable attention in the field of natural language processing (NLP). Among widely used dialog systems, retrieval-based dialog systems (Lowe et al. 2015; Wu et al. 2017; Zhang et al. 2018) are implemented in a variety of industries because they provide accurate, informative, and promising responses. In this study, we focus on multi-turn response selection in retrieval-based dialog systems. This is a task of predicting the most likely response under a given dialog history from a set of candidates.

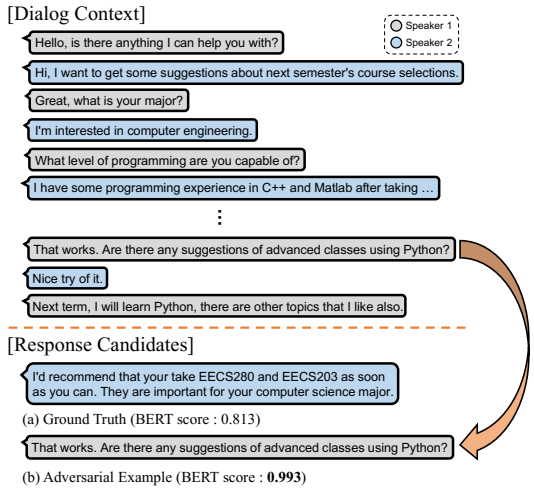

Figure 1: Example of multi-turn response selection. BERT-based model tends to calculate the matching score of a dialog–response pair depending on the semantic relatedness of the dialog and the response ((a) < (b)). More details are in Discussion section.

Existing works (Wu et al. 2017; Zhou et al. 2018; Tao et al. 2019a; Yuan et al. 2019) have studied utterance–response matching based on attention mechanisms including self-attention (Vaswani et al. 2017). Most recently, as pre-trained language models (*e.g.,* BERT (Devlin et al. 2019), RoBERTa (Liu et al. 2019), and ELECTRA (Clark et al. 2020)) have achieved substantial performance improvements in diverse NLP tasks, multi-turn response selection also has been resolved using such language models (Whang et al. 2020; Lu et al. 2020; Gu et al. 2020; Humeau et al. 2020).

However, we tackle three crucial problems in applying language models to response selection. 1) Domain adaptation based on an additional training on a target corpus is extremely time-consuming and computationally costly. 2) Formulating response selection as a dialog–response binary classification task is insufficient to represent intra- and inter-utterance inter-

*These authors equally contributed to this work.
†Corresponding author.

actions as the dialog context is formed by concatenating all utterances. 3) The models tend to select the optimal response depending on how semantically similar it is to a given dialog. As shown in Figure 1, we experiment to verify whether the BERT-based response selection model is trained properly to select the next utterance rather than dialog-related response. The results show that the model tends to give a higher probability score to a response that is more semantically related to the dialog context rather than consistent response. Although it is obvious that the ground truth is suitable for being the next utterance, the model highly depends on its semantic meaning.

To address these issues, this paper proposes Utterance Manipulation Strategies (UMS) for multi-turn response selection. Specifically, UMS consist of three powerful strategies (*i.e.,* insertion, deletion, and search), which effectively help the response selection model to learn temporal dependencies between utterances as well as semantic matching and maintain dialog coherence. In addition, these strategies are fully self-supervised methods that do not require additional annotation and can be easily adapted to existing studies. We briefly summarize the main contributions of this paper as follows: 1) We show that existing response selection models are more likely to predict a semantically relevant response with its dialog rather than the next utterance. 2) We propose simple but novel utterance manipulation strategies, which are highly effective in predicting the next utterance. Our model has strengths in effectively performing in-domain classification. 3) Experimental results on three benchmarks (*i.e.,* Ubuntu, Douban, and E-commerce) show that our proposed model outperforms state-of-the-art methods. We also obtain significant improvements in performance compared to the baselines on a new Korean open-domain corpus.

Proposed Method

Language Models for Response Selection

Pre-trained Language Models Recently, pre-trained language models, such as BERT (Devlin et al. 2019) and ELECTRA (Clark et al. 2020), were successfully adapted to a wide range of NLP tasks, including multi-turn response selection, achieving state-of-the-art results. In this work, we build upon this success and evaluate our method by incorporating it into BERT and ELECTRA.

Domain-specific Post-training As contextual language models are pre-trained on general corpora, such as the Toronto Books Corpus and Wikipedia, it is less effective to directly fine-tune these models on downstream tasks if there is a domain shift. Hence, it is a common practice to further train such models with the language modeling objective using texts from the target domain to reduce the negative impact. This has shown to be effective in various tasks including review reading comprehension (Xu et al. 2019) and SuperGLUE (Wang et al. 2019a). Existing works on multi-turn response selection (Whang et al. 2020; Gu et al. 2020; Humeau et al. 2020) also adapted this post-training approach and obtained state-of-the-art results. We also employ this post-training method in this work and show its effectiveness in improving performance.

Training Response Selection Models Following several researches based on contextual language models for multi-turn response selection (Whang et al. 2020; Lu et al. 2020; Gu et al. 2020), a pointwise approach is used to learn a cross-encoder that receives both dialog context and response simultaneously. Suppose that a dialog agent is given a dialog dataset $\mathcal{D} = \{(U_i, r_i, y_i)\}_{i=1}^{N}$. Each triplet consists of 1) a sequence of utterances $U_i = [u_1^i, u_2^i, \cdots, u_{|U|}^i]$ representing the historical context, where u_t^i is a single utterance, 2) a response r_i, and 3) a label $y_i \in \{0, 1\}$. Each utterance u_t^i and response r_i are composed of multiple tokens including a special [EOT] token at the end of each utterance, following the work of Whang et al. (2020). In general, the input sequence,

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

is fed into pre-trained language models (*i.e.,* BERT, ELECTRA), then output representation of [CLS] token, $\mathbf{x}_{[\text{CLS}]} \in \mathbb{R}^{d\times 1}$, is used to classify whether the dialog–response pair is consistent. The relevance score of the dialog utterances and response is formulated as,

$$g(U, r) = \sigma(\mathbf{w}^\top \mathbf{x}_{[\text{CLS}]} + b), \tag{1}$$

where $\mathbf{w} \in \mathbb{R}^{d\times 1}$ and b are the trainable parameters. We use binary cross-entropy loss to optimize the models.

Utterance Manipulation Strategies

Figure 2 describes the overview of our proposed method, utterance manipulation strategies. We propose a multi-task learning framework, which consists of three highly effective auxiliary tasks for multi-turn response selection, utterance 1) *insertion*, 2) *deletion*, and 3) *search*. These tasks are jointly trained with the response selection model during the fine-tuning period. To train the auxiliary tasks, we add new special tokens, [INS], [DEL], and [SRCH] for the utterance insertion, deletion, and search tasks, respectively. We cover how we train the model with these special tokens in the following sections.

Utterance Insertion Despite the huge success of BERT, it has limitations in understanding discourse-level semantic structure since NSP, one of BERT's objectives, mainly learns semantic topic shift rather than sentence order (Lan et al. 2020). In multi-turn response selection, the model needs the ability not only to distinguish the utterances with different semantic meanings but also to discriminate whether the utterances are consecutive even if they are semantically related. We propose *utterance insertion* to resolve the aforementioned issues.

We first extract k consecutive utterances from the original dialog context, then randomly select one of the utterances to be inserted. To train the model to find where the selected utterance should be inserted, [INS] tokens are positioned before each utterance and after the last utterance. [INS] tokens are represented as possible position of the target utterance. Input sequence for utterance insertion is denoted as

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

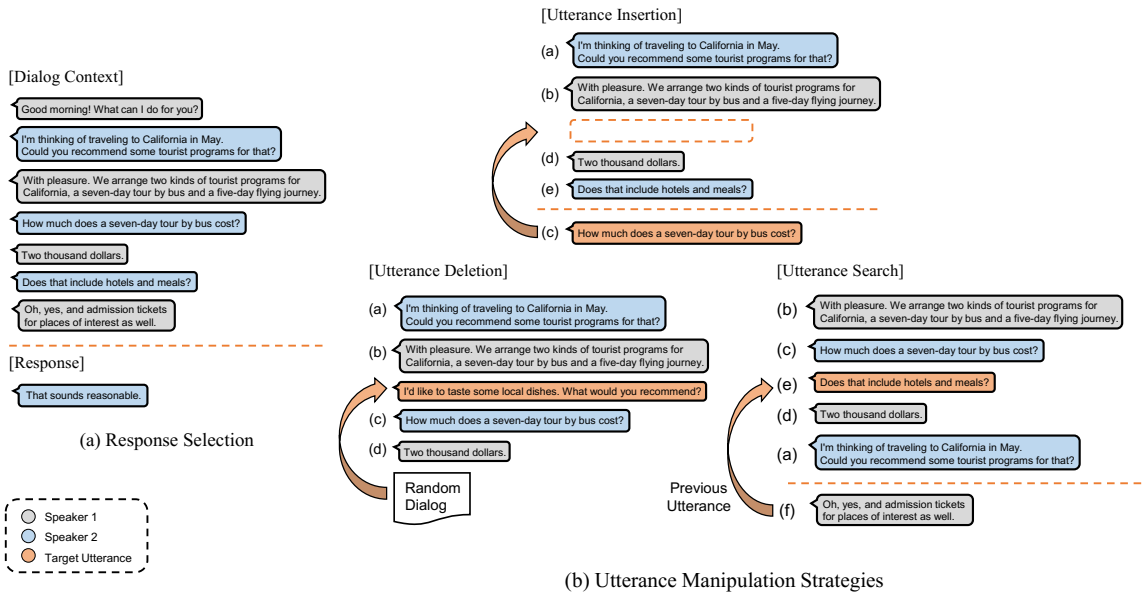

Figure 2: An overview of Utterance Manipulation Strategies. Input sequence for each manipulation strategy is dynamically constructed by extracting k consecutive utterances from the original dialog context during the training period. Also, target utterance is randomly chosen from either the dialog context (Insertion, Search) or the random dialog (Deletion).

where u_t is the target utterance and $[\text{INS}]_t$ is the target insertion token.

Utterance Deletion Recent BERT-based models for multi-turn response selection regard the task as a dialog–response binary classification. Even though they are extended in a multi-turn manner using separating tokens (*e.g.,* [SEP], [EOT]), these models lack utterance-level interaction between dialog context and response. To alleviate this, we propose a highly effective auxiliary task, *utterance deletion*, to enrich utterance-level interaction in multi-turn conversation.

As with utterance insertion, k consecutive utterances are extracted from the original dialog context, and then an utterance from a random dialog is inserted among the k extracted utterances. In other words, $k+1$ utterances are composed of k utterances from the original conversation and one from different dialogs. To train the model to find an unrelated utterance, [DEL] tokens are positioned before each utterance. The objective of the utterance deletion task is to predict which utterance causes inconsistency. We denote the input sequence for utterance deletion as

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

where u^{rand} is the utterance from the random dialog and $[\text{DEL}]_t$ is the target deletion token.

Utterance Search Whereas two previous auxiliary tasks are performed in a properly ordered dialog, we design a novel task, *utterance search*, which aims to find an appropriate utterance from randomly shuffled utterances. The objective of this task is to learn temporal dependencies between semantically similar utterances.

Given k consecutive utterances same as the previous tasks, we shuffle utterances except for the last utterance and insert [SRCH] tokens before each shuffled utterance. The utterance search aims to find the previous utterance of the last utterance from the jumbled utterances. Input sequence for utterance search is denoted as

$$\mathbf{X}_{\text{SRCH}} = [[\text{CLS}]\,[\text{SRCH}]_1\,u'_1[\text{SRCH}]_2\,u'_2\,... \\ [\text{SRCH}]_t\,u'_t\,...\,u'_{k-1}[\text{SEP}]\,u_k\,[\text{SEP}]],$$

where $\{u'_t\}_{t=1}^{k-1}$ is a set of utterances which are randomly shuffled except for the last utterance u_k. The previous utterance of u_k is denoted as u'_t (*i.e.,* u_{k-1}) and $[\text{SRCH}]_t$ is the target search token.

Multi-Task Learning Setup

The input sequence of each task is fed into the language models. The output representations of special tokens (*i.e.,* [INS], [DEL], and [SRCH]) are used to classify whether each token is in a correct position to be inserted, deleted, and searched. Target tokens for each task (*i.e.,* $[\text{INS}]_t$, $[\text{DEL}]_t$, and $[\text{SRCH}]_t$) are labeled as 1, otherwise 0. We calculate the probability of the token being a target as follows:

$$p(y_{\text{TASK}} = 1|\mathbf{X}_{\text{TASK}}) = \sigma(\mathbf{w}^\top \mathbf{x}_{\text{TASK}} + b), \qquad (2)$$

where $\text{TASK} \in \{\text{INS}, \text{DEL}, \text{SRCH}\}$ and $\mathbf{x}_{\text{TASK}}$ is the output representations of each special token. We use binary cross-entropy loss for all auxiliary tasks to optimize each model. The final loss is determined by summing the response selection loss and UMS losses with the same ratio.

Experimental Setup

Datasets

We evaluate our model on three widely used response selection benchmarks: *Ubuntu Corpus V1* (Lowe et al. 2015),

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

Table 1: Corpus statistics of multi-turn response selection datasets.

Douban Corpus (Wu et al. 2017), and *E-Commerce Corpus* (Zhang et al. 2018). In addition, a new open-domain dialog corpus, *Kakao Corpus*, is utilized to evaluate our model. All datasets consist of dyadic multi-turn conversations, and their statistics are summarized in Table 1.

Ubuntu Corpus V1 Ubuntu dataset is a large multi-turn conversation corpus that is constructed from Ubuntu internet relay chats. It mainly consists of conversations between two participants who discuss how to troubleshoot the Ubuntu operating system. We utilize the data released by Xu et al. (2017), where numbers, URLs, paths are replaced with special placeholders following previous works (Wu et al. 2017; Zhou et al. 2018).

Douban Corpus Douban dataset is a Chinese open-domain dialog corpus, whereas the Ubuntu Corpus is a domain-specific dataset. It is constructed by web-crawling from the Douban group[1], which is a popular social networking service (SNS) in China.

E-commerce Corpus E-Commerce dataset is another Chinese multi-turn conversation corpus. It is collected from real-world customer consultation dialogs from Taobao[2], which is the largest Chinese e-commerce platform. It consists of several types of conversations (*e.g.,* commodity consultation, recommendation, and negotiation) based on various commodities.

Kakao Corpus Kakao dataset is a large Korean open-domain dialog corpus that is constructed by Kakao Corporation[3]. It is mainly web-crawled from Korean SNSs such as Korean Twitter and Reddit. In a similar manner to the Ubuntu dataset, we take the last utterance of the dialog as a positive response and the rest as a dialog context. Negative responses are randomly sampled from the other conversations. We split the test set into two sets: 1) *web* is the same as the training set, and 2) *clean* consists of grammatically correct conversations that are constructed by human annotators and inspected by NLP experts.

Evaluation Metrics

We evaluated our model using several retrieval metrics, following previous research (Lowe et al. 2015; Wu et al. 2017; Zhou et al. 2018; Yuan et al. 2019). First, we employ 1 in n recall at k, denoted as $R_n@k$ ($k = \{1, 2, 5\}$), which gets 1 when a ground truth is positioned in the k selected list and 0 otherwise. In addition, three other metrics [mean average precision (MAP), mean reciprocal rank (MRR), and precision at one (P@1)] are used especially for Douban and Kakao, as these two datasets may contain more than one positive response among candidates.

Training Details

We implemented our model[4] by using the PyTorch deep learning framework (Paszke et al. 2019) based on the open-source code[5] (Wolf et al. 2019). As we experimented on three different languages (*i.e.,* English, Chinese, and Korean), initial checkpoints for BERT and ELECTRA are adapted from several works (Devlin et al. 2019; Clark et al. 2020; Cui et al. 2020; Lee et al. 2020). Specifically, we employ *base* pre-trained models for all languages except for Chinese (the whole-word masking (WWM) strategy is used for Chinese BERT[6]). As ELECTRA for Korean is not available, we do not conduct ELECTRA-based experiments on the Kakao Corpus. All experiments, both post-training and fine-tuning, are run on 4 Tesla V100 GPUs. For fine-tuning, we trained the models with a batch size of 32 using the Adam optimizer with an initial learning rate of 3e-5. The maximum sequence length is set to 512 and k for UMS is set to 5.

Baselines

Single-turn Matching Models These baselines, including CNN, LSTM, BiLSTM (Kadlec, Schmid, and Kleindienst 2015), MV-LSTM (Wan et al. 2016), and Match-LSTM (Wang and Jiang 2016), are based on matching between a dialog context and a response. They construct the dialog context by concatenating utterances and regard it as a long document.

Multi-turn Matching Models Multi-View (Zhou et al. 2016) utilize both word- and utterance-level representations. DL2R (Yan, Song, and Wu 2016) reformulates the last utterance with previous utterances in the dialog context. SMN (Wu et al. 2017) first constructs attention matrices based on word and sequential representations of each utterance and response, and then obtains matching vectors by using CNN. DUA (Zhang et al. 2018) utilizes deep utterance aggregation to form a fine-grained context representation. DAM (Zhou et al. 2018) obtains matching representations of the utterances and response using self- and cross-attention based on Transformer architecture (Vaswani et al. 2017). IoI (Tao et al. 2019b) lets utterance–response interaction go deep in a matching model. MSN (Yuan et al. 2019) filters only relevant utterances using a multi-hop selector network.

BERT-based Models Recently, BERT (Devlin et al. 2019) is also applied to response selection, such as vanilla BERT(Gu et al. 2020), BERT-SS-DA (Lu et al. 2020), and SA-BERT (Gu et al. 2020). In these models, the dialog context is represented as a long document, as in single-turn matching models.

[1]https://www.douban.com

[2]https://www.taobao.com

[3]https://www.kakaocorp.com

[4]https://github.com/taesunwhang/UMS-ResSel

[5]https://github.com/huggingface/transformers

[6]https://github.com/ymcui/Chinese-BERT-wwm

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	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

Table 2: Results on Ubuntu, Douban, and E-Commerce datasets. All the evaluation results except ours are cited from published literature (Tao et al. 2019b; Yuan et al. 2019; Gu et al. 2020). The underlined numbers mean the best performance for each block and the bold numbers mean state-of-the-art performance for each metric. † denotes statistical significance (p-value < 0.05).

They mainly utilize speaker information of each utterance in the dialog context to extend BERT into a multi-turn fashion.

Results and Discussion

Quantitative Results

Table 2 lists the quantitative results on Ubuntu, Douban, and E-Commerce datasets. In our experiments, we set two conditions for pre-trained language models. 1) Two different pre-trained language models (*i.e.,* BERT and ELECTRA) are utilized for fine-tuning. 2) We adapt domain-specific post-training approach (each post-trained model is denoted as BERT+ and ELECTRA+). Based on these initial settings, we explore how effective UMS are for multi-turn response selection.

For all datasets, models with UMS significantly outperform the previous state-of-the-art methods. Specifically, $\text{UMS}_{\text{BERT+}}$ achieves absolute improvements of 2.0% and 5.8% in R_{10}@1 on Ubuntu and E-Commerce datasets, respectively. For Douban datset, MAP and MRR are considered to be main metrics rather than R_{10}@1 because the test set contains more than one ground truth in the candidates. $\text{UMS}_{\text{BERT+}}$ achieves absolute improvements of 0.5% in these metrics.

To evaluate the effectiveness of UMS, we compare the models with UMS and those without them. Since existing BERT-based approaches (Lu et al. 2020; Gu et al. 2020) reported different performances of BERT, we reimplemented it for a fair comparison with our proposed UMS_{BERT}. The models with UMS consistently show performance improvement regardless of whether language models are post-trained on each corpus or not. For the models without post-training, different results are obtained depending on the dataset. ELECTRA mainly shows better results for the Ubuntu and Douban datasets, while BERT shows better results for the E-Commerce dataset. By contrast, BERT+ achieves the best performance for all corpora in comparison among the models with post-training. We believe that post-training on domain-specific corpus provides the model with more opportunities to learn whether given two dialogs are relevant through NSP; this has the effect of data augmentation.

Test Split	Approach	MAP	MRR	P@1	R_{10}@1	R_{10}@2	R_{10}@5
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**
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**

Table 3: Evaluation Results on Kakao Corpus.

Results on Kakao Corpus We report the evaluation results on the Kakao Corpus in Table 3. As ELECTRA for Korean is unavailable, we only compare BERT and UMS_{BERT} for two test splits. *Clean* shows better results than *Web* with respect to all metrics regardless of using UMS. This might be because *Clean* contains fewer grammatical errors and typos that interfere with an accurate understanding of the context. Also, UMS_{BERT} significantly improves performance compared to the baseline for both split; specifically, it achieves absolute improvements of 5.1% and 6.8% in P@1 on *Web* and *Clean*, respectively.

Adversarial Experiment

Even though BERT-based models have shown state-of-the-art performance for response selection task, we experiment to know if these models are trained to predict the next utterance properly. Inspired by Jia and Liang (2017) and Yuan et al. (2019), we design an adversarial experiment to investigate whether language models for response selection are trained

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

Table 4: Adversarial experimental results on Ubuntu Corpus. All models are evaluated using R_{10}@1 and MRR metrics.

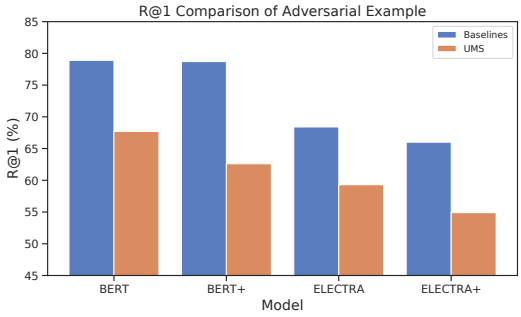

Figure 3: R_{10}@1 comparison of adversarial example for each model. Lower R_{10}@1 means that it is good at predicting the next utterance (ground truth).

properly. First, we train the models using the original training set, then evaluate them on either original or adversarial test set. To construct the adversarial test set, we randomly extract an utterance from the dialog context and replace it with one of negative responses among candidates (See Figure 1). In adversarial test set, assuming there are n candidates per conversation, a set of candidates consists of a ground truth, an extracted utterance from the dialog context, and $n-2$ negative responses. The extracted utterance is not deleted from the original dialog because it can be crucial for selecting the optimal response.

Table 4 lists the experimental results of BERT(+) and ELECTRA(+) models. We compare the models without UMS and those with, denoted as baselines and UMS, respectively. Even though the performances drop significantly in the adversarial set regardless of whether UMS are used or not, we observe that the UMS decline less than baselines. Specifically, R_{10}@1 score decreases by 58% and 48% on average for baselines and UMS, respectively. It is also encouraging that UMS show an absolute improvement of 12% with respect to R_{10}@1 on the adversarial set compared to the 2% improvement on the original set (See Table 4). In addition, while baselines tend to drop in performance on the adversarial set as training progresses, the performance of UMS shows a tendency to increase significantly. Hence, it is reasonable to assume that our UMS are robust to adversarial examples and are good at *in-domain* classification.

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

Table 5: Ablation Study on Ubuntu Corpus. We choose ELECTRA as the baseline in this analysis. INS, DEL, and SRCH denote that the model trained with utterance insertion, deletion, and search, respectively.

Figure 3 describes the performance of each model, ranking adversarial example (*i.e.,* randomly sampled utterance from the conversation) as the most likely response. While BERT- and ELECTRA-based models show similar performance on the original set, ELECTRA-based models outperform BERT-based models with significant margins (a gap of 10%) on the adversarial set regardless of whether they are trained from post-trained checkpoints. For example, different patterns of the evaluation results between BERT+ and ELECTRA are observed according to the test sets (original : BERT+ $>$ ELECTRA, adversarial : BERT+ $<$ ELECTRA). We have two perspectives on these results: 1) *Next sentence prediction* in BERT overfits the model to predict semantically relevant sentence rather than the next sentence. 2) As ELECTRA is trained through *replaced token detection* in which the model learns to discriminate between real input tokens and replacements generated from small *masked language model*, it is more effective in representing contextual information from the sequence.

Ablation Study

We performed ablation studies on the Ubuntu Corpus to investigate which auxiliary tasks are more crucial for response selection. As shown in Table 5, we explored the impact of each auxiliary task by constructing all combinations of possible subsets. Based on the observations of using only one auxiliary task (*i.e.,* $3 > 2 \approx 4$) and two tasks (*i.e.,* $5 \approx 7 > 6$), we obtained the results, DEL $>$ INS $\approx$ SRCH, with respect to the importance of manipulation strategy. As DEL consists of an input sequence that contains an irrelevant utterance to the original dialog context, it may be more advantageous for learning to distinguish dialog consistency and coherence than INS and SRCH. We obtain the best results when all the auxiliary tasks are trained simultaneously with the response selection criterion.

Visualization

As shown in Figure 4, we visualize the output representations of special tokens learned by our proposed UMS through t-SNE embeddings. Scatter plots colored in orange represent target tokens (*i.e.,*$[\text{INS}]_t$, $[\text{DEL}]_t$, and $[\text{SRCH}]_t$) and those in blue represent the rest of tokens. All representations are extracted from test sets of three datasets (Ubuntu, Douban, and E-Commerce) in this analysis. Overall, the results show

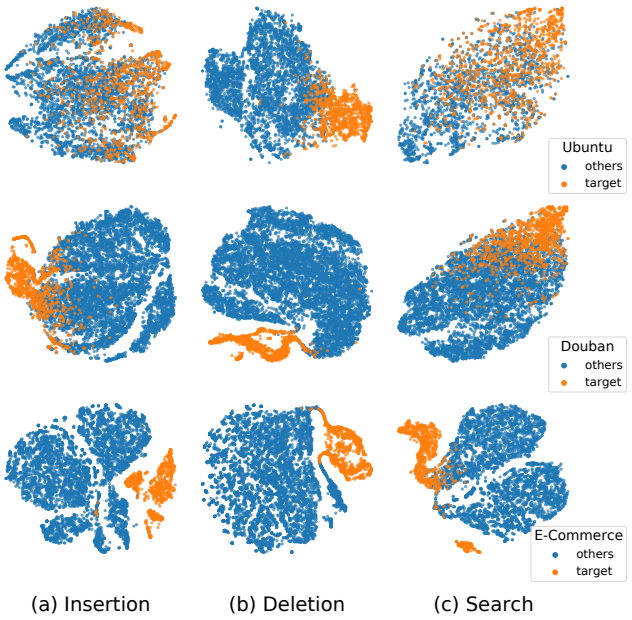

Figure 4: t-SNE embeddings of $\text{UMS}_{\text{BERT+}}$ output representations for each special token in UMS (*i.e.,* [INS], [DEL], and [SRCH]). All embeddings are sampled from test sets of each dataset.

that $\text{UMS}_{\text{BERT+}}$ effectively learns dialog coherence for all datasets. In the case of Ubuntu dataset, insertion and search tasks tend to be less clustered than that of the other two datsets. As many utterances in the Ubuntu dataset mainly consist of many technical terminologies that may cause structural ambiguity, tasks constructed within the same dialog are difficult to be performed. By contrast, the model can easily learn discourse structures on open-domain datasets such as Douban and E-Commerce.

Related Work

Multi-turn Response Selection Early approaches to response selection focused on single-turn response selection (Wang et al. 2013; Hu et al. 2014; Wang et al. 2015). Recently, multi-turn response selection has received more attention by researchers. Lowe et al. (2015) proposed dual encoder architecture which uses an RNN-based models to match the dialog and response. Zhou et al. (2016) proposed the multi-view model that encodes dialog context and response both on the word-level and utterance-level. However, these models have limitations in fully reflecting the relationship between dialog and response. To alleviate this, Wu et al. (2017) proposed the sequential matching network that utilizes matching matrices to match each utterance with a response. As self-attention (Vaswani et al. 2017) mechanism has been proved its effectiveness, it is applied in subsequent works (Zhou et al. 2018; Tao et al. 2019a,b). Yuan et al. (2019) recently pointed out that previous approaches construct dialog representation with abundant information but noisy, which deteriorates the performance. They proposed an effective history filtering technique to avoid using excessive history information.

Most recently, many researches based on pre-trained language models including BERT (Devlin et al. 2019) and RoBERTa (Liu et al. 2019) are proposed. Generally, most models formulate the response selection task as a dialog-response binary classification task. Whang et al. (2020) first applied BERT for multi-turn response selection and obtained state-of-the-art results through further training BERT on domain-specific corpus. Subsequent researches (Lu et al. 2020; Gu et al. 2020) focused on modeling speaker information and showed its effectiveness in response retrieval. Humeau et al. (2020) investigated the trade-off relationship between model complexity and computation efficiency in the language models. They proposed poly-encoders that ensure fast inference speed, even though the performance is slightly lower than that of the cross-encoder.

Self-supervised Learning Self-supervised learning has been explored in various pre-trained language models (Devlin et al. 2019; Clark et al. 2020; Lewis et al. 2020; Joshi et al. 2020) and is also applied in several NLP downstream tasks, such as summarization (Wang et al. 2019b), disfluency detection (Wang et al. 2020), and response generation (Zhao, Xu, and Wu 2020). Existing works in dialog modeling (Wu, Wang, and Wang 2019; Mehri et al. 2019; Liang, Zou, and Yu 2020) mainly focused on building enhanced dialog representations through self-supervised learning. Although the methods proposed in Wu, Wang, and Wang (2019) and Liang, Zou, and Yu (2020) effectively learn to rank coherent dialog higher than corrupted ones, but they have limitations in identifying the utterance that actually caused the inconsistency. Our strategy is different in that it learns to find which utterance is replaced from the full dialog. By doing so, our model can learn which utterance does not suit the conversation, which makes the model learn not only to discriminate semantic differences but also to build coherent dialog. The method proposed in Mehri et al. (2019) is somewhat similar to our deletion task, but they directly use the utterance representation to build the loss. We hypothesize that this is the reason behind the inconsistent improvements in Mehri et al. (2019), where in some downstream tasks the auxiliary task was actually harmful. On the other hand, our approach introduces special tokens that is different from the [CLS] token used in downstream tasks. Our results show that this approach consistently leads to improvements.

Conclusion

In this paper, we pointed out the limitations of existing works based on pre-trained language models such as BERT in retrieval-based multi-turn dialog systems. To address these, we proposed highly effective utterance manipulation strategies (UMS) for multi-turn response selection. The UMS are fully applied in self-supervised manner and can be easily incorporated into existing models. We obtained new state-of-the-art results on multiple public benchmark datasets (*i.e.,* Ubuntu, Douban, and E-Commerce) and significantly improved results on Korean open-domain dialog corpus. For the future work, we plan to develop a response selection model which is more robust to adversarial examples by designing various adversarial objectives.

Acknowledgments

This work was partly supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 1711117120, A Neural-Symbolic Model for Knowledge Acquisition and Inference Techniques) and funded by the Korean National Police Agency (No. PR09-01-000-20, Pol-Bot Development for Conversational Police Knowledge Services)

References

Clark, K.; Luong, M.-T.; Le, Q. V.; and Manning, C. D. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *International Conference on Learning Representations*.

Cui, Y.; Che, W.; Liu, T.; Qin, B.; Wang, S.; and Hu, G. 2020. Revisiting Pre-Trained Models for Chinese Natural Language Processing. *arXiv preprint arXiv:2004.13922* .

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186.

Gu, J.-C.; Li, T.; Liu, Q.; Ling, Z.-H.; Su, Z.; Wei, S.; and Zhu, X. 2020. Speaker-Aware BERT for Multi-Turn Response Selection in Retrieval-Based Chatbots. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*.

Hu, B.; Lu, Z.; Li, H.; and Chen, Q. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*, 2042–2050.

Humeau, S.; Shuster, K.; Lachaux, M.-A.; and Weston, J. 2020. Poly-encoders: Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring. In *International Conference on Learning Representations*.

Jia, R.; and Liang, P. 2017. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2021–2031.

Joshi, M.; Chen, D.; Liu, Y.; Weld, D. S.; Zettlemoyer, L.; and Levy, O. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics* 8: 64–77.

Kadlec, R.; Schmid, M.; and Kleindienst, J. 2015. Improved deep learning baselines for ubuntu corpus dialogs. *arXiv preprint arXiv:1510.03753* .

Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; and Soricut, R. 2020. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In *International Conference on Learning Representations*.

Lee, D.; Shin, M. C.; Whang, T.; Cho, S.; Ko, B.; Lee, D.; Kim, E.; and Jo, J. 2020. Reference and Document Aware Semantic Evaluation Methods for Korean Language Summarization. In *Proceedings of the 28th International Conference on Computational Linguistics*, 5604–5616.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880.

Liang, W.; Zou, J.; and Yu, Z. 2020. Beyond User Self-Reported Likert Scale Ratings: A Comparison Model for Automatic Dialog Evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1363–1374.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* .

Lowe, R.; Pow, N.; Serban, I.; and Pineau, J. 2015. The Ubuntu Dialogue Corpus: A Large Dataset for Research in Unstructured Multi-Turn Dialogue Systems. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 285–294.

Lu, J.; Ren, X.; Ren, Y.; Liu, A.; and Xu, Z. 2020. Improving Contextual Language Models for Response Retrieval in Multi-Turn Conversation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1805–1808.

Mehri, S.; Razumovskaia, E.; Zhao, T.; and Eskenazi, M. 2019. Pretraining Methods for Dialog Context Representation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3836–3845.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, 8026–8037.

Tao, C.; Wu, W.; Xu, C.; Hu, W.; Zhao, D.; and Yan, R. 2019a. Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 267–275.

Tao, C.; Wu, W.; Xu, C.; Hu, W.; Zhao, D.; and Yan, R. 2019b. One Time of Interaction May Not Be Enough: Go Deep with an Interaction-over-Interaction Network for Response Selection in Dialogues. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1–11.

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 *Proceedings of the Advances in Neural Information Processing Systems*, 5998–6008.

Wan, S.; Lan, Y.; Xu, J.; Guo, J.; Pang, L.; and Cheng, X. 2016. Match-SRNN: Modeling the Recursive Matching Structure with Spatial RNN. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 2922–2928.

Wang, A.; Pruksachatkun, Y.; Nangia, N.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Proceedings of the Advances in Neural Information Processing Systems*, 3266–3280.

Wang, H.; Lu, Z.; Li, H.; and Chen, E. 2013. A dataset for research on short-text conversations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 935–945.

Wang, H.; Wang, X.; Xiong, W.; Yu, M.; Guo, X.; Chang, S.; and Wang, W. Y. 2019b. Self-Supervised Learning for Contextualized Extractive Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2221–2227.

Wang, M.; Lu, Z.; Li, H.; and Liu, Q. 2015. Syntax-Based Deep Matching of Short Texts. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence*, 1354–1361.

Wang, S.; Che, W.; Liu, Q.; Qin, P.; Liu, T.; and Wang, W. Y. 2020. Multi-task self-supervised learning for disfluency detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 9193–9200.

Wang, S.; and Jiang, J. 2016. Learning Natural Language Inference with LSTM. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1442–1451.

Whang, T.; Lee, D.; Lee, C.; Yang, K.; Oh, D.; and Lim, H. 2020. An Effective Domain Adaptive Post-Training Method for BERT in Response Selection. In *Proc. Interspeech 2020*, 1585–1589.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; and Brew, J. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv preprint arXiv:1910.03771* .

Wu, J.; Wang, X.; and Wang, W. Y. 2019. Self-Supervised Dialogue Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3857–3867.

Wu, Y.; Wu, W.; Xing, C.; Zhou, M.; and Li, Z. 2017. Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 496–505.

Xu, H.; Liu, B.; Shu, L.; and Philip, S. Y. 2019. BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2324–2335.

Xu, Z.; Liu, B.; Wang, B.; Sun, C.; and Wang, X. 2017. Incorporating loose-structured knowledge into conversation modeling via recall-gate LSTM. In *2017 International Joint Conference on Neural Networks (IJCNN)*, 3506–3513.

Yan, R.; Song, Y.; and Wu, H. 2016. Learning to Respond with Deep Neural Networks for Retrieval-Based Human-Computer Conversation System. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 55–64.

Yuan, C.; Zhou, W.; Li, M.; Lv, S.; Zhu, F.; Han, J.; and Hu, S. 2019. Multi-hop Selector Network for Multi-turn Response Selection in Retrieval-based Chatbots. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 111–120.

Zhang, Z.; Li, J.; Zhu, P.; Zhao, H.; and Liu, G. 2018. Modeling Multi-turn Conversation with Deep Utterance Aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics*, 3740–3752.

Zhao, Y.; Xu, C.; and Wu, W. 2020. Learning a Simple and Effective Model for Multi-turn Response Generation with Auxiliary Tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3472–3483.

Zhou, X.; Dong, D.; Wu, H.; Zhao, S.; Yu, D.; Tian, H.; Liu, X.; and Yan, R. 2016. Multi-view Response Selection for Human-Computer Conversation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 372–381.

Zhou, X.; Li, L.; Dong, D.; Liu, Y.; Chen, Y.; Zhao, W. X.; Yu, D.; and Wu, H. 2018. Multi-Turn Response Selection for Chatbots with Deep Attention Matching Network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1118–1127.