Efficient Dynamic Hard Negative Sampling for Dialogue Selection

Janghoon Han **Dongkyu Lee** **Joongbo Shin** **Hyunkyung Bae**
Jeesoo Bang **Seonghwan Kim** **Stanley Jungkyu Choi** **Honglak Lee**
LG AI Research
{janghoon.han}@lgresearch.ai

Abstract

Recent studies have demonstrated significant improvements in selection tasks, and a considerable portion of this success is attributed to incorporating informative negative samples during training. While traditional methods for constructing hard negatives provide meaningful supervision, they depend on static samples that do not evolve during training, leading to sub-optimal performance. Dynamic hard negative sampling addresses this limitation by continuously adapting to the model's changing state throughout training. However, the high computational demands of this method restrict its applicability to certain model architectures. To overcome these challenges, we introduce an efficient dynamic hard negative sampling (EDHNS). EDHNS enhances efficiency by pre-filtering easily discriminable negatives, thereby reducing the number of candidates the model needs to compute during training. Additionally, it excludes question-candidate pairs where the model already exhibits high confidence from loss computations, further reducing training time. These approaches maintain learning quality while minimizing computation and streamlining the training process. Extensive experiments on DSTC9, DSTC10, Ubuntu, and E-commerce benchmarks demonstrate that EDHNS significantly outperforms baseline models, proving its effectiveness in dialogue selection tasks.[1]

1 Introduction

The problem of selecting the most suitable answer from multiple candidates has been extensively explored in the field of natural language processing, particularly within selection tasks (Lowe et al., 2015; Wu et al., 2018a; Zhang et al., 2018a; Kim et al., 2020, 2021). Typically, these tasks involve one positive candidate and multiple negative candidates associated with a given question. Training on all negative samples can be time-consuming, so it is common practice to randomly select a subset of negative samples for training. However, random negative sampling may not provide meaningful supervision, as models updated with easily discriminable negative samples contribute minimally to gradient updates (Cai et al., 2020; Xu et al., 2022a).

To address this issue, various strategies for hard negative sampling have been proposed and have demonstrated their effectiveness (He et al., 2021; Mi et al., 2021; Tang et al., 2021). Heuristic and data-dependent methods (He et al., 2021; Mi et al., 2021) utilize the unique characteristics of datasets but are constrained by their limited generalizability, making them less effective for other datasets. Lin et al. (2020); Tang et al. (2021) have enhanced these approaches with model-based strategies. However, these approaches still face challenges, as they rely on static (fixed) hard negative samples that do not dynamically adapt during training.

Recently, dynamic hard negative sampling (Xiong et al., 2021; Zhan et al., 2021) has been introduced to overcome these limitations by adaptively selecting hard negatives for learning in response to model updates, effectively aligning with changes in model behavior. However, it requires continual recalculations of matching scores for all negative candidates throughout training, significantly increasing computational costs. This restriction predominantly confines its application to fast dense retrieval models (Karpukhin et al., 2020; Gao and Callan, 2021, 2022), and poses implementation challenges in models with slower inference speeds.

To mitigate these challenges, we propose an Efficient Dynamic Hard Negative Sampling (EDHNS) method applicable to various model architectures. Like traditional approaches, our method computes matching scores for negative candidates at each training step. However, it alleviates the computational burden through two main strategies: shortlisting and selective update. In shortlisting, we compute scores only for a filtered subset of candidates

[1] https://github.com/hanjanghoon/EDHNS

Proceedings of the 6th Workshop on NLP for Conversational AI (NLP4ConvAI 2024), pages 89–100

by removing easily discriminable negative candidates from the pool, enabling the selection of sufficiently hard negatives from a smaller set. In the selective update, we measure confidence scores for question-candidate pairs and exclude those with high scores from training, further save training time. These strategies enable meaningful learning with reduced computational demands, enhancing overall performance. Notably, for the first time, we have applied dynamic hard negative sampling to the cross encoder, which has demonstrated strong performance in selection tasks, leading to significant performance improvements.

We empirically demonstrate the efficacy of our method through extensive experiments on two key tasks. The first task, knowledge selection, focuses on choosing relevant knowledge for a given conversation. We evaluate the performance of this task using the DSTC9 (Kim et al., 2020) and DSTC10 (Kim et al., 2021) benchmarks. The second task, response selection, requires choosing the most appropriate response for a given dialogue context. We assess this task using the Ubuntu (Lowe et al., 2015) and E-commerce (Zhang et al., 2018a) benchmarks. Our experiments show that models using EDHNS significantly outperform baseline models across all four benchmarks. Specifically, EDHNS achieves top performance in most evaluation metrics for DSTC9 and DSTC10, and also demonstrates superior performance in the Ubuntu and E-commerce benchmarks.

2 Related Work

Previous studies have introduced various hard negative sampling approaches, resulting in notable enhancements in various NLP tasks. These strategies can be categorized into two types: static hard negative sampling and dynamic hard negative sampling (Zhan et al., 2021; Xu et al., 2022b).

Static hard negative sampling pre-defines fixed hard negative samples before the training process. This method selects hard negative samples based on data characteristics or by retrieving or generating them using a model. In the knowledge selection task, He et al. (2021) introduce a data-dependent negative sampling strategy by categorizing given knowledge into different groups. Tang et al. (2021) adopt a model-based negative sampling method to sample fixed hard negatives. In the response selection task, Lin et al. (2020) use retrieval and generation models to diversify negative samples, while Lee et al. (2022b) generate adversarial examples using GPT-3. In text retrieval tasks, since negative samples are derived from text candidates recalled by the retrieval module, previous works (Ren et al., 2021; Zhang et al., 2022a) focus on jointly optimizing the retriever and reranker modules.

Dynamic hard negative sampling, in contrast, selects hard negative samples dynamically during the training process, considering the evolving state of the model. In response selection, Li et al. (2019) adapt negative examples to matching models during the learning process, exploring various sampling strategies. Particularly, this approach has been extensively studied in the training of dense retrieval models. Guu et al. (2020) and Xiong et al. (2021) use dense retrieval models to pre-retrieve the top documents as hard negatives during training, periodically rebuilding the index and refreshing the hard negatives. Zhan et al. (2021) propose a query-side training algorithm that directly optimizes the dense retrieval model using dynamic hard negative sampling.

However, applying dynamic hard negative sampling to most model architectures—except for the bi-encoder structure commonly used in dense retrieval—poses challenges due to the slower speeds and high computational demands. This limitation is especially evident in cross-encoder-based models, which, despite their superior performance in selection tasks, require extensive computations for token-level interactions and cannot pre-compute candidate embeddings. To overcome these challenges, we propose a novel and efficient dynamic hard negative sampling method.

3 Preliminary

3.1 Problem Formalization of Selection Task

Let dataset $\mathbf{D} = \{(q_i, \mathbf{C}_i)\}_{i=1}^{M}$ be a set of M pairs that consist of a question q_i, its corresponding candidates $\mathbf{C}_i = \{p_i\} \cup \mathbf{N}_i^L$. A candidate pool $\mathbf{C}_i$ contains a positive candidate p_i and negative candidates $\mathbf{N}_i^L = \{n_{i,1}, n_{i,2}, ..., n_{i,L}\}$, where L is the number of negative candidates. As we address selection tasks as a unified framework for learning a matching model that evaluates relevance scores between a question and its candidates, the task is formulated as learning a matching function $f(q_i, c_{i,j})$ for a given question-candidate pair $(q_i, c_{i,j})$, where $c_{i,j} \in \mathbf{C}_i$.

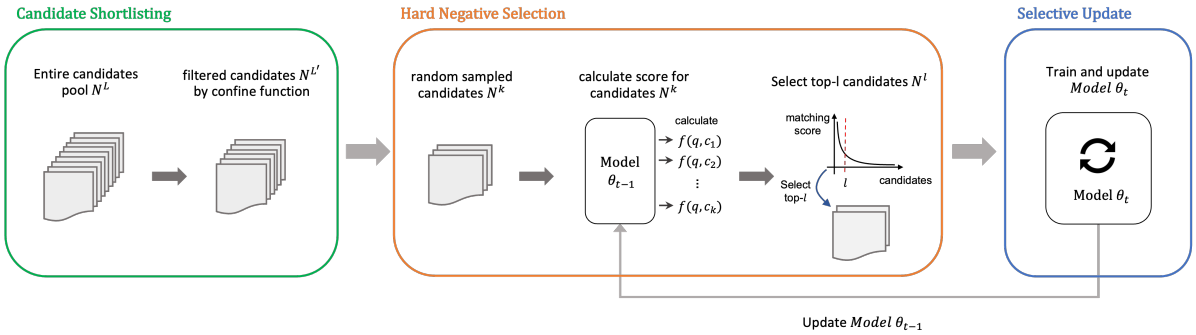

Figure 1: Efficient dynamic hard negative sampling framework. The EDHNS method comprises three key components: Candidate Shortlisting, Hard Negative Selection, and Selective Update.

3.2 Cross Encoder Architecture in Dialogue Selection

Following previous works (Nogueira and Cho, 2019; He et al., 2021; Han et al., 2021; Kim and Ko, 2021) in selection task, we use pre-trained bidirectional language models (Devlin et al., 2019; Liu et al., 2019; He et al., 2020) as a cross encoder to measure the matching degree between a question q_i and a candidate $c_{i,j}$. The input x of our matching model is as follows:

$$x = [\texttt{CLS}]\ q_i\ [\texttt{SEP}]\ c_{i,j}\ [\texttt{SEP}]. \quad (1)$$

Token embedding for input x are summed with position embedding and segment embedding to become input representations. The input representations are fed into the transformer layer, and the self-attention module in the transformer layer computes cross-attention between those of q_i and $c_{i,j}$. In this way, multiple transformer layers can deeply understand the relevance of the question and its candidate, resulting in a high-performance matching model. We use the final representation $o_{cls} \in \mathbb{R}^d$ of the [CLS] token for computing the matching score through an MLP layer:

$$f(q_i, c_{i,j}) = W_2\sigma(W_1 o_{cls} + b_1) + b_2, \quad (2)$$

where $W_1 \in \mathbb{R}^{d_h \times d}$, $W_2 \in \mathbb{R}^{1 \times d_h}$, $b_1 \in \mathbb{R}^{d_h}$, and $b_2 \in \mathbb{R}^1$ are trainable parameters for fine-tuning. Eventually, the weights of the model are updated using the cross-entropy loss function:

$$\mathcal{L} = -\mathbb{E}_{(q_i,p_i,N_i)\sim D}[\log(\frac{e^{f(q_i,p_i)}}{e^{f(q_i,p_i)}+\sum_{j=1}^{l} e^{f(q_i,n_{i,j})}})] \quad (3)$$

where l is the number of negative samples and p_i, $n_{i,j}$ denote positive and negative candidates respectively.

4 Methodology

4.1 Efficient Dynamic Hard Negative Sampling

In this section, we explain the details of our efficient dynamic hard negative sampling (EDHNS) framework for selection tasks. As its name shows, we let the model find hard negative samples that are difficult to discriminate by itself during training. Figure 1 illustrate the process of EDHNS where the model iterates selecting hard negatives and learning to discriminate positive from them at each training step. Since hard negatives are collected at every model update, the selected samples can be the '*most challenging*' for the model at that time. Therefore, the model could learn from more informative hard negatives, which leads to faster convergence and performance gain.

4.1.1 Training Procedure

The EDHNS framework can be generalized as Algorithm 1. We first train the base model θ with random negatives for the initial s step, ensuring the model is capable of selecting hard negatives. After initialization, we iteratively select hard negative samples and update the model with those selected samples. During the hard negative selection phase, we randomly sample a negative subset ($\mathbf{N}_i^k$) from the pool of negative samples. Subsequently, we compute matching scores between the question and the sampled k candidates using the current model θ_{t-1} at step t, as explained in Equation 2. Based on these matching scores, we select top-l hard negatives from $\mathbf{N}_i^k$. After hard negative selection, we update the model θ_t with the positive p_i and the top-l hard negatives $\mathbf{N}_i^l$ using Equation 3.

Algorithm 1 Efficient dynamic hard negative sampling

Input: Dataset with confined negatives candidate sets $D' = \{(q_i, p_i, N_i^L, N_i^{L'})\}_{i=1}^{M}$, Model parameter θ, Initializing step s

1. Initialize the model θ with random samples for p steps
Initialize θ
for train step $t = 1$ to s **do**
  Sample a batch B_t from D'
  for (q_i, p_i, N_i) in B_t **do**
    $N_i^l :=$ l samples randomly extracted from N_i^L
  end for
  Update the model θ_t with $\{(q_i, p_i, N_i^l)\}_{i=1}^{|B_t|}$ using Eq.3
end for

2. Train the model θ
for train step $t = s+1$, ... **do**
  Sample a batch B_t from D'
  for (q_i, p_i^+, N_i^m) in B_t **do**
    $N_i^k :=$ k random candidates sampled from $N_i^{L'}$
    $N_i^l :=$ top-l candidates of sorted list of N_i^k along the matching score computed from the model θ_{t-1} using Eq.2
  end for
  Update the model θ_t with $\{(q_i, p_i^+, N_i^l)\}_{i=1}^{|B_t|}$ using Eq.3
end for

4.2 Time Reduction Strategies in EDHNS

4.2.1 Candidates Shortlisting

Since calculating matching scores for all negative candidates is considerably time-consuming, a practical approach is to randomly sample negative candidate subset $\mathbf{N}^k$ from a pool of all negative samples $\mathbf{N}^L$ where $k \ll L$. However, there is a trade-off in choosing the size of the candidate subset $\mathbf{N}^k$. If the sample size k is not large enough, it may not include an adequate number of challenging negative samples. Conversely, if k is increased, the training time also substantially increases for score calculation.

To train the model effectively even with a small size of candidate subset, we construct a confined negative candidate set, denoted as $\mathbf{N}^{L'}$, where $L' \ll L$. This confined negative candidate set is created by filtering out easy negatives from the original negative candidates set ($\mathbf{N}^L$). When sampling a negative subset ($\mathbf{N}^k$) from the confined candidate set, the likelihood of including difficult samples increases even with a small number of k. This is because the easy negatives have already been filtered out during the construction of $\mathbf{N}^{L'}$. We configure a confined candidate set by finding negative samples relevant to both question and the positive as follows.

$$\mathbf{N}_i^{L'} = \{n_{i,j} \mid g(q_i \oplus p_i, n_{i,j}) > \tau\}, \quad (4)$$

where τ is a threshold, $n_{i,j} \in \mathbf{N}_i^L$ and $\oplus$ denotes concatenation.

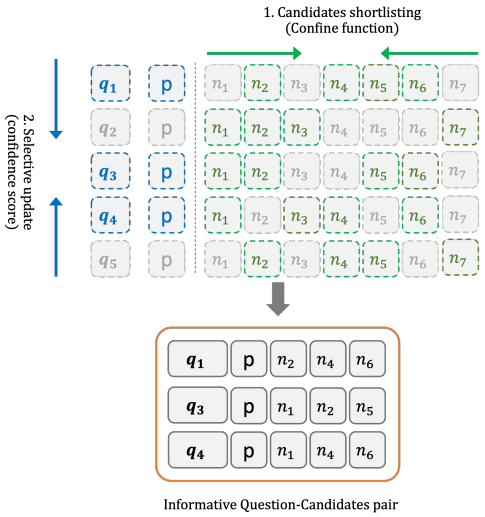

Figure 2: Time reduction strategies of EDHNS. Candidate shortlisting filters out easy negative candidates, and Selective update exclude well-known pairs from the training.

4.2.2 Selective Update

Another feature of EDHNS is its focused training solely on informative question-candidate pairs (q_i, C_i). This is achieved by calculating a confidence score for the positive sample during the negative selection process as follows.

$$Score(q_i, p_i, \mathbf{N}_i^k) = \frac{e^{f(q_i,p_i)}}{e^{f(q_i,p_i)} + \sum_{j=1}^{k} e^{f(q_i,n_{i,j})}} \quad (5)$$

where $n_{i,j} \in N_i^k$. If the confidence score exceeds a predefined threshold, the model considers it a well-known pair and excludes it from training and update. This strategy accelerates the training procedure by minimizing the inclusion of question-candidate pairs that do not contribute substantial supervision to the model and prevents the model from becoming overconfident (Lee et al., 2022a).

5 Experiments

5.1 Implementation Details

We train models with three different random seeds and report the average value for all experiments. Our model is trained with 8 NVIDIA A100 GPUs (with 40GB). For confine function g, we employ

Dataset	DSTC9 (Knowledge)			DSTC10 (Knowledge)			Ubuntu (Response)			E-commerce (Response)		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
# (q, C) pairs	19k	2673	1981	59k(syn)	104	683	500k	50k	50k	500k	5k	1k
# C per q	2900	2900	12039	9139	9139	9139	2(1000)	10	10	2(1000)	2(10)	10

Table 1: Data statistics for the knowledge selection, response selection benchmarks, q denotes question and C denotes candidates.

Sentence Transformers (Reimers and Gurevych, 2019)[2] and compute cosine similarity to measure semantic similarity. Details of each experimental setting can be found in Appendix A.

5.2 Knowledge Selection in Knowledge-grounded Dialogue System

One of the primary objectives in the ninth and tenth dialogue System Technology Challenge (DSTC9, DSTC10) is to develop a knowledge-grounded task-oriented dialogue system (Kim et al., 2020, 2021). The challenges consist of three consecutive subtasks: knowledge-seeking turn detection, knowledge selection, and knowledge-grounded response generation. Our focus is on the knowledge selection task, which requires the system to identify the most appropriate knowledge related to the user's last utterance.

Table 1 indicates the statistics for the DSTC9 and DSTC10 datasets. DSTC9 knowledge selection task includes out-of-domain knowledge in its test set. DSTC10 knowledge selection task involves speech recognition errors as it comprises spoken conversations. We sample synthetic data from prior studies (Tian et al., 2021; Han et al., 2022) and configure training data since there is no official training data for DSTC10.

The selection performance is assessed based on recall at k (*R@k*) and mean reciprocal rank (*MRR*) measures, specifically *R@1*, *R@5*, and *MRR@5*. These metrics are the official evaluation criteria for both DSTC9 and DSTC10 datasets (Kim et al., 2020, 2021).

5.2.1 Baseline Model

RoBERTa-base+EDHNS and *RoBERTa-large+EDHNS* are cross-encoder-based matching models trained with our efficient dynamic hard negative sampling. To evaluate the effectiveness of proposed methodology, we compare these models with *RoBERTa-base* and *RoBERTa-large*, which are identical architectures yet trained using random negative sampling. Additionally, we evaluate our approaches against other multiple baselines for the DSTC9 and DSTC10 tasks as follow.

DSTC9 Baseline *TF-IDF, BM25*, and *BERT-base* from Kim et al. (2020) are the official baselines for the DSTC9 competition. *TF-IDF* and *BM25* are bag-of-words information retrieval baselines and BERT-base utilizes a cross-encoder architecture for selection. *Knover* from He et al. (2021) applies a heuristic data-dependent hard negative sampling called multi-scale negative sampling. *Hirachical-filtering* from Jin et al. (2021) selects knowledge through three modules: domain classification, entity tracking, and knowledge matching. *Hirachical-selection* (Thulke et al., 2023) trains two different models which determines related domains and entities, and measures the relevance score of knowledge.

DSTC10 Baseline *DSTC9-BERT-base* and *DSTC9-Knover* are the official baseline models for the DSTC10 knowledge selection task (Kim et al., 2021), which are trained using the DSTC9 dataset. *Weighted* (Han et al., 2022) trains model utilizing weighted negative sampling, where different weight probabilities are assigned to each negative sample category. *Hirachical-selection+ABS* (Thulke et al., 2023) incorporates an Alternative Beam Search method into the hierarchical selection. *TOD_DA* (Tian et al., 2021) employs Data Augmentation and multi-scale negative sampling to enhance model's performance.

5.2.2 Result

Table 2 shows the performance of EDHNS approach in DSTC9 and DSTC10 benchmarks. The result highlights changing the negative sampling method to EDHNS in both the base and large models led to significant improvements in performance for both datasets. Specifically, The base model and the large model exhibit a consistent enhancement of 4.7% and 3.2%, respectively, in *R@1* on the DSTC9. Similarly, these models demonstrate significant 6.4% and 4.5% enhancements in *R@1* on the DSTC10. These improvements indicate the effectiveness of learning informative negative sam-

[2]We employ a model for confine function from Sentence-Transformers: https://www.sbert.net/

Method	PLM	$R@1$	$R@5$	$MRR@5$
Knowledge selection in DSTC9				
TF-IDF (Kim et al., 2020)	-	0.511	0.807	0.618
BM25 (Kim et al., 2020)	-	0.498	0.827	0.611
BERT-base (Kim et al., 2020)	BERT_{base}	0.834	0.976	0.891
Knover (He et al., 2021)	PLATO-2 (1.6B)	0.910	0.986	0.945
Hirachical-Filtering (Jin et al., 2021)	RoBERTa_{large}	0.925	0.970	<u>0.946</u>
Hirachical-Selection (Thulke et al., 2023)	RoBERTa_{large}	**0.932**	0.973	-
RoBERTa-base	RoBERTa_{base}	0.839	0.989	0.904
RoBERTa-base+EDHNS	RoBERTa_{base}	0.886	0.993	0.935
RoBERTa-large	RoBERTa_{large}	0.899	<u>0.995</u>	0.942
RoBERTa-large+EDHNS	RoBERTa_{large}	<u>0.931</u>	**0.998**	**0.962**
Knowledge selection in DSTC10				
DSTC9-BERT-base (Kim et al., 2021)	BERT_{base}	0.521	0.733	0.606
DSTC9-Knover (Kim et al., 2021)	PLATO-2 (1.6B)	0.619	0.800	0.693
TOD-DA (Tian et al., 2021)	PLATO-2 (1.6B)	<u>0.801</u>	**0.94**	<u>0.857</u>
Weighted (Han et al., 2022)	RoBERTa_{base}	0.72	0.862	0.780
Hirachical-Selection+ABS (Thulke et al., 2023)	RoBERTa_{large}	0.777	-	-
RoBERTa-base+MLM	RoBERTa_{base}	0.727	0.897	0.798
RoBERTa-base+MLM+EDHNS	RoBERTa_{base}	0.791	0.910	0.841
RoBERTa-large+MLM	RoBERTa_{large}	0.776	0.930	0.838
RoBERTa-large+MLM+EDHNS	RoBERTa_{large}	**0.821**	<u>0.935</u>	**0.869**

Table 2: Test set performance of knowledge selection in DSTC9 and DSTC10. The best and second-best results are in bold and underlined fonts respectively. For the DSTC10 dataset, since spoken errors are present, masked language modeling is applied for robust token representation.

ples from a model perspective. In addition to the substantial performance improvement compared to their base model, the proposed models outperform other baselines on both datasets. In comparison to the state-of-the-art model in DSTC9, *hierarchical* selection, our *RoBERTa-large+EDHNS* demonstrates shows a significant enhancement of 2.5% in *R@5*. In the DSTC10 dataset, the *RoBERTa-large+MLM+EDHNS* model outperforms the state-of-the-art TOD-DA by 2% in *R@1*.

5.3 Response Selection in Retrieval-based Dialogue Systems

Response selection is a task in retrieval-based dialogue systems where the goal is to select the appropriate response from given response candidates based on the provided dialogue context. We validate the effectiveness of EDHNS using commonly used benchmarks for this task, namely the Ubuntu Corpus and the E-commerce Corpus.

The Ubuntu Corpus V1 (Lowe et al., 2015) is a dataset consisting of multi-turn dialogues extracted from Ubuntu chat logs. It primarily contains technical-support conversations about Ubuntu problems. For this study, we utilize the preprocessed data provided by Xu et al. (2017). The E-commerce Corpus (Zhang et al., 2018a) is a Chinese multi-turn dialogue dataset collected from Taobao, China's largest e-commerce platform. It includes authentic interactions between customers and customer service representatives, covering various conversational topics such as consultations and product recommendations.

Since the original training set contains only one negative candidate per dialogue context, we augment the negative candidates by sampling 1k utterances from 1 million other response candidates for both benchmarks, as shown in Table 1. Additionally, we augmented the validation set of the E-commerce corpus in a similar manner to reduce discrepancies with the test set.

The response selection performance for both the Ubuntu Corpus and the E-commerce Corpus is evaluated using $R_{10}@1$, $R_{10}@2$, and $R_{10}@5$, following previous work (Gu et al., 2020; Xu et al., 2021; Han et al., 2021).

5.3.1 Baseline Model

BERT (Gu et al., 2020) is a BERT-based (Devlin et al., 2019) cross encoder matching model. *UMS_bert+* (Whang et al., 2021) and *BERT_SL* (Xu et al., 2021) jointly train a PLM-based response selection model with other self-supervised tasks to learn temporal dependencies between ut-

Models	Ubuntu			E-commerce		
	R_{10}@1	R_{10}@2	R_{10}@5	R_{10}@1	R_{10}@2	R_{10}@5
TF-IDF (Lowe et al., 2015)	0.410	0.545	0.708	0.159	0.256	0.477
RNN (Lowe et al., 2015)	0.403	0.547	0.819	0.118	0.223	0.589
CNN (Kadlec et al., 2015)	0.549	0.684	0.896	0.328	0.515	0.792
LSTM (Kadlec et al., 2015)	0.638	0.784	0.949	0.365	0.536	0.828
SMN (Wu et al., 2018b)	0.726	0.847	0.961	0.453	0.654	0.886
DUA (Zhang et al., 2018b)	0.752	0.868	0.962	0.501	0.700	0.921
DAM (Zhou et al., 2018)	0.767	0.874	0.969	0.526	0.727	0.933
IOI (Tao et al., 2019)	0.796	0.894	0.974	0.563	0.768	0.950
ESIM (Chen and Wang, 2019)	0.796	0.894	0.975	0.570	0.767	0.948
MSN (Yuan et al., 2019)	0.800	0.899	0.978	0.606	0.770	0.937
BERT (Gu et al., 2020)	0.808	0.897	0.975	0.610	0.814	0.973
*BERT-VFT (Whang et al., 2020)	0.855	0.928	0.985	-	-	-
*SA-BERT (Gu et al., 2020)	0.855	0.928	0.983	0.704	0.879	0.985
*UMSBERT+ (Whang et al., 2021)	0.875	0.942	0.988	0.764	0.905	0.986
*BERT-SL (Xu et al., 2021)	0.884	0.946	0.990	0.776	0.919	0.991
*BERT-FP (Han et al., 2021)	0.911	0.962	0.994	0.870	0.956	0.993
*BERT-UMS+FGC (Li et al., 2022)	0.886	0.948	0.990	-	-	-
*Uni-Enc+BERT-FP (Song et al., 2023)	0.916	**0.965**	**0.994**	-	-	-
BERT+EDHNS	0.837	0.910	0.975	0.868	0.938	0.991
*BERT-FP+EDHNS	**0.917**	**0.965**	**0.994**	**0.957**	**0.986**	**0.997**

Table 3: Test set performance of response selection in Ubuntu and E-commerce corpus. All baseline models employ BERT_{base} as their PLM. The models marked with * have been post-trained.

terances. *BERT-FP* (Han et al., 2021) proposes a Fine-grained Post-training method that post-trains the short context response pair before fine-tuning. *BERT-UMS+FGC* (Li et al., 2022) is model that train *UMS_bert+* in Fine-Grained Contrastive learning manner. *Uni-Enc+BERT-FP* (Song et al., 2023) apply Uni-encoder architecture to advanced post-training model from Han et al. (2021). *BERT+EDHNS* and *BERT-FP+EDHNS* are proposed models that apply efficient dynamic negative sampling to the *BERT* and *BERT-FP*, respectively.

5.3.2 Result

As illustrated in Table 3, the application of EDHNS significantly enhances model performance in response selection tasks across different benchmarks. In the Ubuntu benchmark, *BERT+EDHNS* shows a significant improvement of 2.9% in *R@1* compared to its baseline model *BERT*, while *BERT-FP+EDHNS* achieves an enhancement of 0.6% in *R@1* over its baseline *BERT-FP*. In the E-commerce benchmark, the performance enhancements are even more pronounced. Specifically, *BERT+EDHNS* and *BERT-FP+EDHNS* demonstrate performance improvements of 25.8% and 8.7% in *R@1*, respectively, when compared to their corresponding baselines *BERT* and *BERT-FP*.

Method	$R@1$	$R@5$	$MRR@5$
RoBERTa +Random	0.899	0.995	0.942
RoBERTa +Static_model	0.906	0.997	0.947
RoBERTa +BM25	0.910	0.994	0.948
RoBERTa +Multi-scale	0.911	0.992	0.947
RoBERTa +**EDHNS**	**0.931**	**0.998**	**0.962**

Table 4: Comparison of efficient dynamic hard negative sampling with diverse hard negative sampling in DSTC9 test set using *RoBERTa-large*.

6 Further Analysis

6.1 Comparison of EDHNS with Other Negative Sampling Methods

We compared EDHNS with various other hard negative sampling approaches on DSTC9 test set as shown in Table 4. *RoBERTa+Random* is cross encoder matching model with random negative sampling. *+Static_model* refers to static hard negative sampling, where the model selects fixed hard negatives. *+BM25*, denote obtains hard negatives through the BM25 algorithm (Yang et al., 2017). *+Multi-scale* indicates multi-scale hard negative sampling proposed by He et al. (2021).

All the hard negative sampling methods lead to performance improvements compared to *RoBERTa+Random*. However, proposed *+EDHNS* method surpasses all other hard negative sampling

Model Variant	Training Time	Acc
Random	10m	0.926
DHNS(k=100)	1h 16m	0.967
DHNS(k=10)	17m	0.940
DHNS(k=10)+CS	16m	0.967
EDHNS: DHNS(k=10)+CS+SU	8m	0.964

Table 5: Ablation study for time reduction strategy on DSTC9 validation set using *RoBERTa-large*. *CS, HNS, SU* denote Candidate Shortlisting, Hard Negative Selection, and Selective Update of EDHNS in Figure 1. Each model is trained for five epochs.

techniques by a significant margin. This demonstrates that dynamically selecting hard negative from a model standpoint is superior in finding informative negative samples which enhance model performance.

6.2 Ablation Study about Time Reduction Strategies in EDHNS

We investigated the efficacy of the time reduction methods in EDHNS through a series of ablation experiments on the DSTC9 validation set, as shown in Table 5. *CS, HNS, SU* denote three main components of EDHNS: Candidate Shortlisting, Hard Negative Selection, and Selective Update, as shown in Figure 1. k represents the number of candidates for which the model measures the matching scores during the *HNS* phase.

Models with a hard negative selection exhibit notable performance improvement compared to previous random negative sampling. However, when k is large, such as *HNS(k=100)*, the training time significantly increases. Conversely, when the k is small, as in *HNS(k=10)*, the training time is reduced, but the performance is likewise diminished. The model with the shortlisting phase *CS+HNS (k=10)* maintain a similar training speed to *HNS(k=10)* while achieving comparable performance to *HNS(k=100)*. This observation underscores that model can sufficiently select informative hard negatives with a small number of k by removing easy negative samples from the negative pool through shortlisting. Moreover, when compared *CS+HNS(k=10)* to complete EDHNS (*CS+HNS(k=10)+SU*) including the selective update phase reduces the training time by less than half while still exhibiting comparable performance. This result demonstrates excluding the training of overconfident pairs improves training efficiency without compromising model performance.

Conclusion

This study introduces a fast and efficient dynamic hard negative sampling method for selection tasks. We overcome the constraints of previous dynamic hard negative sampling methods by enhancing their efficiency, thereby enabling their application across various model architectures. Our approach includes two time-saving strategies: candidate shortlisting to filter out easy negative candidates and selective updates to focus on meaningful question-candidate pairs for learning. Through this, the model dynamically and efficiently learns from challenging negative samples, effectively gaining valuable supervision. Specifically, we apply this methodology to a cross-encoder architecture, demonstrating its effectiveness and generalizability in dialogue selection across two tasks and four benchmarks. Experimental results show that models with EDHNS consistently outperform their baseline models across all benchmarks, highlighting the effectiveness of the proposed approach.

Limitation

Although EDHNS accelerates learning by providing informative samples to the model, there are also limitations. One potential limitation is a false negative problem, a common problem in hard negative sampling. For instance, false negatives (i.e., unlabeled positives) may exist in the MS MARCO dataset since the annotators can only annotate a few top-retrieved passages (Qu et al., 2021). If these false negatives are mistakenly considered true negatives during the training process, it may disturb the model to correctly discriminate between positive and negative instances.

References

Tiffany Tianhui Cai, Jonathan Frankle, David J. Schwab, and Ari S. Morcos. 2020. Are all negatives created equal in contrastive instance discrimination? *CoRR*, abs/2010.06682.

Qian Chen and Wen Wang. 2019. Sequential attention-based network for noetic end-to-end response selection.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages

4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Luyu Gao and Jamie Callan. 2021. Condenser: a pre-training architecture for dense retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 981–993, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Luyu Gao and Jamie Callan. 2022. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2843–2853, Dublin, Ireland. Association for Computational Linguistics.

Jia-Chen Gu, Tianda Li, Quan Liu, Zhen-Hua Ling, Zhiming Su, Si Wei, and Xiaodan Zhu. 2020. Speaker-aware BERT for multi-turn response selection in retrieval-based chatbots. In *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 2041–2044. ACM.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Janghoon Han, Taesuk Hong, Byoungjae Kim, Youngjoong Ko, and Jungyun Seo. 2021. Fine-grained post-training for improving retrieval-based dialogue systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1549–1558, Online. Association for Computational Linguistics.

Janghoon Han, Joongbo Shin, Hosung Song, Hyunjik Jo, Gyeonghun Kim, Yireun Kim, and Stanley Jungkyu Choi. 2022. External knowledge selection with weighted negative sampling in knowledge-grounded task-oriented dialogue systems.

Huang He, Hua Lu, Siqi Bao, Fan Wang, Hua Wu, Zhengyu Niu, and Haifeng Wang. 2021. Learning to select external knowledge with multi-scale negative sampling. *arXiv preprint arXiv:2102.02096*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention.

Di Jin, Seokhwan Kim, and Dilek Hakkani-Tur. 2021. Can I be of further assistance? using unstructured knowledge access to improve task-oriented conversational modeling. In *Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering (DialDoc 2021)*, pages 119–127, Online. Association for Computational Linguistics.

Rudolf Kadlec, Martin Schmid, and Jan Kleindienst. 2015. Improved deep learning baselines for ubuntu corpus dialogs.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Meoungjun Kim and Youngjoong Ko. 2021. *Self-Supervised Fine-Tuning for Efficient Passage Re-Ranking*, page 3142–3146. Association for Computing Machinery, New York, NY, USA.

Seokhwan Kim, Mihail Eric, Karthik Gopalakrishnan, Behnam Hedayatnia, Yang Liu, and Dilek Hakkani-Tur. 2020. Beyond domain apis: Task-oriented conversational modeling with unstructured knowledge access. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 278–289.

Seokhwan Kim, Yang Liu, Di Jin, Alexandros Papangelis, Karthik Gopalakrishnan, Behnam Hedayatnia, and Dilek Hakkani-Tür. 2021. "how robust ru?": Evaluating task-oriented dialogue systems on spoken conversations. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 1147–1154. IEEE.

Dongkyu Lee, Zhiliang Tian, Yingxiu Zhao, Ka Chun Cheung, and Nevin L. Zhang. 2022a. Hard gate knowledge distillation – leverage calibration for robust and reliable language model.

Nyoungwoo Lee, ChaeHun Park, Ho-Jin Choi, and Jaegul Choo. 2022b. Pneg: Prompt-based negative response generation for dialogue response selection task.

Jia Li, Chongyang Tao, Wei Wu, Yansong Feng, Dongyan Zhao, and Rui Yan. 2019. Sampling matters! an empirical study of negative sampling strategies for learning of matching models in retrieval-based dialogue systems. 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)*, pages 1291–1296, Hong Kong, China. Association for Computational Linguistics.

Yuntao Li, Can Xu, Huang Hu, Lei Sha, Yan Zhang, and Daxin Jiang. 2022. Small changes make big differences: Improving multi-turn response selection in dialogue systems via fine-grained contrastive learning. In *Interspeech 2022, 23rd Annual Conference of the International Speech Communication Association, Incheon, Korea, 18-22 September 2022*, pages 2723–2727. ISCA.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Frassetto

Nogueira. 2021. Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 2356–2362. ACM.

Zibo Lin, Deng Cai, Yan Wang, Xiaojiang Liu, Haitao Zheng, and Shuming Shi. 2020. The world is not binary: Learning to rank with grayscale data for dialogue response selection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9220–9229, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhenghao Liu, Kaitao Zhang, Chenyan Xiong, Zhiyuan Liu, and Maosong Sun. 2021. Openmatch: An open source library for neu-ir research. In *Proceedings of SIGIR*, page 2531–2535.

Ryan Lowe, Nissan Pow, Iulian Vlad Serban, and Joelle Pineau. 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*, pages 285–294.

Haitao Mi, Qiyu Ren, Yinpei Dai, Yifan He, Jian Sun, Yongbin Li, Jing Zheng, and Peng Xu. 2021. Towards generalized models for beyond domain api task-oriented dialogue. In *AAAI-21 DSTC9 Workshop*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. In *CoCo@ NIPs*.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *ArXiv*, abs/1901.04085.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-stage document ranking with bert.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5835–5847, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. Rocketqav2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 2825–2835. Association for Computational Linguistics.

Chiyu Song, Hongliang He, Haofei Yu, Pengfei Fang, Leyang Cui, and Zhenzhong Lan. 2023. Uni-encoder: A fast and accurate response selection paradigm for generation-based dialogue systems. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 6231–6244. Association for Computational Linguistics.

L Tang, Q Shang, K Lv, Z Fu, S Zhang, C Huang, and Z Zhang. 2021. Radge: Relevance learning and generation evaluating method for task-oriented conversational systems. In *AAAI 2021, Workshop on DSTC9*, volume 7.

Chongyang Tao, Wei Wu, Can Xu, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. 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*, pages 1–11, Florence, Italy. Association for Computational Linguistics.

David Thulke, Nico Daheim, Christian Dugast, and Hermann Ney. 2023. Task-oriented document-grounded dialog systems by hltpr@rwth for dstc9 and dstc10. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pages 1–10.

Xin Tian, Xinxian Huang, Dongfeng He, Yingzhan Lin, Siqi Bao, Huang He, Liankai Huang, Qiang Ju, Xiyuan Zhang, Jian Xie, Shuqi Sun, Fan Wang, Hua Wu, and Haifeng Wang. 2021. Tod-da: Towards boosting the robustness of task-oriented dialogue modeling on spoken conversations.

Taesun Whang, Dongyub Lee, Chanhee Lee, Kisu Yang, Dongsuk Oh, and Heuiseok Lim. 2020. An effective domain adaptive post-training method for BERT in response selection. In *Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020*, pages 1585–1589. ISCA.

Taesun Whang, Dongyub Lee, Dongsuk Oh, Chanhee Lee, Kijong Han, Dong-hun Lee, and Saebyeok Lee. 2021. Do response selection models really know what's next? utterance manipulation strategies for multi-turn response selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Xianchao Wu, Ander Martínez, and Momo Klyen. 2018a. Dialog generation using multi-turn reasoning neural networks. In *Proceedings of the 2018 Conference of the North American Chapter of the*

Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 2049–2059, New Orleans, Louisiana. Association for Computational Linguistics.

Xianchao Wu, Ander Martínez, and Momo Klyen. 2018b. Dialog generation using multi-turn reasoning neural networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2049–2059, New Orleans, Louisiana. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Lanling Xu, Jianxun Lian, Wayne Xin Zhao, Ming Gong, Linjun Shou, Daxin Jiang, Xing Xie, and Ji-Rong Wen. 2022a. Negative sampling for contrastive representation learning: A review. *CoRR*, abs/2206.00212.

Lanling Xu, Jianxun Lian, Wayne Xin Zhao, Ming Gong, Linjun Shou, Daxin Jiang, Xing Xie, and Ji-Rong Wen. 2022b. Negative sampling for contrastive representation learning: A review.

Ruijian Xu, Chongyang Tao, Daxin Jiang, Xueliang Zhao, Dongyan Zhao, and Rui Yan. 2021. Learning an effective context-response matching model with self-supervised tasks for retrieval-based dialogues. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14158–14166.

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

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, page 1253–1256, New York, NY, USA. Association for Computing Machinery.

Chunyuan Yuan, Wei Zhou, Mingming Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. 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)*, pages 111–120, Hong Kong, China. Association for Computational Linguistics.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Optimizing dense retrieval model training with hard negatives. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 1503–1512, New York, NY, USA. Association for Computing Machinery.

Hang Zhang, Yeyun Gong, Yelong Shen, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2022a. Adversarial retriever-ranker for dense text retrieval. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Yanzhao Zhang, Dingkun Long, Guangwei Xu, and Pengjun Xie. 2022b. HLATR: enhance multi-stage text retrieval with hybrid list aware transformer reranking. *CoRR*, abs/2205.10569.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018a. Modeling multi-turn conversation with deep utterance aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3740–3752, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018b. Modeling multi-turn conversation with deep utterance aggregation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3740–3752, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 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)*, pages 1118–1127, Melbourne, Australia. Association for Computational Linguistics.

A Appendix

A.1 More Experimental Details

Table 6 shows our detailed hyperparameter for four benchmarks. For knowledge selection, we set the maximum question length and candidate length each. For the response selection task, we discard the front of context. This is because for response selection last utterance of context is more significant.

We set a threshold for shortlisting as shown in Table 6. Since the number of easy candidates under the threshold differs per query candidate pair, the number of confined candidates m differs. Therefore, m is the average number of confined candidates for all pairs.

Benchmark	max sequence length	shortlisting threshold	m confined candidates	k randomly sampled candidate	l negatives for training	learning rate	batch size	Multi-task	# of epochs	confidence score threshold
DSTC9 knowledge selection	512	0.45	150	10	3	5.00E-06	128	x	5	0.99
DSTC10 knowledge selection	512	0.45	600	50	3	5.00E-06	128	MLM	5	0.99
Ubuntu response selection	512	0.001	700	10	2	1.00E-05	128	x	5	0.99
E-commerce response selection	512	0.1	500	10	2	1.00E-05	128	x	5	0.99
MS MARCO passage reranking	512	0.3	500	30	3	5.00E-06	128	x	5	0.99

Table 6: Detailed model hyperparameter for five benchmarks.

We didn't apply the time reduction strategy for EDHNS in response selection on the Ubuntu corpus because when $k = 10$, speed is not that decreased.

A.2 Synthetic Training Data Construction for DSTC10

In the DSTC10 knowledge selection task, there is no official data. Therefore we reconstruct synthetic data from previous work (Tian et al., 2021; Han et al., 2022). Specifically, we sampled 32k pairs from (Tian et al., 2021), and created 27k pairs following the proposed method of (Han et al., 2022). Moreover, since spoken recognition errors exist in the DSTC10 dataset, we train the model in a multi-task learning manner with a masked language model to be robust to automatic speech recognition errors.

A.3 Passage Reranking in MS MARCO

To evaluate our method beyond the selection task, we employ the MS MARCO dataset for the reranking task. MS MARCO (Nguyen et al., 2016) dataset for passage ranking task consists of 1 million questions from Bing search query logs and 8.8 million candidate passages. Each query is labeled with relevant passages by human annotators. The passage ranking task in MS MARCO includes two subtasks: full-ranking and reranking. The full-ranking task aim to generate the top 1000 passages sorted by their relevance from the entire pool of 8.8 million passages, while the reranking task aim to rerank a given set of 1000 candidate passages already retrieved using the BM25 retriever (Yang et al., 2017). Comparing reranker modules directly in the full-ranking task is challenging due to variations in retriever performance. Therefore, we focus on reranking tasks with pre-retrieved 1000 passages using BM25 for more accurate assessments.[3] The performance of passage reranking was evaluated using *MRR@10* metric following previous work (Kim and Ko, 2021).

[3]We utilized officially provided 1000 candidate passages retrieved using the BM25 retriever for training from https://microsoft.github.io/msmarco/Datasets

Method	PLM	Retriever	$MRR@10$
BM25	-	BM25	0.167
BERT	BERT_{large}	BM25	0.365
Multi-stage	BERT_{large}	BM25	0.390
RoBERTa+WMLM	RoBERTa_{large}	BM25	0.389
RocketQAv2	ERNIE_{base}	BM25	<u>0.401</u>
HLATR-RoBERTa	RoBERTa_{large}	⋆BM25	0.368
RoBERTa	RoBERTa_{large}	BM25	0.386
RoBERTa+EDHNS	RoBERTa_{large}	BM25	**0.402**

Table 7: Development set performance of passage reranking task in MS MARCO. ⋆ indicate BM25 retrieval by the pyserini toolkit (Lin et al., 2021).

A.3.1 Baseline Model

BERT (Nogueira and Cho, 2019) and *RoBERTa* (Liu et al., 2021) are cross-encoder-based reranking models. *Multi-stage* (Nogueira et al., 2019) propose two stage reranking architecture which use two models for pointwise and pairwise classification. *RoBERTa+WMLM* (Kim and Ko, 2021) apply Weighted Masked Language Model in a multi-task learning manner. *ROcketQAv2* (Ren et al., 2021) propose novel joint training approach for dense passage retrieval module and passage reranking module. *HLATR-RoBERTa* (Zhang et al., 2022b) introduce Hybrid List Aware Transformer Reranking (HLATR) as a subsequent reranking module in two stage reranking manner. *RoBERTa+EDHNS* are cross-encoder-based reranking models trained with our efficient dynamic hard negative sampling.

A.3.2 Result

The results presented in Table 7 highlight the effectiveness of EDHNS in the passage reranking task of the MS MARCO dataset. Specifically, *RoBERTa+EDHNS* model achieves a significant improvement of 1.6% in $MRR@10$ compared to *RoBERTa* which train with random sampling. Moreover, our *RoBERTa+EDHNS* model outperform all previous baseline.