Improving Open-Domain Answer Sentence Selection by Distributed Clients with Privacy Preservation

Anonymous ACL submission

Abstract

Open-domain answer sentence selection (OD-AS2), as a practical branch of open-domain question answering (OD-QA), aims to respond to a query by a potential answer sentence from a large-scale collection. A dense retrieval model plays a significant role across different solution paradigms, while its success depends heavily on sufficient labeled positive QA pairs and diverse hard negative sampling in contrastive learning. However, it is hard to satisfy such dependencies in a privacy-preserving distributed scenario, where in each client, less in-domain pairs and a relatively small collection cannot support effective dense retriever training. To alleviate this, we propose a brand-new learning framework for **P**rivacy-preserving **D**istributed O**D-AS2**, dubbed PDD-AS2. Built upon federated learning, it consists of a client-customized query encoding for better personalization and a cross-client negative sampling for learning effectiveness. To evaluate our learning framework, we first construct a new OD-AS2 dataset, called Fed-NewsQA, based on NewsQA to simulate distributed clients with different genre/domain data. Experiment results shows that our learning framework can outperform its baselines and exhibit its personalization ability.

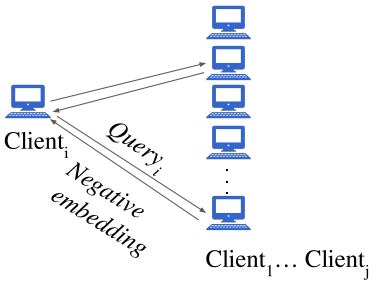
1 Introduction

Open-domain answer sentence selection (OD-AS2) aims to fetch relevant sentences from a large-scale collection given a query, which is also known as long answer in open-domain question answering (OD-QA). It has been attracting more and more interest from both academia and industry (Yang et al., 2018; Kwiatkowski et al., 2019) as it reaches a balanced granularity between coarse-grained passages (Nguyen et al., 2016) and fine-grained phrases (Kwiatkowski et al., 2019). Such balanced-granular answers can relieve crowd-sourcing burdens and satisfy most real-world scenarios.

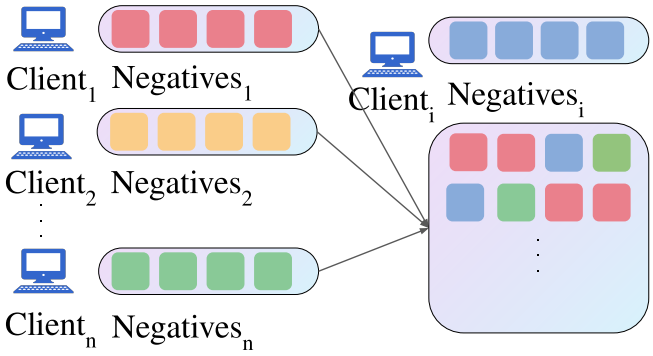
Advanced by surging pre-trained language models (Devlin et al., 2019; Liu et al., 2019), representation learning entered a new era and renders dense retrieval as a significant prerequisite across different solution paradigms (e.g., '*retrieval & read*') to OD-AS2. Built upon a dual-encoder (a.k.a. bi-encoder, two-stream encoder), dense retrieval represents both questions from users and sentences in the collections as dense vectors in the same semantic space, and measures question-sentence relevance via a lightweight metric, e.g., doc-product (Guu et al., 2020; Karpukhin et al., 2020).

As training an effective dense retrieval model requires sufficient data – both human-created positive question-answering pairs and a large-scale collection to support negative mining, it remains formidable challenges to directly apply the dense retrieval to the real-world industrial scenarios, e.g., in-house data inquiry, individual email searches, and personal intelligent assistants. The corpus (i.e., the labeled QA pairs and collections) in each client is usually too scarce and biased to train an effective model, while the corpus from each client cannot be uploaded to a central server for standard distributed learning for a privacy-preserving purpose.

To this end, we propose a new learning framework for **P**rivacy-preserving **D**istributed O**D-AS2**, called PDD-AS2. In particular, built upon a prevailing federated learning (FL) framework, FedAvg(McMahan et al., 2017), PDD-AS2 alleviates the data-scarcity problem along with two significant directions. On the one hand, while learning generic representation across clients via FL, we present a client-customized query encoding for personalization for client-specific query distribution. In line with dynamic hard negatives and query-side fine-tuning, it will largely improve the model's effectiveness. On the other hand, without access to other clients' collections to secure privacy, we propose a cross-client negative sampling strategy, called fed-negative, compatible with previ-



(a) The communication process of fed-negative.



(b) Negative Aggregation of fed-negative.

Figure 1: (a) The client sends its queries to some other clients and receive negative embeddings from these clients. (b) The client aggregates local negatives with received negatives to construct a negative subset.

ous strategies (e.g., in-batch, pre-batch, static hard negative sampling) to further boost the model.

To evaluate our learning framework, PDD-AS2, we propose to construct a new distributed OD-AS2 dataset based on NewsQA (Trischler et al., 2017) w.r.t. news story's genre.

In the experiments, we show that our PDD-AS2 framework can improve the performance of our baseline by 5%-15%. Clients with insufficient training data benefit from the model aggregation greatly. We also show that our fed-negative can improve the performance of PDD-AS2 framework by 1%-10% compared with the original negative sampling method. The main contribution of this work can be summarized as

- We highlight a promising setting of open-domain answer sentence selection (OD-AS2) for real-world industrial applications and propose a privacy-preserving distributed OD-AS2 (PDD-AS2) learning framework towards both personalization and effectiveness.
- We propose two key techniques, i.e., client-customized query encoding method and a cross-client negative sampling strategy, to effectively learn PDD-AS2 framework.
- We construct a new distributed OD-AS2 dataset upon NewsQA, dubbed Fed-NewsQA to evaluate the effectiveness of our framework and its baselines.

2 Methodology

In this section, we first introduce the preliminaries of our work. Then we present our proposed client-customized query encoding and cross-client negative sampling in our PDD-AS2 framework. Later, we detail the training process of our PDD-AS2 framework and our proposed Fed-NewsQA benchmark for evaluating our framework.

2.1 Preliminary

Task formulation. In line with existing works (Shen et al., 2017; Garg et al., 2020; Karpukhin et al., 2020; Zhan et al., 2021), we first formulate open-domain answer sentence selection (OD-AS2) under distributed setting as follows: For each client $c^i \in \mathbb{C}$ with its large-scale sentence collection $\mathbb{S}^i = \{s_1^i ... s_n^i\}$, it aims to fetch potential answer sentence(s) s_k^i from $\mathbb{S}^i$ that answers a given query $q \in \mathbb{Q}$. In the OD-AS2 setting, the sentence set $\mathbb{S}^i$ contains sentences from all passages in c^i. If no confusion is caused, we omit the superscript 'i' for a specific client in the remainder.

Usually, a query q and its answer sentence s_q^+ are often provided as positive training data in each client. Hence, it is necessary to sample a set of negative for q to construct , i.e.,

$$\mathbb{N}_q = \{d | d \sim P(\mathbb{S})\}, \quad (1)$$

where $P(\cdot)$ denotes a distribution over $\mathbb{S}$. For simplicity, we omit the query-specific subscript indicator, q.

Then, a contrastive learning framework is usually employed to learn an efficient retrieval model. In formal, a representation learning module is first used to embed q and each $s \in \{s^+\} \cup \mathbb{N}$ and then derive a probability distribution over $\{s^+\} \cup \mathbb{N}$.

That is,

$$P(\{s^+\} \cup \mathbb{N}|q;\Theta) = 1/Z \quad (2)$$
$$exp(< \mathrm{Enc}(q;\Theta^{(q)}), \mathrm{Enc}(s;\Theta^{(s)}) >$$

where $\Theta = \{\Theta^{(q)}, \Theta^{(s)}\}$, Z denotes $\mathrm{softmax}$ normalization term, Θ parameterizes a text encoder for a single vector representation, $<,>$ denotes a lightweight relevance metric (says, dot-product) for their similarity score. Here, $\Theta^{(q)}$ and $\Theta^{(s)}$, whether tied or not, compose a dual-encoder structure for efficient dense retrieval. Lastly, the training loss of contrastive learning can be defined to optimize Θ, i.e.,

$$L^{(\mathrm{ct})}(\mathbb{Q};\Theta) = -\sum_{q\in\mathbb{Q}} \log P(\mathrm{s} = s^+|q, \{s^+\} \cup \mathbb{N};\Theta), \quad (3)$$

where $P(\cdot|q;\Theta)$ denotes the probability distribution over $\{s^+\} \cup \mathbb{N}$ for q by Eq.(2).

Next, considering the distributed setting of OD-AS2, the overall training loss can be defined as

$$L(\{\mathbb{Q}^i\}_i;\{\Theta^i\}_i) = \sum_i L^{(\mathrm{ct})}(\mathbb{Q}^i;\Theta^i). \quad (4)$$

However, directly optimizing Eq.(4) cannot deliver a satisfactory performance for each client i since both labeled question-answering pairs and the collection are too scarce to effectively learn. Therefore, we adopt a popular federated learning method, FedAvg (McMahan et al., 2017), as the backbone of our framework. It will leverage the training data distributed in each client in a privacy-preserving way. We denote the weight of global model as Θ^{global}. For each $c \in \mathbb{C}$ with model weight Θ^i, we update Θ^i with a learning rate of α locally by

$$\Theta^i = \Theta^i - \alpha\nabla\mathcal{L}(\mathbb{Q}^i;\Theta^i), \quad (5)$$

where $\mathcal{L}$ is the loss function of local training objective defined in Eq.4. After local updates, each client sends their weights Θ^i to the central server. Central server aggregate the weights by

$$\Theta^{global} = \sum_{i=1}^{k} \frac{|\mathbb{D}_i|}{\sum_{i=1}^{k}|\mathbb{D}_i|}\Theta^i, \quad (6)$$

where k is the number of clients. Note that our PDD-AS2 framework is also compatible with other federated learning methods.

2.2 Fed-Negative: Cross-client Negatives

However, federated learning cannot fulfill negative samples' needs in terms of quality and quantity in some clients with few document collections. Building on this problem, we propose fed-negative: a cross-client negative sampling method inspired by dynamic negative sampling for introducing more diverse negative samples. As shown in the Figure 1, given a client c, we first encode q into representations by $\mathrm{Enc}(q;\Theta)$. Then we select a subset of clients from the whole client set as

$$C_s = Select(\{C\}), c \notin C_s, \quad (7)$$

where the select function can based on network condition or geography distance estimated by client's region. Then we send the query representation $\mathrm{Enc}(q;\Theta)$ to each client in C_s.

Once each client receive the query, they did a similarity search on their own sentence embedding matrix to retrieve top *n* sentences embeddings and send them back to c. c choose top *n* negatives from all negatives by the similarity score as

$$N^{fed} = TopK(\{(N_{c_k})\}), c_k \in C_s \quad (8)$$

where N_{c_k} is the negative set of q sampled in client c_k.

2.3 Client-customized Query Encoding

On top of fed-negative, we propose client-customized query encoding inspired by query-side fine-tuning. We aim to provide each client with a personalized query encoder to resolve miscellaneous queries. For this purpose, we personalize $\mathrm{Enc}(q;\Theta)$ with local training while fixing the $\mathrm{Enc}(s;\Theta)$. $\mathrm{Enc}(s;\Theta)$ shares a global weight among all clients. In this stage, we utilize our proposed fed-negative for diverse negative samples.

Training objective. To learn a personalized query encoder, we apply the constrative loss defined in Eq.4. Formally, given a query q and its gold answer s^+, we first sample the negative set N^{fed} defined by Eq.8. Therefore, we only update the weight Θ of query encoder with loss function defined in Eq.4

2.4 Training Pipeline of PDD-AS2

Finally, we introduce the overall training pipeline of our PDD-AS2 framework. As shown in Figure 2, we organize our training procedure as two stages adapted from some prevailing works (Zhan et al.,

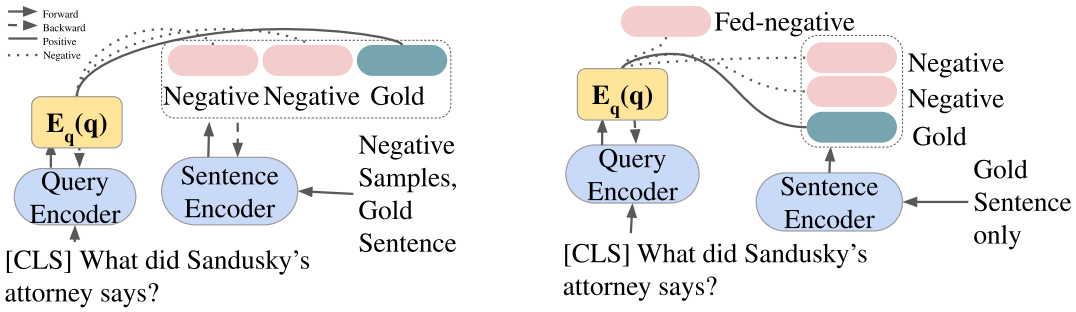


Figure 2: (a) Train query encoder $\mathrm{Enc}(q;\Theta)$ and sentence encoder $\mathrm{Enc}(s;\Theta)$ with Static hard-negative sampling (b) Personalize the query encoder $\mathrm{Enc}(q;\Theta)$ with fed-negative

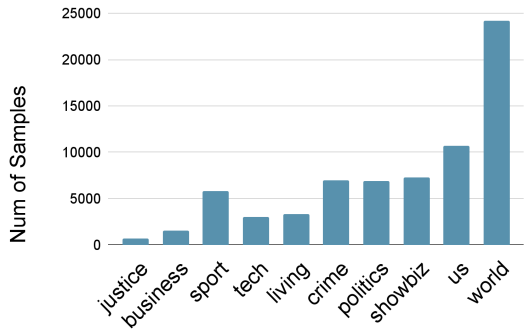


Figure 3: Statistics of each genre in our Benchmark

2021; Karpukhin et al., 2020): (Stage 1) ***Federated Static negative training***: we train the encoders with static hard negative sampling N^{static} under FedAvg. Due to the instability of the model in the early training stage, we initially sample BM25 negatives N^{BM25} to warm up the model following some works (Zhan et al., 2021; Gao and Callan, 2022). We update both $\mathrm{Enc}(q;\Theta)$ and $\mathrm{Enc}(s;\Theta)$ by $\mathcal{L}$ defined in Eq.4. The overview of the federated framework is illustrated in Algorithm.1. (Stage 2) ***Query encoder personalization***: Continual from first stage, we samples N^{fed} defined in section 2.2 to train a client-customized query encoder follows section 2.3.

2.5 Fed-NewsQA: A Multi-client OD-AS2 Benchmark

For better evaluate our method in distributed setting, we propose a multi-client OD-AS2 benchmark based on NewsQA. Recent open-domain question answering works often use datasets such as SQuAD (Rajpurkar et al., 2016), TREC (Wang et al., 2007), WebQuestions (Berant et al., 2013), Natural Questions (Bird et al., 2009) in their experiments. However, we propose to use NewsQA (Trischler et al., 2017) as our original dataset for two main reasons.

First, to better mimic the difference between each client's personal documents and the data scarcity problem in the real-world cases, we propose to split the dataset into different genres for simulating different clients. Among all these datasets, we found that NewsQA meets our requirements perfectly. We split the dataset into different genres directly from the web-link of each passage. We choose ten genres from NewsQA since the rest genres do not have enough numbers of samples in the dev/test set. Each of these genres represents a different client in our Federated setting. The statistics of each genre is shown in the Figure 3.

Second, NewsQA significantly outnumbers some other datasets on the distribution of the more difficult reasoning questions, such as SQuAD (Trischler et al., 2017). We believe inferencing and reasoning queries are essential to open-domain question answering in real-world cases.

2.6 Retrieval Schemes

Our model is compatible with two retrieval schemes: sentence-level retrieval and passage-level retrieval. For sentence-level retrieval, we retrieve the top sentences follow the probability distribution defined in Eq.2. For passage-level retrieval, based on the fact that sentences are extracted from their source passages. We retrieve the passage with highest relevance score as

$$f(p,q) := \max_{s \in p}\{< \mathrm{Enc}(q;\Theta), \mathrm{Enc}(s;\Theta) >\} \quad ,\forall s \in \mathbb{S}. \tag{9}$$

The additional cost of sorting sentence scores can be ignored. Therefore the inference speed of our sentence-based passage retrieval is the same as for sentence retrieval.

Algorithm 1: The federated learning framework of PDD-AS2 in stage 1 training

Input: Clients set $\mathbb{C}$, Training set D_i on client $_i$, global model weight Θ^{global}, learning rate α

Function `Server execute`**:**
  initialize Θ^i with Θ^{global};
  for *round t=1,2...* **do**
    for *each client* $c_i \in \mathbb{C}$ ***in parallel*** **do**
      $\Theta^i \leftarrow ClientUpdate(c_i, \Theta^i, D_i)$
    end
  end
  $\Theta^{global} = \sum_{i=1}^{k} \frac{D_i}{\mathbb{D}} \Theta^i$
Function `ClientUpdate`(c_i, Θ^i, D_i)**:**
  `// execute on client` c_i
  for *batch b in* D_i **do**
    $\Theta^i \leftarrow \Theta^i - \eta \nabla \mathcal{L}(q; \Theta^i)$
  end

3 Experiments

3.1 Setup

Baselines. We conduct experiments[1] to compare the performance of our method with several dense retrieval methods, including: (1) dense retrieval trained with random negative (Huang et al., 2020) (2) dense retrieval trained with BM25 negative (Gao et al., 2021); (3) dense retrieval trained with STAR (Zhan et al., 2021). In personalization stage, we compare our proposed fed-negative to dynamic hard-negatives in (Zhan et al., 2021).(4) a simple sparse retriever constrcuted by BM25.

We also includes a upper bound baseline trained on a central server which shows the degree of performance drop brought by the distributed setting.

Implementation. We use pre-trained DistilBERT (Sanh et al., 2019) by huggingface as our model. We use AdamW with a learning rate of 3e-5. We use Faiss (Johnson et al., 2021) to perform the similarity search. We use open-sourced BM25 model in training. Queries and sentences are truncated to a maximum of 32 tokens and 512 tokens, respectively. We represent query embeddings simply by the $[CLS]$ token and sentence embeddings by the average pooling of word embeddings in the sentence.

The detail of our training procedure is described as follows: In the federated static negative training, we pair each query with BM25 negatives and gold-negatives with a batch size of 8 in the warm-up stage. Then we replace them with static hard-negatives. To demonstrate the influence of numbers of negatives, we also experiment with settings with different numbers of negatives. We enable in-batch negative in this stage. We implemented vanilla FedAvg as our Federated learning framework. We aggregate local weights after each epoch.

In personalized query encoder training, we pair each query with dynamic hard negatives or fed-negatives with a batch size of 32. To demonstrate the influence of numbers of negatives, we also experiment on settings with different numbers of negatives. We enable in-batch negatives in this stage.

We report two levels of metrics in our experiments: sentence-level and passage-level. The retrieval procedure of both levels is defined in section 2.6. In both levels, we report the MRR@10, Recall@1,20,100 scores.

3.2 Experiment Results

The main result of our experiments is shown in Table 1. We conclude with two main findings from the results. First, compared with the dense retrieval baselines trained on a single client, our PDD-AS2 outperformed all other methods. This is because the number of documents in some clients are very restricted. Our method can leverage training data on each client in a privacy-preserving way. Therefore, our federated method can achieve better performance than non-Federated methods.

Second, our personalization method with fed-negative can outperform the method with local dynamic hard negatives. This is because the scarcity of training data in some clients can lead to a much worse hard-negative sampling result. Compared with static hard negative sampling, the training of client-customized query encoder introduces far more negative samples, strengthening the need of hard negatives in terms of quality and quantity. Our method alleviates the problem by leveraging diverse hard negatives on other clients in a privacy-preserving way.

3.3 Influence of Numbers of Negatives

We explore the influence of $num\_negatives$ in our setting. We experiment with the combinations of different numbers of negatives used in each method. The result of different $num\_negatives$ is showed in Appendix A. We show the impact of

[1] We will make our data and codes public.

Table 1: Results on our Fed-NewsQA Benchmark.

Models	Sentence-level Retrieval				Passage-level Retrieval			
	MRR@10	R@1	R@20	R@100	MRR@10	R@1	R@20	R@100
Upper Bound								
Central-training	0.338	0.284	0.629	0.781	0.502	0.447	0.553	0.821
Sparse Retriever								
BM25	0.172	0.152	0.343	0.533	0.343	0.288	0.345	0.598
Dense Retriever								
dense retrieval-Random Neg	0.194	0.171	0.466	0.62	0.376	0.323	0.401	0.702
dense retrieval-Bm25 Neg	0.188	0.151	0.475	0.639	0.353	0.303	0.388	0.679
dense retrieval-STAR	0.232	0.190	0.535	0.679	0.403	0.350	0.421	0.709
Dense Retriever: Ours								
PDD-AS2	0.261	0.217	0.546	0.695	0.429	0.395	0.479	0.745
+client-customized query encoding	0.289	0.232	0.556	0.711	0.445	0.414	0.489	0.75
+client-customized query encoding with fed-negative	0.309	0.252	0.577	0.72	0.458	0.431	0.504	0.762

$num\_negatives$ on both stages of training separately. The maximum number of hard-negatives we can test in stage 1 training is limited due to GPU RAM cost. For BM25 negative sampling and static hard-negative sampling, we train the model with our PDD-AS2 framework from the beginning of our training procedure. In experiments of stage 2 training with fed-negative, we continue our training from the model weights trained in previous steps, which follows our training procedure.

We have two findings from the results. First, we found that insufficient numbers of negative samples can lead to much worse performance. This is intuitive since the model saw fewer numbers of samples during training. Second, client-customized query encoder training can benefit more from the larger amount of negatives. Our experiment shows that the optimal number for BM25 negative sampling is not very large. BM25 negative sampling cannot leverage the larger amount of negatives effectively. However, due to the limitation of hardware resources, we cannot test on larger numbers of negatives in stage 1 training.

Meanwhile, client-customized query encoder can be steadily improved while feeding much more negatives compared with stage 1 training. This result indicates the need for introducing more hard-negatives with higher quality in stage 2 training, further proving the effectiveness and necessity of our fed-negative. Whats more, the computational cost does not scale with the $num\_negatives$. As a consequence, client-customized query encoder can benefit from fed-negative with little cost.

3.4 Influence of Training Data Size

In this section, we first explore whether our PDD-AS2 can effectively handle the data scarcity problem on each client by leveraging data on different clients. In training, we select different ratios of data randomly. We present the sentence-level R@1 score on our Fed-NewsQA in Figure 5. Compared with single-client training, the PDD-AS2 can achieve higher accuracy in all data ratio settings. Moreover, as the ratio of training data on each client decreases, the data scarcity problem in single-client is more serious. As a consequence, PDD-AS2 can bring about a more signification performance improvement over single-client training.

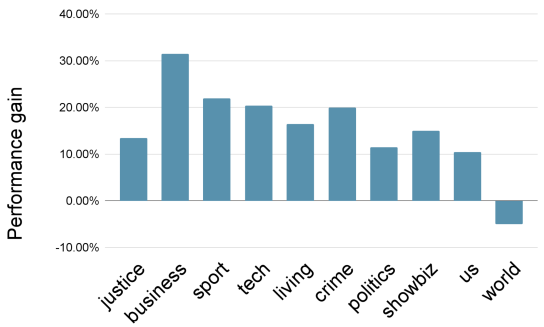


Figure 4: Performance improvement of each client in PDD-AS2 stage 1 training with FedAvg compared with single-client training

Also, we explore to what extent each client benefits from the PDD-AS2. We show the performance improvement in sentence-level R@1 on Fed-NewsQA of each client in Figure 4. We found that clients with fewer training data can benefit more from the PDD-AS2 framework. These results indicate that our framework can effectively leverage the training data on different clients. However, performance on some clients with a larger amount of training data was decreased while applying our framework, implying the need for personalization in this scenario.

3.5 Influence of query hubness

However, retrieving all top-k hard negatives from similarity search or BM25 engine can lead to a performance drop in some scenarios. The reason is that, not every possible answer for a given query q_i has been labeled as positive. This is very intuitive since most machine reading comprehension datasets only label the answer of the query, which is only in its context passage. However, in open-domain question answering, possible answers from all passages must be labled as positive. This problem is more severe when the query is not specific and precise.

As a consequence, for each q_i, if we retrieve all top-k sentences as negative, we actually harm the performance of the model. We conduct a case study in Appendix B. The case study shows that whether or not the query is specific and precise, the top-k negatives often contain possible answers that were not labeled as positive. We refer to this problem as 'query hubness'. To alleviate this problem, we uniformly samples *n* negatives from *k* candidate where k≫n in our approach. This approach yields better results when we choose a correct k. The difference in model performance in different k is shown in Table 2. However, more theoretical insight is needed in query hubness problem.

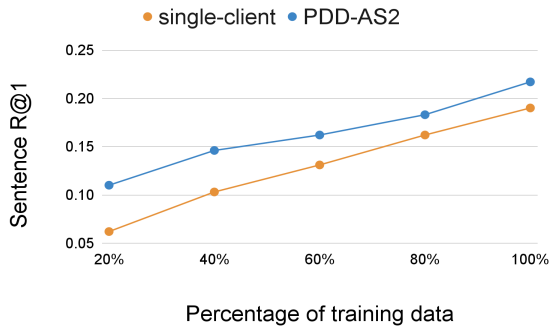


Figure 5: Influence of training data in Sentence R@1 size

Table 2: Different k while sampling 10 negatives

Method	Sentence R@1	Passage R@1
k=10	0.121	0.235
k=50	0.202	0.379
k=100	0.211	0.352
k=300	0.217	0.395

Table 3: Perplexity of gpt-2 on our dataset.

Method	Perplexity
Without training	36.3
CLM without embedding	25.9
CLM with sentence embedding	25.6

3.6 Privacy

When transferring sentence embeddings between clients, one key concern is whether the user's privacy would be leaked. However, no work is dedicated to restoring private information from merely sentence embeddings. In order to measure the risk involved, we conducted an experiment to detect whether our transmitted sentence embedding contained information related to the original text.

In this experiment, we used GPT-2, a model that performs well on text generation tasks. In the first part of experiment, we trained GPT-2 on the language modeling task on our dataset and measured its perplexity on the test set. In the second part of the experiment, we add the sentence embedding generated by the previously trained sentence encoder in PDD-AS2 to the training and testing procedure. In detail, we feed the sentence embeddings into the gpt-2 as key-value pairs together with the text input. After receiving the input, the model tries to establish the connection between the embedding and the actual sentence it represents through the self-attention structure.

Table 3 showed no significant difference in the perplexity between the two groups of experiments. The group with sentence embeddings has a slightly lower perplexity on the testset. Also, we show that the embedding group has a lower loss over the training process in the Appendix C. However, these differences are not statistically significant. To further demonstrate that we cannot obtain private information from the sentence embeddings, we let gpt-2 generate actual sentences directly from their corresponding embeddings without any input and prompts. We show the result in the Appendix D.

We found that gpt-2 could not restore the actual sentence from the sentence embeddings only. Sentence embeddings did have an impact on the generated results. However, these effects are seemingly random and irrelevant to the actual sentence.

4 Related Work

4.1 Dense retrieval

Dense retrieval has recently become a popular topic in industry and academia due to its advantages of both latency and performance. The essential to the success of dense retrieval is its leverage of negative samples to train the model. The early stage of research only uses random negatives to train dense retrieval models(Huang et al., 2020). Recently, researchers applied hard-negatives to train the model. Hard negatives refer to samples that are semantically similar to postive samples but are in fact negatives. Some studies (Zhan et al., 2021) demonstrate that most of the boosts in the training phase come from these hard negatives. Some researchers use BM25 to retrieve hard negatives (Karpukhin et al., 2020; Gao et al., 2021). Some others use static hard-negatives fixed during the entire training or an epoch(Guu et al., 2020; Xiong et al., 2020). (Zhan et al., 2021)propose a dynamic hard-negative method which called query-side fine-tuning.

However, insufficient training data would result in severe performance degradation. (Karpukhin et al., 2020) shows around a 10% performance difference in top-5 passage retrieval due to an insufficient number of negative samples. (Qu et al., 2021) found it is beneficial to increase the number of random negatives in the mini-batch. When using only 10% percent of training data, the normal dense retrieval model's performance can drop by 20% (Lu et al., 2021). In this work, we propose an open-domain question answering method empowered by Federated learning to alleviate the problem. Also, we further explore the potential of query-side fine-tuning for personalization.

4.2 Answer Sentence Selection

Answer Sentence Selection task was defined by (Wang et al., 2007). This task aims to select a sentence that correctly answers the question from a set of sentence candidates. This task has been studied by many works (Shen et al., 2017; Tran et al., 2018; Yoon et al., 2019; Garg et al., 2020). However, in a typical AS2 task, the model is required to select sentences from several candidates. In our Open-domain Sentence Selection setting, the number of candidates can scale up to one million, which significantly increases the task's difficulty.

4.3 Federated Learning

Federated learning was proposed by (McMahan et al., 2017) as a privacy-preserving solution to leverage personal data on different clients. All the training data is stored locally on each client. Each client uses local data to train its own model locally. After each round of training or a certain training time, these clients allow other clients to learn from the training data of this client with privacy protection by sharing the model weights or gradients.

Recently, some researchers have applied Federated learning to different NLP tasks (Ge et al., 2020; Hardy et al., 2017; Jiang et al., 2019). In these scenarios, user data are scattered in different devices (e.g., cell phones) or different facilities (e.g., banks, hospitals). Moreover, these data cannot be uploaded to the central server because of user privacy, such as user's input method records, medical records, etc. However, the combination of Federated learning of open-domain question answering has not been studied yet.

5 Conclusion

In this paper, we propose an Privacy-preserving Distributed OD-AS2 method, dubbed PDD-AS2. Our method utilizes training data on different clients while eliminating the need to transfer the raw data between clients. The training process of our approach is two-stage. In the first stage, we train both query encoder and sentence encoder with static hard-negatives under a federated framework. In the second stage, we personalize a client-customized query encoder for each client. We also propose a new negative sampling method called fed-negative. In fed-negative, we introduce diverse negatives from other clients to boost the training. We further test our method on a new Federated Open-domain Sentence Selection benchmark based on NewsQA. This Benchmark better mimics the real-world cases than other benchmarks in data distribution and query types.

The experiment results show that our method can effectively improve the performance of open-domain answer sentence selection under distributed settings by leveraging training data on different clients in a privacy-preserving way. We prove that not every client can benefit from the Federated learning, which indicates the need for personalization in such scenario. As a solution, we provide each client with a client-customized query encoder which handles miscellaneous queries.

6 Limitations

However, we did not discuss all possible privacy leakage methods due to length limitations. For example, users can get the information in query embeddings from other users in fed-negative stage by comparing the similarity of their own query embeddings with others. Meanwhile, attackers can infer the training data from the gradient updates from the word embedding layer in the shared model weight in Federated learning stage.

Furthermore, the communication cost between each client is not included in the discussion. Many studies indicate that the size of the communication cost directly impacts model performance. In our settings, different clients need to transfer word embeddings during training. More experiment is needed to explore the impact of communication cost on our proposed fed-negative method.

Finally, the size of participant clients in our experiment was limited to 10 due to limitations in computational resources. However, in a real-world setting, the number of clients participating in federal learning would be much larger than the number of participants in the experiment.

References

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

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. 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.

Luyu Gao, Zhuyun Dai, Tongfei Chen, Zhen Fan, Benjamin Van Durme, and Jamie Callan. 2021. Complement lexical retrieval model with semantic residual embeddings. In *Advances in Information Retrieval*, pages 146–160, Cham. Springer International Publishing.

Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7780–7788.

Suyu Ge, Fangzhao Wu, Chuhan Wu, Tao Qi, Yongfeng Huang, and Xing Xie. 2020. Fedner: Privacy-preserving medical named entity recognition with federated learning. *ArXiv*, abs/2003.09288.

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.

Stephen Hardy, Wilko Henecka, Hamish Ivey-Law, Richard Nock, Giorgio Patrini, Guillaume Smith, and Brian Thorne. 2017. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *ArXiv*, abs/1711.10677.

Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020. Embedding-based retrieval in facebook search. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2553–2561.

Di Jiang, Yuanfeng Song, Yongxin Tong, Xueyang Wu, Weiwei Zhao, Qian Xu, and Qiang Yang. 2019. Federated topic modeling. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 1071–1080, New York, NY, USA. Association for Computing Machinery.

J. Johnson, M. Douze, and H. Jegou. 2021. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(03):535–547.

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.

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

research. *Transactions of the Association of 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*, abs/1907.11692.

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 the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2780–2791, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR.

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.

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.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Gehui Shen, Yunlun Yang, and Zhi-Hong Deng. 2017. Inter-weighted alignment network for sentence pair modeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1179–1189, Copenhagen, Denmark. Association for Computational Linguistics.

Quan Hung Tran, Tuan Lai, Gholamreza Haffari, Ingrid Zukerman, Trung Bui, and Hung Bui. 2018. The context-dependent additive recurrent neural net. 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 1274–1283, New Orleans, Louisiana. Association for Computational Linguistics.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. 2007. What is the Jeopardy model? a quasi-synchronous grammar for QA. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 22–32, Prague, Czech Republic. Association for Computational Linguistics.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv preprint arXiv:2007.00808*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*.

Seunghyun Yoon, Franck Dernoncourt, Doo Soon Kim, Trung Bui, and Kyomin Jung. 2019. A compare-aggregate model with latent clustering for answer selection. *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*.

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.

Appendix A Different numbers of negatives in training

We trained different models with different number of negatives. The results is shown in Table 4

Appendix B Query hubness

We present the case study of query hubness examples in Table 6

Table 4: Different num_negative in Training

Models	Sentence-level Retrieval				Passage-level Retrieval			
	MRR@10	**R@1**	**R@20**	**R@100**	**MRR@10**	**R@1**	**R@20**	**R@100**
Dense Retriever with BM25 negatives								
num_negative=2	0.143	0.123	0.302	0.489	0.310	0.247	0.311	0.582
num_negative=8	0.172	0.151	0.343	0.533	0.343	0.288	0.345	0.598
Dense Retriever with STAR								
num_negative=2	0.201	0.160	0.506	0.655	0.352	0.305	0.379	0.705
num_negative=8	0.232	0.191	0.535	0.679	0.403	0.350	0.421	0.709
PDD-AS2								
num_negative=2	0.242	0.193	0.516	0.645	0.392	0.354	0.432	0.719
num_negative=8	0.261	0.217	0.546	0.695	0.429	0.395	0.479	0.745
+client-customized query encoding								
num_negative=10	0.272	0.233	0.557	0.705	0.431	0.415	0.487	0.746
num_negative=200	0.289	0.251	0.576	0.711	0.445	0.434	0.489	0.75

Table 5: Case study of retrieved hard-negatives

	Case 1	Case 2
Question	What did the lawyer say	Who will star in the upcoming ABC pilot "The Manzanis"?
Gold answer	Murray defense lawyer Michael Flanagan, who was in court to defend Dr. White Wednesday, said after the hearing that he believed Murray should be eligible for early release if he is given prison time	When Kirstie Alley cleared the 100 lb. weight-loss hurdle this summer, it was time for a big, fat celebration.
Hard-negative 1	In addition, Anthony's attorney Charles Greene asserted he would also invoke the Fifth Amendment on her behalf if questioning delved into the 2008 death of her 2-year-old daughter, Caylee.	And she's ready for her next challenge: "What I'm looking for is to be madly, deeply in love," says Alley, who will also star in the upcoming ABC pilot, "The Manzanis."
Hard-negative 2	CNN) – Attorneys representing Casey Anthony invoked her Fifth Amendment right against self-incrimination 60 times during a deposition given in a civil suit against her, according to a transcript of the proceedings.	Kirstie Alley said she's going to start dating "butt-ugly men" on an episode of "The Ellen DeGeneres Show" airing Friday.

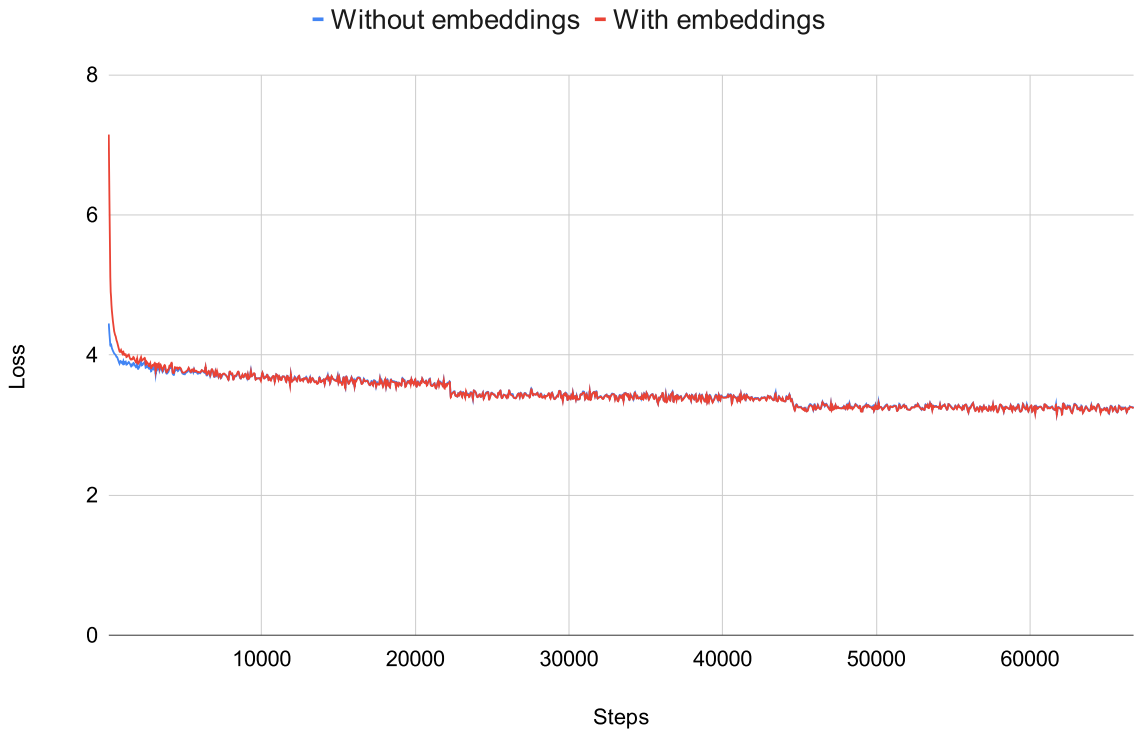


Figure 6: The difference between performing CLM training with and without sentence embeddings using gpt-2

Appendix C Quantitive results with privacy leakage experiment

We trained gpt-2 with casual language modeling on our dataset. The training process is shown in 6 In the end, the group with embedding has a loss of 3.239, while the group without embedding has a loss of 3.245. Their corresponding perplexity is shown in Table 3.

Appendix D Qualitative results with privacy leakage experiment

We randomly select a few sentences from the dataset and input the corresponding sentence embeddings as $past\_key\_values$ into gpt-2. We applied beam search while generating texts. Table 6 shows the results of the model's outputs from decoding the embeddings. The conclusion is that the model can not decode any private information from the sentence embeddings.

Table 6: Case study of sentence-embeddings decoding

Original Sentences	Generated Sentences
Four Australian troops have now died in the conflict in Afghanistan.	"It's not the first time that we've had "It's not the first time that we've seen a "It's not the first time that we've had to "It's not the first time that we've seen the
It made my stomach turn," Bertha Lewis, chief executive officer of ACORN, told reporters at the National Press Club in Washington.	"I think it's important? very important? Very difficult to the one. I think. is, part of me. I the to blame, I don't blame my "I think it's important? very important? Very difficult to the one. I think. is, part of me. I the to blame, I don't people, "I think it's important? very important? Very difficult to the one. I think. is, part of me. I the to blame, I don't people who "I think it's important? very important? Very difficult to the one. I think. is, part of me. I the to blame, I don't blame the
Read the story at the WRTV web site	CNN's a great-school program that's not CNN's a great-school program for example of CNN's a great-school program," said reason CNN's a great-school program for example: