Reduce Catastrophic Forgetting of Dense Retrieval Training with Teleportation Negatives

Si Sun[1], **Chenyan Xiong**[2], **Yue Yu**[3], **Arnold Overwijk**[2], **Zhiyuan Liu**[4,5], **Jie Bao**[1]
[1]Dept. of Electron. Eng., Tsinghua University, Beijing, China
[2]Microsoft Research, Redmond, USA [3]Georgia Institute of Technology, Atlanta, USA
[4]Dept. of Comp. Sci. & Tech., Institute for AI, Tsinghua University, Beijing, China
[5]Beijing National Research Center for Information Science and Technology
s-sun17@mails.tsinghua.edu.cn; {chenyan.xiong, arnold.overwijk}@microsoft.com
yueyu@gatech.edu; {liuzy, bao}@tsinghua.edu.cn

Abstract

In this paper, we investigate the instability in the standard dense retrieval training, which iterates between model training and hard negative selection using the being-trained model. We show the catastrophic forgetting phenomena behind the training instability, where models learn and forget different negative groups during training iterations. We then propose ANCE-Tele, which accumulates momentum negatives from past iterations and approximates future iterations using lookahead negatives, as "teleportations" along the time axis to smooth the learning process. On web search and OpenQA, ANCE-Tele outperforms previous state-of-the-art systems of similar size, eliminates the dependency on sparse retrieval negatives, and is competitive among systems using significantly more (50x) parameters. Our analysis demonstrates that teleportation negatives reduce catastrophic forgetting and improve convergence speed for dense retrieval training. The source code of this paper is available at https://github.com/OpenMatch/ANCE-Tele.

1 Introduction

Dense retrieval (DR) learns to represent data into a continuous representation space and matches user information needs ("query") with target information ("document") via efficient nearest neighbor search (Huang et al., 2013; Lee et al., 2019). Recent research shows strong empirical advantages of dense retrieval in various information access scenarios, such as OpenQA (Karpukhin et al., 2020), web search (Xiong et al., 2021), and conversational search (Yu et al., 2021).

A unique challenge of dense retrieval is in the selection of training negatives (Karpukhin et al., 2020). For each query, dense retrieval models need to distinguish a few relevant documents from all other negative documents in the entire corpus, the latter often beyond the scale of millions, making negative sampling a necessity. At the same time, the nature of retrieval makes random negatives trivial and uninformative (Xiong et al., 2021), making effective negative sampling difficult. Recent research addressed this challenge using an iterative training approach: first training using negatives generated by sparse retrieval for a while, then refreshing the training negatives using the being-trained DR models themselves, and alternating the two phases till convergence (Xiong et al., 2021).

The iterative training-and-negative-refreshing approach yields strong results and became a standard in many DR systems (Hofstätter et al., 2021; Ren et al., 2021b; Gao and Callan, 2022, e.g.). However, the refresh of hard negatives may change the learning landscape too dramatically and cause optimization issues. Many found the little benefit of training more than one refresh (Oğuz et al., 2022; Gao and Callan, 2022) and the significant fluctuations in the model accuracy with more iterations due to training instability (Xiong et al., 2021).

In this paper, we aim to address the instability issue in dense retrieval training. We first conduct a deep investigation on the model behaviors during training and reveal a phenomenon behind the instability: catastrophic forgetting (Kirkpatrick et al., 2017). After training with refreshed negatives in one episode, the dense retriever's accuracy drops on a large fraction of *training queries* (20-30%) that have been learned earlier. Our analysis shows that negatives refreshed in different iterations may differ drastically, e.g., covering distinct distractions, and model training "swings" between these distractor groups, rather than capturing them all together.

With these observations, we develop ANCE-Tele, which upgrades ANCE (Xiong et al., 2021), a standard dense retrieval training strategy, using *teleportation negatives* with momentum and lookahead mechanisms. Specifically, momentum negatives record the negatives encountered in past iterations, while lookahead negatives use the neighbors of positive documents as surrogates to approximate the

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6639–6654
December 7-11, 2022

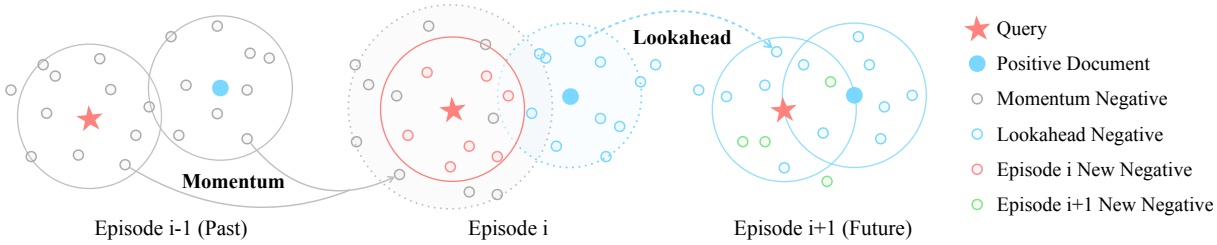


Figure 1: Illustration of constructing teleportation (momentum and lookahead) negatives in episode i of ANCE-Tele.

negatives appearing in the future. These teleportation negatives smooth out the training process of dense retrieval. Figure 1 illustrates the training process of ANCE-Tele.

In our experiments on web search and OpenQA, ANCE-Tele outperforms previous state-of-the-art systems of similar parameter sizes, without combining negatives from sparse retrieval or distillation from cross-encoder teachers. Simply including the teleportation negatives in training enables a BERT-base sized dense retriever (110 million parameters) to outperform the recent state-of-the-art model with five billion parameters.

Our analysis on the learning dynamics of ANCE-Tele confirmed the benefits of teleportation negatives. The momentum negatives from previous iterations reduce the catastrophic forgetting issue and improve learning stability, as they note the model to "remember" previously learned signals. The lookahead negatives sampled from the neighborhood of the positive documents provide an efficient forecast of future hard negatives and also improve the convergence speed. The two together improve the stability, efficiency, and effectiveness of dense retrieval training, while imposing zero GPU computation overhead.

After recapping related work, we investigate the instability issue of the iterative training process in Section 3 and present ANCE-Tele in Section 4. Experimental settings and evaluations are discussed in Section 5 and Section 6. We conclude in Section 7.

2 Related Work

The classic idea of learning representations for retrieval (Deerwester et al., 1990; Huang et al., 2013, e.g.) is recently revived with pretrained language models (Devlin et al., 2019). Lee et al. (2019) first presented the BERT-based dual-encoder dense retrieval formulation. Karpukhin et al. (2020) introduced BM25 negatives in DR training. Xiong et al. (2021) showed the necessity of hard negatives and introduced the iterative training-and-negative-refreshing process. These techniques formed the basic dense retrieval setup that achieves strong performances on a wide range of scenarios.

To improve dense retrieval training strategy is an active research front. Zhan et al. (2021) mitigated the risk of the delayed negative refresh by mining "real-time" hard negatives with a step-wise updating query encoder and a fixed document index. Hofstätter et al. (2021) observed that the iterations may lead to a fragile local optimal and balanced the negatives among query clusters. Qu et al. (2021) filtered the hard negatives using a stronger cross-encoder ranking model. With access to query-document term interactions, the ranking models are more powerful and can enhance DR training via knowledge distillation as well (Lin et al., 2021; Hofstätter et al., 2021; Ren et al., 2021b). Lewis et al. (2022) trained a series of dense retrievers through boosting and improved retrieval accuracy under approximate nearest neighbor search (ANN).

Recent research also identified several mismatches between pretrained models and DR. One mismatch is the locality of token level pretraining versus the needs of full sequence embeddings in DR (Lu et al., 2021), which can be reduced by enforcing an information bottleneck on the sequence embedding in pretraining (Wang et al., 2021; Gao and Callan, 2021). Another is the lack of alignment in the pretrained sequence representations, which can be improved using sequence contrastive pretraining (Meng et al., 2021; Gao and Callan, 2022). Pretrained models with billions or more parameters also improve DR accuracy, though the benefit of scaling in DR is less than observed in other tasks (Ni et al., 2021; Neelakantan et al., 2022).

DR often serves as the first stage retrieval in many language systems. Jointly training DR with later stage models can lead to better accuracy, with labels and signals from more sophisticated models in later stages, for example, question answering systems (Izacard and Grave, 2020; Zhao et al., 2021) and reranking models (Zhang et al., 2021).

3 Dense Retrieval Training Analysis

In this section, we first present the preliminaries of a standard dense retrieval setup and then investigate its training instability issues.

3.1 Preliminaries on Dense Retrieval

The first stage retrieval task is to find a set of relevant documents D^+ from a corpus C, for a given query q. The efficiency constraints often require the retrieval system to first represent query and documents independently into a vector space and then match them by efficient vector similarity metrics. Dense retrieval refers to methods that use a continuous dense vector space, in contrast to the discrete bag-of-word space used in sparse retrieval.

Bi-Encoder Model. A standard formulation of DR is to embed query and documents using dual/bi-encoders initialized from pretrained language models (Lee et al., 2019):

$$f(q, d; \theta) = g(q; \theta) \cdot g(d; \theta). \tag{1}$$

The encoder $g(\circ, \theta)$ embeds q and d using its parameter θ. The match uses dot product $(\cdot)$.

Retrieval in the embedding space with common similarity metrics, such as cosine, dot product, L2 distance, is supported by fast approximate nearest neighbor (ANN) search (Chen et al., 2018; Guo et al., 2020; Johnson et al., 2021). For example, we can retrieve the top K documents for a given q from the ANN index with high efficiency at a small cost of exactness:

$$\text{ANN}_{f(q,\circ;\theta)} = \text{Top K}^{\text{ANN}}_{d \in C} f(q, d; \theta). \tag{2}$$

Iterative Training. The training labels for retrieval are often provided as a set of relevant documents D^+ for each q, clicked web documents, passages containing the answer, etc. As the retrieval model needs to separate relevant documents d^+ from the entire corpus, all the rest corpus $C \backslash D^+$ are negatives, which are often too many to enumerate and require sampling.

Another nature of retrieval is that most irrelevant documents are trivial to distinguish. Only a few are challenging. It is unlikely for random sampling to hit these hard ones and produce informative training negatives. A widely used approach (Xiong et al., 2021; Ren et al., 2021b; Oguz et al., 2022, e.g.,) is to first train with hard negatives from sparse retrieval's top results, then use the trained DR model to refresh the hard negatives (self negative), and alternate through this training-and-negative-refreshing circle.

Without loss of generality, for a given training query q and its relevance documents D^+, one training iteration i in this training process is to find θ_i^* that minimizes the following loss:

$$\mathcal{L}_i = \sum_{q; d^+ \in D^+} \sum_{d^- \sim D_i^-} l(f(q, d^+; \theta_i), f(q, d^-; \theta_i));$$

$$D_i^- = \begin{cases} \text{ANN}_{f(q,\circ;\theta^*_{i-1})} \setminus D^+, & i > 1 \\ \text{BM25}(q, \circ) \setminus D^+ & i = 1. \end{cases} \tag{3}$$

The negatives D_i^- are constructed using model checkpoint from past episode or warmed up using sparse retrieval (BM25) (Xiong et al., 2021), where d^- are sampled uniformly without replacement. The model is trained to separate D_i^- from d^+ using the ranking loss l, e.g., cross entropy or hinge loss. For clarity, we refer to an iteration of negative construction and model training as one *training episode* in the rest of this paper, with episode 1 referring to the first fine-tuning iteration from the pretrained model.

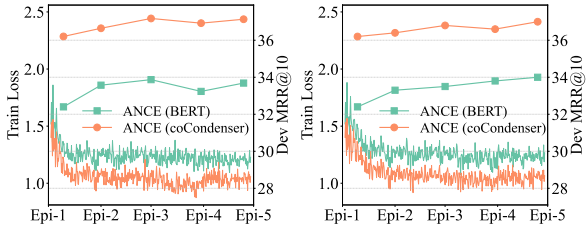
3.2 Learning Instability Investigation

We now investigate the training process described in Eqn. 3. We first study the training curves of various iterative training configurations, proceed with model behaviors, and then the dynamics of hard negative selection through training episodes.

Analysis Setting. We use MS MARCO passage retrieval benchmark (Bajaj et al., 2016) and the iterative training configurations from ANCE, starting from BM25 negatives and then self-mined negatives (Xiong et al., 2021). In addition, we conduct ANCE training from both vanilla BERT (Devlin et al., 2019) and coCondenser (Gao and Callan, 2022). The latter continues pretraining with sequence contrastive learning for dense retrieval.

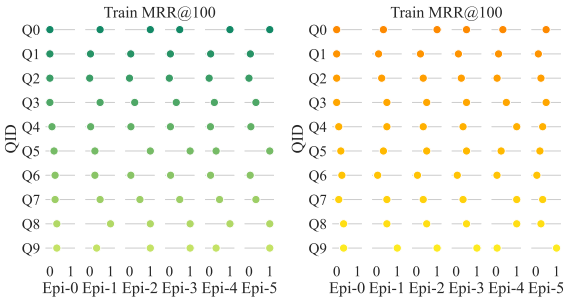
The iterative training can be viewed as a continual learning process (Mitchell et al., 2018), with each negative refresh forms a mini-task with new training signals. To better account for the training negative changes, we also experiment with cyclical learning rate scheduling (Smith, 2017), which warms up for each episode individually. We tune the hyperparameters to make the training as stable as possible. The detailed configurations in our analysis are listed in Appendix D.

Learning Curves. The training losses and development set accuracy (MRR@10) of ANCE variations are plotted in Figure 2. In standard ANCE,



(a) Standard ANCE. (b) ANCE with CyclicLR.

Figure 2: Training and testing curves in ANCE training episodes on MS MARCO. We use standard ANCE with linear decay in (a) and our tuned Cyclical LR in (b).



(a) ANCE (BERT). (b) ANCE (coCondenser).

Figure 3: Accuracy on ten random *training* queries during ANCE training with CyclicLR.

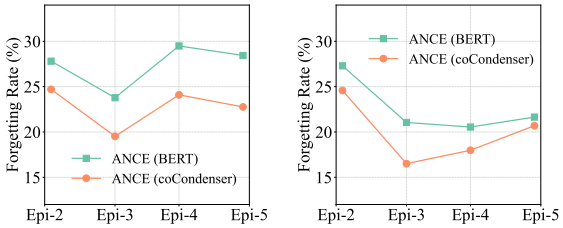
the training losses often increase in the beginning of episodes. This is designed as ANCE is to find hardest negatives for the model. However, it is undesired that the retrieval accuracy on Dev also fluctuates across the episodes. Using retrieval-oriented pretraining model, ANCE (coCondenser) elevates overall accuracy, but does not eliminate instability.

Our well-tuned cyclical learning rate makes the training smoother but at a notable cost. ANCE with CyclicLR only reached similar performance with standard ANCE after five episodes and its peak accuracy is slightly lower. Despite our best effort, the performance of ANCE (coCondenser), still drops at Epi-4 before it climbs back at Epi-5.

Catastrophic Forgetting. To understand this instability, we randomly sampled ten training queries and tracked their performances in Figure 3.

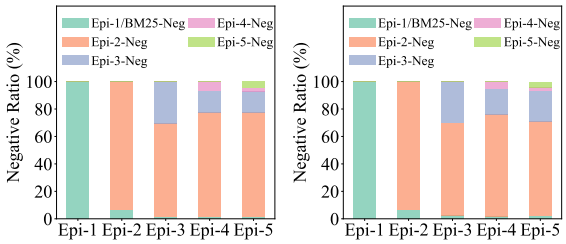
The examples indicate that the fluctuation in DR training is not due to overfitting, but catastrophic forgetting (Kirkpatrick et al., 2017), where models forget the examples they have learned in previous training steps, a common challenge in continual learning. The per query MRR jumps up and down during training and peaks at variant episodes. This shows the training instability problem is more severe than the averaged accuracy suggested.

In Figure 4, we plot the average forgetting ratio



(a) Standard ANCE. (b) ANCE with CyclicLR.

Figure 4: Fraction of training queries the models performed worse after one training episode, compared with their MRR@100 in the past episode.



(a) ANCE (BERT). (b) ANCE (coCondenser).

Figure 5: Composition of all negatives in each episode of ANCE (BERT/coCondenser) with CyclicLR.

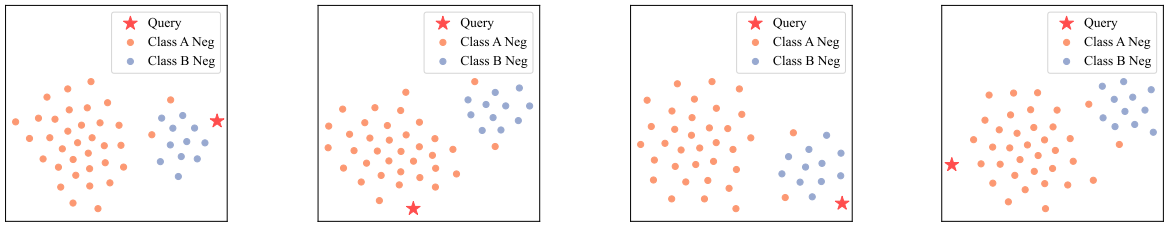
of ANCE variants on all MARCO training queries. With standard ANCE training, a model may forget a startling 20-30% of *training* queries after one hard negative refresh! At the cost of slower convergence, CyclicLR mitigates forgetting rate to around 20%, except at Epi-2, when ANCE switches from BM25 negatives to self-negatives.

Dynamics of Hard Negatives. We keep analyzing the forgetting issue by investigating the training negatives used in each ANCE episode—the biggest moving piece in iterative training. Figure 5 presents the composition of training negatives of two ANCE variants with CyclicLR. Results for more variants can be found in Appendix D.

The dynamics of negatives in Figure 5 reveal two behaviors of dense retrieval training: BM25 Warm Up Only and Negative Swing.

BM25 Warm Up Only. After the warm-up stage (Epi-1), BM25 Negatives are discarded quickly and at near entirety. This echos previous observation that dense retrieval disagrees significantly with sparse retrieval (Xiong et al., 2021). Negatives informative for one side are trivial for the other.

Negative Swing. The models swing between the negatives introduced at different episodes, rather than capturing all of them together. The self negatives introduced in episode i often reduced at the next episode i+1. The model has learned in episode i and some of negatives are no longer hard nega-



(a) Epi-2. (b) Epi-3. (c) Epi-4. (d) Epi-5.

Figure 6: The t-SNE plots for negative swing. Query: *most popular breed of rabbit*. Its MRR@100 jumps up and down during iterative training: Epi-1 (0.14); Epi-2 (0.08 ↓); Epi-3 (0.13 ↑); Epi-4 (0.11 ↓); Epi-5 (0.13 ↑). The details of class A/B negatives and more swing cases are shown in Table 10 (appendix).

tives at episode i+1. However, in episode i+2, some already captured episode i negatives reappear as hard negatives, showing that the model forgets the information captured in episode i when learning in episode i+1 negatives. The model seems to swing back and forth between several learning mods and reflects the catastrophic forgetting behavior.

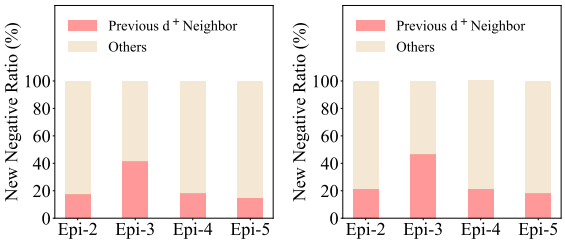
To further illustrate the negative swing, we visualize an example query and its negatives during ANCE learning via t-SNE (Van der Maaten and Hinton, 2008) in Figure 6. In this example there are two negative classes and the query is pushed between the two groups throughout the learning episodes. Capturing the information in one mod resulted in the forgetting of the other, as the model only sees one of them as hard negatives in each episode of ANCE training. Our further manual examination finds that for many queries with this negative swing behavior, the negative groups correspond to different common mistakes the retrieval system would make, for example, irrelevant documents that only cover part of the query. In Appendix F we show some example queries and negative groups of this behavior.

4 Teleportation Negative Sampling

In this section, we present ANCE-Tele, which introduces *teleportation negatives* to ANCE training. Motivated by our analysis, ANCE-Tele unions the training negatives along the time-axis of ANCE episodes, with the goal of smoothing training signal changes, improving negative group coverage, and, ultimately, reducing catastrophic forgetting.

The construction of *teleportation negatives* of ANCE-Tele is illustrated in Figure 1. Specifically, the training episode i is defined as:

$$\mathcal{L}_i^{\text{Tele}} = \sum_{q;d^+\in D^+} \sum_{d^-\sim D_i^{\text{Tele-neg}}} l(f(q,d^+), f(q,d^-)).$$



(a) ANCE (BERT) (b) ANCE (coCondenser).

Figure 7: Composition of new negatives in each episode of ANCE with CyclicLR. New negatives are those first introduced in each episode.

ANCE-Tele utilizes the same iterative training, but introduces two new components to standard ANCE negatives around q (Eqn. 4), momentum negatives (Eqn. 5) and lookahead negatives (Eqn. 6):

$$D_i^{\text{Tele-neg}} = \text{ANN}_{f(q,\circ;\theta_{i-1}^*)} \quad (4)$$
$$+\alpha D_{i-1}^{\text{Tele-neg}} \quad (5)$$
$$+\beta \text{ANN}_{f(d^+,\circ;\theta_{i-1}^*)} \quad (6)$$

Momentum negatives include training negatives from past episodes using a momentum queue (Eqn. 5). They smooth out the training signal changes between past and current episodes. A standard technique to reduce catastrophic forgetting (Kirkpatrick et al., 2017), including training signals from previous learning stages reminds the model to keep their learned knowledge.

Lookahead negatives predict the potential hard negatives in future training episodes using the neighbors of the positive document (Eqn. 6). Information from future is known to be beneficial in many other scenarios, for example, to probe the optimization space (Zhang et al., 2019) and to guide language generations (Lu et al., 2022). However, obtaining future information often requires extra costly operation steps, e.g., actually performing an extra ANCE episode to collect.

Methods	Parameters (DR/Teacher)	Training Negatives (In DR Training)	MS MARCO Dev MRR@10	MS MARCO Dev R@1K	Natural Question Test R@5	Natural Question Test R@20	Natural Question Test R@100	TriviaQA Test R@5	TriviaQA Test R@20	TriviaQA Test R@100
Sparse Retrieval										
BM25 (2017)	–	–	18.7	85.7	n.a	59.1	73.7	n.a	66.9	76.7
DeepCT (2020)	–	–	24.3	91.0	n.a	n.a	n.a	n.a	n.a	n.a
docT5query (2019)	–	–	27.7	94.7	n.a	n.a	n.a	n.a	n.a	n.a
GAR (2021)	–	–	n.a.	n.a.	60.9	74.4	85.3	73.1	80.4	85.7
Dense Retrieval										
DPR (2020)	110M	BM25	31.1	95.2	n.a.	78.4	85.4	n.a.	79.4	85.0
DrBoost (2022)	110M×5	Self (Boosted)	34.4	n.a.	n.a.	80.9	87.6	n.a	n.a	n.a
ANCE (2021)	110M	BM25+Self	33.0	95.9	n.a.	81.9	87.5	n.a.	80.3	85.3
SEED-Encoder (2021)	110M	BM25+Self	33.9	96.1	n.a.	83.1	88.7	n.a.	n.a.	n.a.
RocketQA (2021)	110M	Self+Filter	37.0	97.9	74.0	82.7	88.5	n.a	n.a	n.a
ME-BERT (2021)	110M	BM25+Rand	33.8	n.a.	n.a	n.a	n.a	n.a	n.a	n.a
GTR-base (2021)	110M	RocketQA	36.6	98.3	n.a	n.a	n.a	n.a	n.a	n.a
Condenser (2021)	110M	BM25+Self	36.6	97.4	n.a.	83.2	88.4	n.a.	81.9	86.2
coCondenser (2022)	110M	BM25+Self	38.2	**98.4**	75.8	84.3	89.0	76.8	83.2	**87.3**
coCondenser (Ours)	110M	BM25+Self	38.2	**98.4**	75.6	84.4	89.0	75.3	82.8	86.8
ANCE-Tele	110M	Self (Teleportation)	**39.1**$^\sharp$	**98.4**	**77.0**$^\sharp$	**84.9**	**89.7**$^\sharp$	**76.9**$^\sharp$	**83.4**$^\sharp$	**87.3**
For Reference: Bigger models and/or distillation from reranking teachers										
DPR-PAQ-large (2022)	355M	BM25+Self	34.0	n.a.	76.9	84.7	89.2	n.a	n.a	n.a
GTR-large (2021)	335M	RocketQA	37.9	99.1	n.a	n.a	n.a	n.a	n.a	n.a
GTR-XL (2021)	1.24B	RocketQA	38.5	98.9	n.a.	n.a	n.a	n.a	n.a	n.a
GTR-XXL (2021)	4.8B	RocketQA	38.8	99.0	n.a.	n.a	n.a	n.a	n.a	n.a
PAIR (2021a)	110M/330M	Pseudo Labels	37.9	98.2	74.9	83.5	89.1	n.a	n.a	n.a
RocketQA-v2 (2021b)	110M/110M	RocketQA+Filter	38.8	98.1	75.1	83.7	89.0	n.a	n.a	n.a
AR2 (2021)	110M/330M	BM25+Self	39.5	98.6	77.9	86.0	90.1	78.2	84.4	87.9

Table 1: First stage retrieval performances. The total number of parameters used in the DR model and its teacher, if applicable, are listed under *Parameters*. *Training Negatives* list the negative training examples sampled from BM25, randomly, using DR model itself, or inherited from previous methods. DR systems using significantly more parameters than $\text{BERT}_{\text{base}}$ (110M) during training are listed *For Reference* but not for fair comparisons. $\sharp$ indicates statistically significant improvements over coCondenser (Ours).

Instead, we propose to use the neighbors of d^+ to approximate the negatives that may appear in future episodes, i.e., "lookahead". Intuitively, as the dense retrieval training is to pull closer q and d^+ in the representation space, a side effect is that the neighbors around d^+ are also pulled to the query. Figure 7 shows that in ANCE training d^+ neighbors indeed contribute to a large fraction of new negatives introduced in future episodes, a handy feature for ANCE-Tele to efficiently incorporate future learning signals.

The sampling weights of momentum and lookahead negatives are controlled by hyperparameters (α, β), which we simply set as 0.5 without tuning.

Eliminating dependency on sparse retrieval. As shown in the last section, the sparse retrieval negatives are dropped after the warm up episode. ANCE-Tele thus directly starts from itself:

$$D_0^{\text{Tele-neg}} = \emptyset \tag{7}$$

$$\theta_0^* = \text{Pretrained Weights}. \tag{8}$$

This removes the dependency of dense retrieval training on sparse retrieval.

ANCE-Tele introduces little computation overhead. The momentum negatives can be cached on disk and merged when constructing the training data for new episodes. The lookahead negatives need one extra ANN retrieval operation per positive document, which is efficient (Johnson et al., 2021). None of them increases GPU computations, which are the bottleneck in training. Other than adding teleportation negatives, ANCE-Tele keeps other system components intact and can be plugged into with most dense retrieval systems.

5 Experimental Methodology

We describe our general experimental settings in this section and leave more details in Appendix.

Dataset. Following recent research, we conduct experiments on the first stage retrieval of three benchmarks: MS MARCO passage retrieval (Bajaj et al., 2016), Natural Questions (Kwiatkowski et al., 2019), and TriviaQA (Joshi et al., 2017). We use the exact same setting with DPR (Karpukhin et al., 2020), ANCE (Xiong et al., 2021), and coCondenser (Gao and Callan, 2022). More details of these datasets are in Appendix A.

Our Methods. ANCE-Tele fine-tunes on each benchmark starting from coCondenser (Gao and Callan, 2022), a recent retrieval-oriented language

α	β	N	MRR@10	R@1K
0.5	0.5	47	39.1	98.3
0.5	0.5	31	39.1	98.4
0.5	0.5	23	38.9	98.3
0.3	0.5	31	39.1	98.3
0.7	0.5	31	39.0	98.4
0.5	0.3	31	38.9	98.4
0.5	0.7	31	38.9	98.3

Table 2: ANCE-Tele on MARCO regarding to different hyperparameter values: the weights of momentum negatives α, lookahead negatives β and the total number of negative samples per (q, d^+) pair N.

model continuously pretrained from $\text{BERT}_{\text{base}}$. The loss function is cross-entropy. We use 31 training negatives per query for MARCO, and 11 for NQ and TriviaQA, sampled from the union of Top 200 KNN results from query (ANCE), momentum, and lookahead negatives. Momentum and lookahead sample weights (α, β) are 0.5. All experiments are conducted at $\text{BERT}_{\text{base}}$ scale (110M parameters), the most popular setting currently.

Baselines. All previous first stage retrieval methods on the benchmarks are directly comparable with our performance empirically, as long as they follow the standard setting. The fair baselines are those DR methods that are at the same $\text{BERT}_{\text{base}}$ scale, which we compare with their reported numbers. We include our run of coCondenser for direct comparison, especially on TriviaQA where full implementation details were not publicly available. Descriptions of the baselines are in Appendix B.

We also list the results from larger pretraining models and/or distillation from stronger cross-encoder reranking teachers, but only for reference purposes. How to more efficiently leverage the power of large scale pretraining models and how to scale techniques to billions of parameters are important future research directions.

Implementation Details. We implement ANCE-Tele using PyTorch (Paszke et al., 2019) and HuggingFace (Wolf et al., 2020), and run all MARCO experiments on a single A100 GPU (40G) and all NQ and TriviaQA experiments on 4 V100 GPUs (32G). In each episode, the number of training epochs and query batch size is the same as in the previous work (Gao and Callan, 2022; Karpukhin et al., 2020). We directly use the last checkpoint in our run instead of selecting checkpoints based on empirical performances. The exact configurations of our experiments are listed in Appendix C and our open-source repository.

Pretrain Models	Methods	MRR@10	R@1K
BERT	Zero-Shot	0.09	0.21
	DPR	32.4	94.9
	ANCE	33.5	95.8
	Q-Neg	28.7	88.3
	Q-Neg w/ Mom-Neg	35.6	96.0
	Q-Neg w/ LA-Neg	33.7	94.8
	ANCE-Tele	**36.0**$^{\dagger\ddagger\flat\natural\S}$	**96.2**$^{\dagger\ddagger\flat\natural\S}$
Condenser	Zero-Shot	0.61	11.4
	DPR	33.8	96.1
	ANCE	35.0	96.7
	Q-Neg	30.4	87.8
	Q-Neg w/ Mom-Neg	37.0	97.0
	Q-Neg w/ LA-Neg	36.7	96.9
	ANCE-Tele	**37.2**$^{\dagger\ddagger\flat\natural\S}$	**97.1**$^{\dagger\ddagger\flat\natural}$
coCondenser	Zero-Shot	11.4	77.1
	DPR	36.2	97.7
	ANCE	36.8	98.1
	Q-Neg	36.6	95.3
	Q-Neg w/ Mom-Neg	38.6	**98.4**
	Q-Neg w/ LA-Neg	38.8	98.3
	ANCE-Tele	**39.1**$^{\dagger\ddagger\flat\natural}$	**98.4**$^{\dagger\ddagger\flat\natural}$

Table 3: MARCO performances with different pretrained models and training negative samples. All methods are trained for three episodes, except for Zero-Shot and DPR. Superscripts indicate statistically significant improvements over Zero-Shot†, DPR‡, ANCE$^{\flat}$, Query-Neg$^{\natural}$, Query-Neg w/ Momentum-Neg$^{\sharp}$, Query-Neg w/ Lookahead-Neg§.

6 Evaluation Results

This section first evaluates ANCE-Tele and its ablations. Then we analyze the influences and characteristics of teleportation negatives.

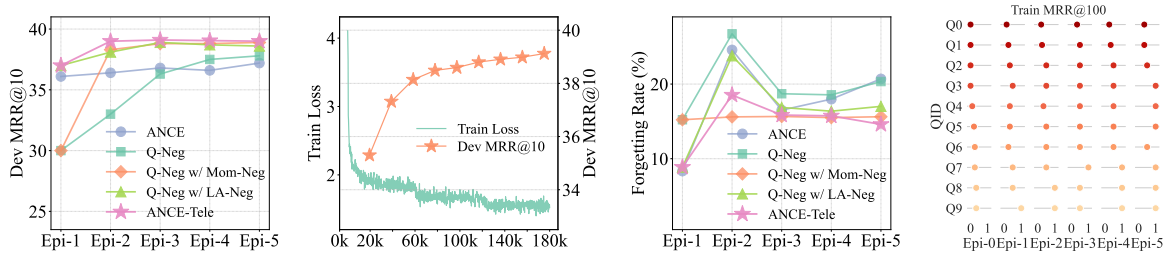
6.1 Overall Results

The overall performances are listed in Table 1. At $\text{BERT}_{\text{base}}$ size, ANCE-Tele outperforms previous state-of-the-arts on nearly all metrics. As shown in *Training Negatives*, previous methods use various ways to construct training signals: sparse retrieval negatives, self negatives, and/or additional filters. ANCE-Tele only uses teleportation negatives from itself and does not depend on sparse retrieval.

ANCE-Tele shows competitive performances with systems with more parameters and/or distillations from reranking teachers. It outperforms GTR-XXL on MRR@10 at MARCO, albeit the latter uses T5-XXL (Ni et al., 2021) with about 50x more parameters. ANCE-Tele also achieves similar performance with AR2, which jointly trains retriever with knowledge distilled from an $\text{ERNIE}_{\text{Large}}$ reranking model (Zhang et al., 2021).

6.2 Ablations

We perform two ablations to study the hyperparameters and design choices of ANCE-Tele.



(a) Iteration performance. (b) ANCE-Tele (Epi-3). (c) Forgetting Rate. (d) Per Query Accuracy.

Figure 8: Training behavior with ANCE-Tele variants. (a) shows the iteration performance. (b) presents the training loss and dev curve of ANCE-Tele at Epi-3. (c) exhibits the forgetting rate. (b) shows the accuracy of ten training queries during ANCE-Tele training, which are the same queries sampled in the previous ANCE analysis.

Hyperparameters. We keep the design choices of ANCE-Tele as simple as possible. The only hyperparameters introduced are the α and β to balance momentum and lookahead negatives. Another notable hyperparameter is the number of negatives sampled per (q, d^+) pair, inherited from ANCE. In Table 2 we show that ANCE-Tele is robust to these hyperparameter variations.

Negative Sampling. Table 3 lists the performance of ANCE-Tele when starting from different pretrained models with different negatives. We use Condenser as representatives of pretraining models using information bottleneck for better sequence embedding (Gao and Callan, 2021; Lu et al., 2021; Wang et al., 2021), and coCondenser to represent pretraining models with sequence contrastive learning (Meng et al., 2021; Ni et al., 2021; Gao and Callan, 2022). The baseline performances of zero-shot, DPR (BM25 negatives), and ANCE with Condenser and coCondenser confirm the benefits of these two pretraining techniques.

ANCE-Tele provides robust improvements despite different pretrained starting points. Among different negative selection approaches, Query negatives alone lag behind ANCE after three training episodes, confirming the important role of BM25 negatives in ANCE. Adding either momentum or lookahead negatives eliminates the dependency on sparse retrieval, even when starting from BERT whose zero-shot performance is barely nonrandom. The two provide a significant boost when added individually or combined. The benefits are further analyzed in next experiments.

6.3 Influence of Teleportation Negatives

In this experiment we evaluate the influence of teleportation negatives in the training process.

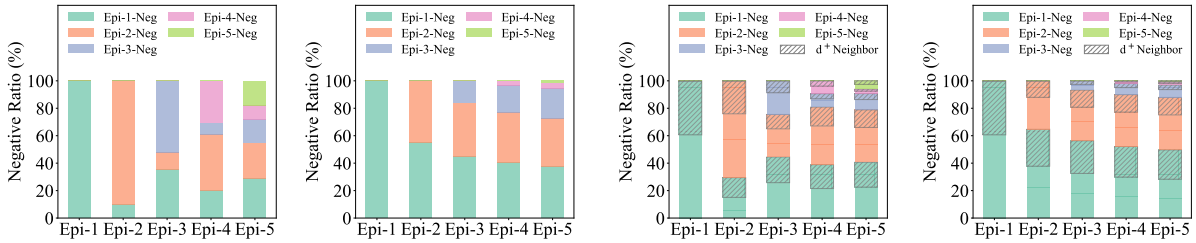
Training Stability. In Figure 8a we plot the performance of ANCE-Tele after different training episodes. In comparison to ANCE in Figure 2, ANCE-Tele variants with momentum negatives are more stable. Lookahead negatives improve converging speed. Adding them improves accuracy at the first episode by more than 20% relatively, a key contributor to the elimination of sparse retrieval negatives. We also show experimentally that adding BM25 negatives brings no additional benefits to ANCE-Tele, as presented in Table 8 (Appendix E). In addition, Figure 8b zooms into the third episode of ANCE-Tele and further confirms its stability and converging speed.

Forgetting Rate. Figure 8c plot the forgetting ratio of ANCE-Tele variations. Q-Neg alone yields large forgetting rate, especially at the second episode (Figure 8a). The momentum-negatives significantly reduce the forgetting rate throughout training. Including past negatives does remind the model to maintain learned knowledge. Furthermore, we revisit the ten training queries analyzed in Figure 3 and track the performance of ANCE-Tele on them. As shown in Figure 8d, the per query MRR oscillation reduces significantly in ANCE-Tele. Most of them maintain high accuracy throughout training episodes.

6.4 Composition of Teleportation Negatives

This experiment analyzes the composition of teleportation negatives. The sources of training negatives (by first encounter) are plotted in Figure 9.

Using Q-Neg alone suffers from negative swing (Figure 9a). Many negatives selected in Epi-1 got pushed away in Epi-2 (*ratio* $\downarrow$) and then reemerge in Epi-3 (*ratio* $\uparrow$). It corresponds to the high forgetting rate at the end of Epi-2 (Figure 8c). Momentum negatives nearly eliminate this issue via a smooth combination of new and inherited negatives. In Figure 9b, hard negatives are introduced gradually and remain in effect for multiple episodes.



(a) Q-Neg. (b) Q-Neg w/ Momentum. (c) Q-Neg w/ Lookahead. (d) ANCE-Tele.

Figure 9: Composition of training negatives in ANCE-Tele variations. Only lookahead includes d^+ neighbors.

The neighbors of positive document (lookahead negatives) is a decent approximation of oracle future negatives in later episodes. In Figure 9c, significantly lower fractions of new negatives are introduced in later episodes, compared to Q-Neg. The d^+ neighbors from the last episode cover 20%-30% of would-be new negatives of the current episode, pretty efficient given its negligible computing cost.

As shown in Figure 9d, the teleportation negatives are composed by a diverse collection of negative sources and evolve more smoothly through episodes. This helps ANCE-Tele improve the optimization stability, efficiency, and reduces catastrophic forgetting in dense retrieval training. Appendix E also studies the influence of removing the overlapping part from multiple negative sources.

Appendix F exhibits some example negatives which ANCE learned, forgot, and then relearned. The swing between different distractor groups aligns well with our intuitions in information retrieval. Those irrelevant documents cover different partial meanings of the query and require different ways to detect. ANCE-Tele eliminates this behavior by combining them via teleportation negatives.

7 Conclusions

We present ANCE-Tele, a simple approach that effectively improves the stability of dense retrieval training. Our investigation reveals the issues underlying the training instability: the catastrophic forgetting and negative swing behaviors. ANCE-Tele resolves these issues by introducing teleportation negatives, which smooth out the learning process with momentum and lookahead negatives.

ANCE-Tele leads to strong empirical performance on web search and OpenQA with improved training stability, convergence speed, and reduced catastrophic forgetting. Our analysis demonstrates the benefits of teleportation negatives and their behavior during training iterations. Overall, ANCE-Tele addresses an intrinsic challenge in dense retrieval training with reduced engineering effort and minimum computation overhead. It can be used as a plug-in upgrade for the first stage retrieval of many language systems.

Limitations

One limitation of ANCE-Tele is the lack of more detailed characterization of the representation space, in dense retrieval and other embedding-based matching tasks. Better understanding of the representation space will enable the development of more automatic ways to navigate the training signal space. The training signal selection is getting more and more important with deep neural systems. We need more tools to capture, analyze, and improve this new aspect of data-driven AI.

Though ANCE-Tele is robust to different pretraining models, whether tailored to retrieval or not, there is still limited understanding on the relationship between the pretraining task and the pretrained model's generalization ability in dense retrieval. Recent research observed several mismatches between the two, but we still do not quite fully understand their interactions.

Acknowledgments

This work is partly supported by Institute Guo Qiang at Tsinghua University and NExT++ project from the National Research Foundation, Prime Minister's Office, Singapore under its IRC@Singapore Funding Initiative. We thank all anonymous reviewers for their suggestions.

References

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. In *Proceedings of CoCo@ NIPS 2016*.

Qi Chen, Haidong Wang, Mingqin Li, Gang Ren, Scarlett Li, Jeffery Zhu, Jason Li, Chuanjie Liu, Lintao Zhang, and Jingdong Wang. 2018. *SPTAG: A library for fast approximate nearest neighbor search*.

Zhuyun Dai and Jamie Callan. 2020. Context-aware term weighting for first stage passage retrieval. In *Proceedings of SIGIR 2020*.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*.

Luyu Gao and Jamie Callan. 2021. Condenser: a pre-training architecture for dense retrieval. In *Proceedings of EMNLP 2021*.

Luyu Gao and Jamie Callan. 2022. Unsupervised corpus aware language model pre-training for dense passage retrieval. In *Proceedings of ACL 2022*.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Tevatron: An efficient and flexible toolkit for dense retrieval. *arXiv preprint arXiv:2203.05765*.

Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. 2020. Accelerating large-scale inference with anisotropic vector quantization. In *Proceedings of ICML 2020*.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *Proceedings of SIGIR 2021*.

Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently teaching an effective dense retriever with balanced topic aware sampling. In *Proceedings of SIGIR 2021*.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of CIKM 2013*.

Gautier Izacard and Edouard Grave. 2020. Distilling knowledge from reader to retriever for question answering. *arXiv preprint arXiv:2012.04584*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of ACL 2017*.

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 EMNLP 2020*.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming catastrophic forgetting in neural networks. *PNAS*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *TACL*.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of ACL 2019*.

Patrick Lewis, Barlas Oguz, Wenhan Xiong, Fabio Petroni, Scott Yih, and Sebastian Riedel. 2022. Boosted dense retriever. In *Proceedings of NAACL 2022*.

Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. In-batch negatives for knowledge distillation with tightly-coupled teachers for dense retrieval. In *Proceedings of RepL4NLP-2021*.

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 2021*.

Shuqi Lu, Di He, Chenyan Xiong, Guolin Ke, Waleed Malik, Zhicheng Dou, Paul Bennett, Tie-Yan Liu, and Arnold Overwijk. 2021. Less is more: Pretrain a strong siamese encoder for dense text retrieval using a weak decoder. In *Proceedings of EMNLP 2021*.

Ximing Lu, Sean Welleck, Peter West, Liwei Jiang, Jungo Kasai, Daniel Khashabi, Ronan Le Bras, Lianhui Qin, Youngjae Yu, Rowan Zellers, Noah A. Smith, and Yejin Choi. 2022. NeuroLogic a*esque decoding: Constrained text generation with lookahead heuristics. In *Proceedings of NAACL 2022*.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *TACL*.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2021. Generation-augmented retrieval for open-domain question answering. In *Proceedings of ACL 2021*.

Yu Meng, Chenyan Xiong, Payal Bajaj, Saurabh Tiwary, Paul Bennett, Jiawei Han, and Xia Song. 2021. COCO-LM: Correcting and contrasting text sequences for language model pretraining. In *Proceedings of NeurIPS 2021*.

Tom Mitchell, William Cohen, Estevam Hruschka, Partha Talukdar, Bishan Yang, Justin Betteridge, Andrew Carlson, Bhavana Dalvi, Matt Gardner, Bryan Kisiel, et al. 2018. Never-ending learning. *Communications of the ACM*.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas Tezak, Jong Wook Kim, Chris Hallacy, et al. 2022. Text and code embeddings by contrastive pretraining. *arXiv preprint arXiv:2201.10005*.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Ábrego, Ji Ma, Vincent Y Zhao, Yi Luan, Keith B Hall, Ming-Wei Chang, et al. 2021. Large dual encoders are generalizable retrievers. *arXiv preprint arXiv:2112.07899*.

Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to doctttttquery.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. UniK-QA: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of NAACL 2022*.

Barlas Oğuz, Kushal Lakhotia, Anchit Gupta, Patrick Lewis, Vladimir Karpukhin, Aleksandra Piktus, Xilun Chen, Sebastian Riedel, Wen-tau Yih, Sonal Gupta, et al. 2022. Domain-matched pre-training tasks for dense retrieval. In *Findings of NAACL 2022*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Proceedings of NeurIPS 2019*.

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 NAACL-HLT 2021*.

Ruiyang Ren, Shangwen Lv, Yingqi Qu, Jing Liu, Wayne Xin Zhao, Qiaoqiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021a. Pair: Leveraging passage-centric similarity relation for improving dense passage retrieval. In *Findings of ACL-IJCNLP 2021*.

Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021b. RocketQAv2: A joint training method for dense passage retrieval and passage re-ranking. In *Proceedings of EMNLP 2021*.

Leslie N Smith. 2017. Cyclical learning rates for training neural networks. In *Proceedings of WACV 2017*.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*.

Kexin Wang, Nils Reimers, and Iryna Gurevych. 2021. Tsdae: Using transformer-based sequential denoising auto-encoderfor unsupervised sentence embedding learning. In *Findings of EMNLP 2021*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of EMNLP 2020*.

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 *Proceedings of ICLR 2021*.

Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the use of lucene for information retrieval research. In *Proceedings of SIGIR 2017*.

Shi Yu, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2021. Few-shot conversational dense retrieval. In *SIGIR 2021*.

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 SIGIR 2021*.

Hang Zhang, Yeyun Gong, Yelong Shen, Jiancheng Lv, Nan Duan, and Weizhu Chen. 2021. Adversarial retriever-ranker for dense text retrieval. In *Proceedings of ICLR 2021*.

Michael Zhang, James Lucas, Jimmy Ba, and Geoffrey E Hinton. 2019. Lookahead optimizer: k steps forward, 1 step back. *Proceedings of NeurIPS 2019*.

Chen Zhao, Chenyan Xiong, Jordan Boyd-Graber, and Hal Daumé III. 2021. Distantly-supervised dense retrieval enables open-domain question answering without evidence annotation. In *Proceedings of EMNLP 2021*.

A Dataset Details

In our experiments, we utilize three evaluation datasets, i.e., MS MARCO, NQ, and TriviaQA, and their statistical details are shown in Table 4.

- **MS MARCO**: The MS MARCO Passage Ranking (Bajaj et al., 2016) is a large scale of web search dataset. The queries are constructed from Bing's query logs, and each query comes with at least one related passage.
- **NQ**: Natural Questions (Kwiatkowski et al., 2019) is a widely used question-answering dataset constructed on Wikipedia. The questions come from the Google search engine, and the answers are identified as text spans in the Wikipedia article.
- **TriviaQA**: The TriviaQA (Joshi et al., 2017) is a reading comprehension dataset collected from Wikipedia and the web. Trivia enthusiasts author the question-answer pairs.

Following the prior work (Xiong et al., 2021), we report MRR@10 and Recall@1K on the Dev set of MARCO. The Mean Reciprocal Rank (MRR) is the average of the inverse ranking positions of the first retrieved relevant passages for all queries. Recall@1K represents the proportion of queries containing relevant passages in the top 1K retrieved passages. We report Recall@{5,10,100} on the test set of NQ and TriviaQA, and use passage titles on these three datasets, which is consistent with the previous research (Karpukhin et al., 2020; Gao and Callan, 2022). Statistical significance is examined by permutation test with $p < 0.05$.

Datasets	Train	Dev	Test	Corpus Size
MS MARCO	502,939	6,980	6,837	8,841,823
NQ	79,168	8,757	3,610	21,015,324
TriviaQA	78,785	8,837	11,313	21,015,324

Table 4: The statistics of evaluation dataset. Train denotes the original training examples without filtering.

B Main Baselines

This section briefly introduces the DR methods in Table 1, the main baselines of ANCE-Tele.

DPR (Karpukhin et al., 2020) is a classical dual-encoder DR method, which maps the query and passage to dense vectors separately, using the `[CLS]` token from the pre-trained model. Based on the dense vectors, the dot product is used to compute their similarity. DPR is trained with in-batch and BM25 negatives, using $\text{BERT}_{\text{base}}$ as initial models.

DrBoost (Lewis et al., 2022) is an ensemble-like DR method that utilizes negatives sampled from itself for iterative training. The training starts with $\text{BERT}_{\text{base}}$ and random negatives. Its final representation of the query and passage is the concatenation of the output dense vectors of all component models during training.

ANCE (Xiong et al., 2021) is a popular DR training strategy. It starts warm-up training based on BM25 negatives and continues iterative training using the hard negatives retrieved from the latest checkpoint. $\text{RoBERTa}_{\text{base}}$ is the initial model.

SEED-Encoder (Lu et al., 2021) is an IR-oriented pre-trained model with an auto-encoder architecture. Pretraining configures the encoder with a relatively weak decoder to push the encoder to obtain more robust text representations. Fine-tuning only leverages the encoder, whose parameter quantity is equivalent to $\text{BERT}_{\text{base}}$ and the fine-tuning process is the same as ANCE.

RocketQA (Qu et al., 2021) is a dual-encoder DR initialized with the pre-trained model ERNIE 2.0 (Base). The training uses cross-batch negatives and a re-ranker to filter the false negatives. Besides, it is augmented with additional training data.

ME-BERT (Luan et al., 2021) represents each passage with multiple vectors and down-projects the vectors to the final representation. The training leverages random, BM25, and in-batch negatives, and utilizes $\text{BERT}_{\text{base}}$ for initialization.

GTR-base (Ni et al., 2021) is a dual-encoder DR method, initialized using the encoder part of the pre-trained model T5_{base}, trained with hard negatives released by RocketQA (Qu et al., 2021).

Condenser (Gao and Callan, 2021) is also an IR-oriented pre-trained model based on $\text{BERT}_{\text{base}}$. During pretraining, it enhances the representation ability of `[CLS]` token by changing the connections between different layers of Transformer blocks. Fine-tuning uses BM25 and self-mining negatives.

coCondenser (Gao and Callan, 2022) adds a contrastive pre-training task to Condenser. The task randomly extracts spans from the specific corpus, regarding spans from the same/different passages as positive/negative examples, and learns to discriminate them. The model scale and fine-tuning process are the same as that of Condenser.

Hyperparameters	MS MARCO	NQ	TriviaQA
Max query length	32	32	32
Max passage length	128	156	156
Negative mining depth	200	200	200
Batch size (query size per batch)	8	128	128
Positive number per query	1	1	1
Negative number per query	31	11	11
Initial model	`co-condenser-marco`	`co-condenser-wiki`	`co-condenser-wiki`
Learning rate	5e-6	5e-6	5e-6
Optimizer	AdamW	AdamW	AdamW
Scheduler	Linear	Linear	Linear
Warmup ratio	0.1	0.1	0.1
Training epoch	3	40	40
Momentum negative weight α	0.5	0.5	0.5
Lookahead negative weight β	0.5	0.5	0.5
Epi-1 new negative mining source	`co-condenser-marco`	`co-condenser-wiki`	`co-condenser-wiki`
Epi-2 new negative mining source	Epi-1 (20k step)	Epi-1 (2k step)	Epi-1 (2k step)
Epi-3 new negative mining source	Epi-2 (20k step)	Epi-2 (2k step)	Epi-2 (2k step)

Table 5: Hyperparameters of ANCE-Tele training. ANCE-Tele uses coCondenser as the initial pre-trained model. We use the `co-condenser-marco` version on MARCO, which is continuously pre-trained on the MARCO corpus. On NQ and TriviaQA, we use the `co-condenser-wiki` version, which continues pre-training on the Wikipedia corpus. We directly utilize the open-source model files of `co-condenser-marco` and `co-condenser-wiki` from the huggingface.co community.

Stage	MS MARCO	NQ	TriviaQA
Epi-1	2.5h	1.2h	1.2h
Epi-2	2.5h	1.2h	1.2h
Epi-3	23.5h	10.8h	10.8h
Index refresh	1.2h	2.7h	2.7h
Refresh number	3	3	3
Overall	32.1h	21.3h	21.3h

Table 6: Training time for ANCE-Tele with three training episodes. Training on MARCO uses a single A100 GPU (40G), and training on NQ and TriviaQA utilizes 4 V100 GPUs (32G).

Hyperparameters	Standard ANCE
Negative refresh step	10k
Negative mining depth	200
Batch size (query size per batch)	8
Positive number per query	1
Negative number per query	7
Learnig rate	1e-6
Optimizer	AdamW
Scheduler	Linear
Warmup step	5k
Cyclical warmup	w/o

Table 7: Hyperparameters of training standard ANCE in the analysis of Section 3.2.

C Implementation Details

This section exhibits the detailed hyperparameters of ANCE-Tele and analyzes its training efficiency.

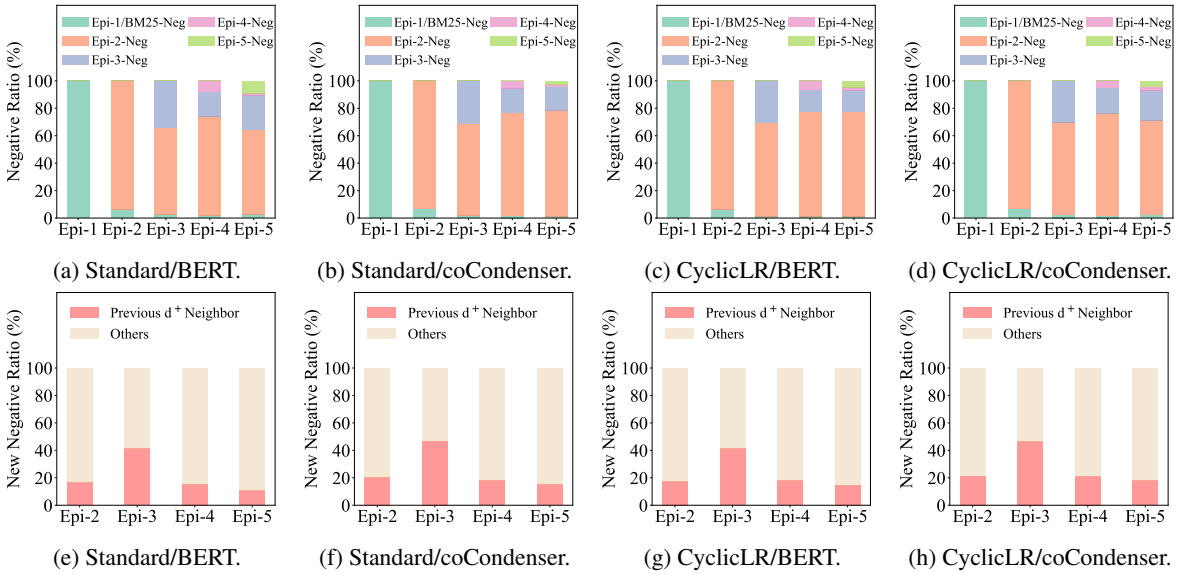
C.1 Hyperparameters

Table 5 lists the detailed hyperparameters used by ANCE-Tele on the three evaluation datasets. The entire training process of ANCE-Tele consists of three episodes, labeled Epi-1, Epi-2, and Epi-3. Each episode trains the model from scratch based on the same initial model, using the same hyperparameters but different training negatives. Take Epi-3 as an example: On MARCO, each training batch contains eight queries, each query is equipped with one positive and 31 negatives, and the number of training epochs is three. On NQ and TriviaQA, the query batch size is 128, each query is equipped with one positive and eleven negatives, and the total training epoch is 40. Next, we introduce how to obtain the training negatives for each episode.

C.2 Negative Mining

The training negatives of ANCE-Tele come from newly-mined negatives and momentum negatives, where the newly-mined negatives include standard ANCE negatives and lookahead negatives. We set the ratio of newly-mined negatives to momentum negatives to 1:1 (α=0.5). Specifically, Epi-1 uses the initial pre-trained model to mine new negatives, acting as the first training episode, so there is no negative momentum. To gather ANCE negatives and lookahead negatives for Epi-1, we first build two indexes for the query and positive passage separately, and then sample negatives from the top retrieved passages, where we keep the proportion of the two class negatives at 1:1 (β=0.5).

During iterative training, we employ an earlier



(a) Standard/BERT. (b) Standard/coCondenser. (c) CyclicLR/BERT. (d) CyclicLR/coCondenser.

(e) Standard/BERT. (f) Standard/coCondenser. (g) CyclicLR/BERT. (h) CyclicLR/coCondenser.

Figure 10: Composition of training negatives for four ANCE variants. Figures (a) and (b) show the components of all training negatives for standard ANCE (BERT/coCondenser), and Figures (c) and (d) show the components of all training negatives for ANCE (BERT/coCondenser) trained with CyclicLR. We exhibit the new negative constituent of standard ANCE (BERT/coCondenser) in Figures (e) and (f), and show the composition of new negatives for ANCE (BERT/coCondenser) trained with CyclicLR in Figures (g) and (h).

Methods	MRR@10	R@1k
ANCE-Tele	39.1	98.4
ANCE-Tele (continue training)	39.2	98.3
ANCE-Tele (w/ BM25 negatives)	39.0	98.4
ANCE-Tele (w/o overlap negatives)	38.2	98.4

Table 8: The results of ANCE-Tele variants on MARCO.

Episode	Methods	Forget Rate (%)
Epi-1	ANCE-Tele	8.9
	ANCE-Tele (continue training)	8.9
Epi-2	ANCE-Tele	18.5
	ANCE-Tele (continue training)	13.5
Epi-3	ANCE-Tele	15.9
	ANCE-Tele (continue training)	13.2

Table 9: Forgetting rate of ANCE-Tele in different training modes.

negative-refreshing step because we find that refreshing negatives at an early stage achieves comparable performance and saves a lot of training time. For example, Epi-2 mines new negatives based on the early training checkpoints of Epi-1, where we set the negative-refreshing step as one-tenth of the total training steps without much tuning, i.e., 20k for MARCO and 2k for NQ and TrviaQA. In addition, Epi-2's momentum negatives are derived from the training negatives of the last episode (Epi-1).

C.3 Training Efficiency

Table 6 shows the time cost of training ANCE-Tele in three episodes. Our implementation is initially based on Tevatron (Gao et al., 2022) with modifications and has been integrated into OpenMatch (Liu et al., 2021).

D Supplement to ANCE Results

In this section, we first introduce the analysis setting of ACNE used in Section 3 and then investigate the negative composition for more ANCE variants.

ANCE Steup. The hyperparameters of ANCE remain the same as in the previous research (Xiong et al., 2021). Table 7 shows the configurations. The warm-up method during iterative training is the difference between standard ANCE and ANCE with CyclicLR. Standard ANCE uses a single warm-up learning rate, while ANCE with CyclicLR uses a recurrent warm-up learning rate and achieves higher stability performance but slower convergence.

ANCE Variants. Figure 10 presents the training negative composition of four ANCE variants, i.e., standard ANCE (BERT/coCondenser), ANCE trained with CyclicLR (BERT/coCondenser). Figure 10a to Figure 10d show the component of the total training negatives, and Figure 10e to Figure 10h show the composition of their new negatives.

Notably, BM25 negatives are quickly discarded for all variants after the first training episode, and

Queries	Class A Negatives	Class B Negatives
(a) most popular breed of rabbit	The Golden Retriever is one of the **most popular breeds** in the United States. Learn more about this loveable dog with Golden Retriever facts & pictures on petMD.	**Rabbit** habitats include meadows, woods, forests, grasslands, deserts and wetlands. **Rabbits** live in groups, and the best known species, the European **rabbit**, lives in underground burrows, or **rabbit** holes.
(b) what is the main difference between a **lightning strike** and **static electricity**	**Lightning** is a bright flash of electricity produced by a thunderstorm. All thunderstorms produce **lightning** and are very dangerous. If you hear the sound of thunder, then you are in danger from **lightning**.	**Static Electricity**: Introducing Atoms This lesson lays the groundwork for further study of **static** and current **electricity** by focusing on the idea of positive and negative charges at the atomic level.
(c) what **essential oil** to **stop the itching of bug bites**	**Essential Oils** to Treat Poison Ivy Rashes Having strong antiseptic properties, cypress **essential oil** is also among the best **essential oils** to treat poison ivy rashes.	Home Remedy to **Stop Itching From Bug Bites** Home remedies treating **itching from bug bites** vary in effectiveness, depending upon the type of **bug bite** and the severity of the **itching**.
(d) **how long** did the **han dynasty last**	**How long** did the Roman Empire **last**? From the time of Augustus to the time of Constantine Dragases, it was 1484 years.	**Han Dynasty** Achievements It was during the **Han** period that contact with the West through the Silk Road was first established.
(e) where are the **pectoral muscles located**	Biceps femoris Biceps femoris. The biceps femoris is a double-headed **muscle located** on the back of thigh. It consists of two parts: the long head, attached to the ischium (the lower and back part of the hip bone), and the short head, which is attached to the femur bone.	**Pectoralis** Major Muscle Function. The **pectoralis** major muscle is the most important muscle for the adduction and anteversion of the shoulder joint which is why it is also known as the breaststroke muscle.
(f) how many **calories and carbs** in **cantaloupe**	Thomas' everything bagel - 280 **calories**, 3g of fat, and 53g of **carbs** per bagel. Visit our site for complete nutrition facts information for this item and 100,000+ additional foods.	10 Amazing Nutritional Benefits of **Cantaloupe** 10 Amazing Health & Nutritional Benefits of **Cantaloupe**, Nutrition Facts of **Cantaloupe**. **Cantaloupe** is a delicious fruit with a unique flavor.
(g) who **invented** the **shopping cart**	The zoetrope was **invented** in 1834 by William Horner who called it a daedalum or daedatelum. However it is believed that Horner may have based his **invention** on that of a basic zoetrope created by a Chinese inventor.	Your **shopping cart** is empty! Founded in 1921 as Verlag Chemie, we can look back over 90 years of publishing in the fields of chemistry, material science, physics and life sciences as well as business and trade.
(h) when was **houston** art **institute** founded	Cato **Institute** The Cato **Institute's** articles of incorporation were filed in December 1974, with the name Charles Koch Foundation, listing the original directors as Charles Koch, George Pearson, and Roger MacBride.	**Houston** City Council The **Houston** City Council is a city council for the city of **Houston** in the U.S. state of Texas. Currently, there are sixteen members, 11 elected from council districts and five at-large.

Table 10: Cases of ANCE's negative swing behavior in the iterative training process. We sample 8 MARCO training queries whose negatives show two classes of features during training: **Class A negative** represents the negative that comes in Epi-2 but goes out in Epi-3 and again in Epi-4. Conversely, **Class B negative** comes in Epi-3 but goes out in Epi-4 and again in Epi-5. It is worth noting that these two types of negatives cover different aspects of the query.

the negative swing phenomenon is most evident in standard ANCE (BERT), as shown in Figure 10a. Moreover, we observe the phenomenon on all four variants, i.e., considerable new negatives of the current episode reside close to d^+ of the last episode.

E Supplement to ANCE-Tele Results

Additionally, we provide the results of other ANCE-Tele variants using different training modes and negatives. Experiments are performed on MARCO, and the pre-trained model uses coCondenser, and the maximum training episode is set to three. The results are presented in Table 8.

ANCE-Tele (continue training). ANCE adopts the *continue training* mode — the current episode takes the output model of the last episode as the initial model. By contrast, ANCE-Tele employs the *training from scratch* mode, whereby each training episode initializes with the initial pre-train model coCondenser. For comparison, we also implement ANCE-Tele using the *continue training* mode. As shown in Table 8, the change in training mode makes little difference in performance, and even a slight MRR@10 improvement has been observed in the *continue training* variant. Furthermore, we compare the forgetting rate between the two training modes and exhibit the results in Table 9. The results indicates that the *continue training* mode further reduces the forgetting rate of ANCE-Tele.

ANCE-Tele (w/ BM25 negatives). We also test

the effect of adding BM25 negatives to ANCE-Tele negatives. As expected, the addition of BM25 negatives brings no additional benefits, which shows little complementarity between ANCE-Tele negatives and BM25 negatives.

ANCE-Tele (w/o overlap negatives). ANCE-Tele negatives from different mining sources (momentum/lookahead) share some overlaps. We thus test the effect of performing deduplication based on ANCE-Tele negatives. As shown in Table 8, it is expected that negative deduplication significantly impacts ANCE-Tele. There are overlapping negatives because different mining sources refer to certain negatives multiple times, which can be considered significant negatives. Intuitively, keeping the overlapping part in the negative pool is equivalent to weighting them during training, which benefits the final performance.

F Case Studies

Table 10 lists ten training cases on MARCO to visually show the negative swing behavior of ANCE.

We observe two typical oscillating negative classes near the training query during iterative training, i.e., **Class A** and **Class B**. Class A negatives first appear in Epi-2 (Top200 KNN), are discarded at Epi-3, and reappear in Epi-4 training negatives. On the contrary, Class B negatives first appear in Epi-3, are discarded at Epi-4 and reappear at Epi-5. The swing behavior of these two types of negative groups is precisely the opposite. More Interestingly, we observe that the two classes of hard negatives cover different aspects of the query.

For the query "most popular breed of rabbit" in case (a), the class A negative captures the semantics of "most popular breed" but the breed is "dog" rather than "rabbit". Instead, the class B negative alone captures "rabbit". In case (b), the query asks the difference between lightning and static electricity; its class A negative alone captures "lightning", while its class B negative captures only "static electricity". In case (c), the query asks "what essential oils can stop itching from bug bites", the class A negative passage mentions "essential oils", but the passage is about treating poison ivy rashes. In contrast, the class B negative mentions the "stop itching bug bites" part but the key query term "essential oils" does not appear. Likewise, case (d) asks "how long the Han dynasty lasted". Its class A negative captures the duration of the dynasty but focuses on the Roman Empire; class B negative captures the "Han dynasty", but it describes the achievements of the Han dynasty rather than its duration. This phenomenon also exists in the remaining examples.

To sum up, these examples reveal that the model swings between the distinct classes of negatives during iterative training. In contrast, ANCE-Tele carries teleportation negatives to mitigate negative swings for better stability and convergence speed.

G Contributions

Si Sun and Chenyan Xiong designed the methods and the experiments. Si Sun conducted the experiments. Chenyan Xiong and Si Sun wrote the paper. Yue Yu and Arnold Overwijk engaged in the discussion, revision, and response to reviewers. Zhiyuan Liu gave suggestions and feedback about the project and provided the experimental hardware. Jie Bao proofread the paper.