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

Weikuan Wang[1], Tao Shen[2], Michael Blumenstein[3], and Guodong Long[4] ✉

[1] University of Technology Sydney weikuan.wang@student.uts.edu.au
[2] University of Technology Sydney tao.shen@uts.edu.au
[3] University of Technology Sydney Michael.Blumenstein@uts.edu.au
[4] University of Technology Sydney guodong.long@uts.edu.au

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, fewer 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 show that our learning framework can outperform its baselines and exhibit its personalization ability.

Keywords: Information Retrieval · Federated Learning · Personalization

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). Its interest is growing from both academia and industry [18] as it reaches a balanced granularity between coarse-grained passages [25] and fine-grained phrases [18]. Such balanced-granular answers can relieve crowdsourcing burdens and satisfy most real-world scenarios.

Advanced by surging pre-trained language models [5], representation learning entered a new era and rendered 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 [10, 17].

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 a 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[24], PDD-AS2 alleviates the data-scarcity problem along with two significant directions. On the one hand, our framework learns generic representation across clients via FL. On the other hand, we present a client-customized query encoding for personalization and client-specific query distribution. In line with dynamic hard negatives and query-side fine-tuning, it will significantly improve the model's effectiveness. To evaluate our learning framework, PDD-AS2, we propose to construct a new distributed OD-AS2 dataset based on NewsQA [33] 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.

The main contributions 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 a key technique, i.e., client-customized query encoding method 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 Related Work

2.1 Open-domain question answering

Open-domain question answering (OD-QA) answers a given question using a collection of documents. It does not require a specified context. Compared with Machine Reading Comprehension (MRC), which is another popular task in NLP,

OD-QA is more in line with human behaviors. MRC can only retrieve answers from a given context. Therefore, OD-QA has a very promising future in industry applications.

Traditional OD-QA systems often consists of a multi-stage method, i.e., query analysis, context retrieval, and answer retrieval [26, 1, 12]. DrQA [4] is the first work to incorporate neural MRC models into OD-QA. A new diagram of OD-QA is proposed. This diagram is a two-stage retriever-reader diagram, which combines IR methods like TF-IDF with a neural MRC model. Nowadays, the retriever-reader diagram is studied in many works [11, 22, 20, 21, 30] and proved to outperform significantly traditional methods in performance and efficiency.

However, the two-stage structure of this retriever-reader has a big problem in practical use. Whenever the model receives a query, it requires a complex and heavy reader model to encode several or even dozens of long contexts in real time, which is unacceptable in practice.

2.2 Dense retrieval

Dense retrieval has recently become a popular topic in industry and academia due to its advantages of both latency and performance. The key 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 [14]. Recently, researchers applied hard-negatives to train the model. Hard negatives refer to samples that are semantically similar to positive samples but are in fact negatives. Some studies [37] demonstrate that most of the boost in the training phase come from these hard negatives. Some researchers use BM25 to retrieve hard negatives [17, 7]. Some others use static hard-negatives fixed during the entire training or an epoch [10, 35]. [37] propose a dynamic hard-negative method which called query-side fine-tuning.

However, insufficient training data would result in severe performance degradation. [17] shows around a 10% performance difference in top-5 passage retrieval due to an insufficient number of negative samples. [27] found that it is beneficial to increase the number of random negatives in the mini-batch. When using only 10% of training data, the normal dense retrieval model's performance can drop by 20% [23]. 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.

2.3 Answer Sentence Selection

The Answer Sentence Selection task was defined by [34]. This task aims to select a sentence that correctly answers the question from a set of sentence candidates. This task has been studied in many works [31, 32, 36, 8]. 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.

2.4 Federated Learning

Federated learning was proposed by [24] 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 [9, 13, 15]. 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 due to privacy concerns related to items such as users' input method records, medical records, etc. However, the combination of Federated learning of open-domain question answering has not been studied yet.

3 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.

3.1 Preliminary

Task formulation. In line with existing works [31, 8, 17, 37], 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 negative samples, i.e.,

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

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. Formally, 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 $\{s^+\} \cup \mathbb{N}$. Specifically,

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

where $\Theta = \{\Theta^{(q)}, \Theta^{(s)}\}$, Z denotes softmax normalization term, Θ parameterizes a text encoder for a single vector representation, $<,>$ denotes a lightweight relevance metric (say, a 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.,

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

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

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

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

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 [24], 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

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

where 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

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

where k is the number of clients, $\mathbb{D}_i$ denotes the volume of the dataset on each client. Note that our PDD-AS2 framework is also compatible with other federated learning methods.

3.2 Fed-Negative: Cross-client Negatives

However, federated learning cannot fulfill the needs of negative samples in terms of quality and quantity for 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. 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

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

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

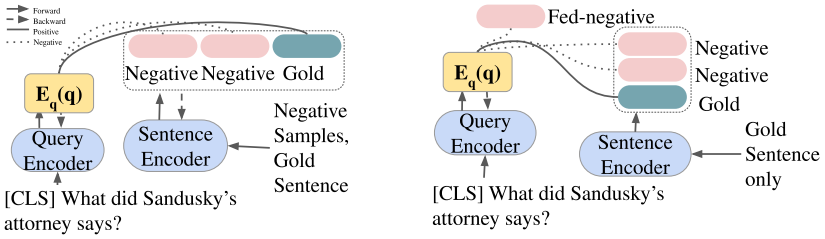

Fig. 1: (a) Train query encoder $\text{Enc}(q;\Theta)$ and sentence encoder $\text{Enc}(s;\Theta)$ with Static hard-negative sampling (b) Personalize the query encoder $\text{Enc}(q;\Theta)$ with fed-negative

Once each client receives the query, they perform a similarity search on their own sentence embedding matrix to retrieve the top n sentence embeddings and send them back to c. c chooses the top n negatives from all negatives based on the similarity score as

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

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

3.3 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 $\text{Enc}(q;\Theta)$ with local training while fixing the $\text{Enc}(s;\Theta)$. $\text{Enc}(s;\Theta)$ shares a global weight among all clients. In this stage, we utilize our proposed fed-negative method to generate 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 the query encoder with the loss function defined in Eq.4

3.4 Training Pipeline of PDD-AS2

Finally, we introduce the overall training pipeline of our PDD-AS2 framework. As shown in Figure 1, we organize our training procedure into two stages, adapted from some prevailing works [37, 17]: (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 the approach of some works [37, 6]. We update both $\text{Enc}(q;\Theta)$ and $\text{Enc}(s;\Theta)$ by $\mathcal{L}$ defined in Eq.4. The overview of the federated framework is illustrated in Algorithm.1. (Stage 2) ***Query encoder personalization***: Continuing from the first stage, we sample N^{fed}, as defined in section 3.2, to train a client-customized query encoder following the method described in section 3.3.

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

To better evaluate our method in a 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 [28], TREC [34], WebQuestions [2], Natural Questions [3] in their experiments. However, we propose to use NewsQA [33] as our original dataset for two main reasons.

Algorithm 1: PDD-AS2: Federated Static negative training

Input: Clients set $\mathbb{C}$, Training set D_i on client c_i, global model weight Θ^{global}, learning rate α
Initialize: global model Θ^{global}; .
for $r = 0, 1, \ldots, R$ **do**
  for Client $c_i \in \mathbb{C}$ **in parallel do**
    Initialize local model $\Theta^i \leftarrow \Theta$.
    for batch b in D_i **do**
      Send queries $q_b \in b$ to other clients $c_j \in \mathbb{C}$
      Receive negative samples $\mathbb{N}_{q_b}$
      $\Theta^i \leftarrow \Theta^i - \eta \nabla \mathcal{L}(s^+ \cup \mathbb{N}; q; \Theta^i)$
    end for
  end for
  Server updates Θ by global aggregation
end for

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 find 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 remaining genres do not have enough number of samples in the dev/test set. Each of these genres represents a different client in our Federated setting. The statistics of each genre are shown in the Figure 2.

Second, NewsQA significantly outnumbers some other datasets on the distribution of the more difficult reasoning questions, such as SQuAD [33]. We believe inferencing and reasoning queries are essential for open-domain question answering in real-world cases.

3.6 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

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

where $s \in p$ represents the set of sentences in a given passage p. The additional cost of sorting sentence scores can be ignored [19]. Therefore, the inference speed of our sentence-based passage retrieval is the same as for sentence retrieval.

4 Experiments

4.1 Setup

Baselines. We conduct experiments[5] to compare the performance of our method with several dense retrieval methods, including: (1) dense retrieval trained with random negative [14] (2) dense retrieval trained with BM25 negative [7]; (3) dense retrieval trained with STAR [37]. In personalization stage, we compare our proposed fed-negative to dynamic hard-negatives in [37].(4) a simple sparse retriever constructed by BM25.

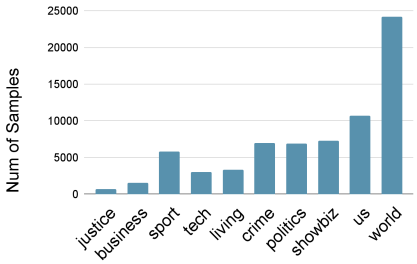

Fig. 2: Statistics of each genre in our Benchmark

Implementation. We use pre-trained DistilBERT [29] by Hugging Face as our model. We use AdamW with a learning rate of 3e-5. We use Faiss [16] 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 using the $[CLS]$ token, and we represent sentence embeddings using the average pooling of word embeddings in the sentence.

The details 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 the 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 3.6. In both levels, we report the MRR@10, Recall@1,20,100 scores.

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**

4.2 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 is 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 the client-customized query encoder introduces far more negative samples, strengthening the need for 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.

4.3 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 shown in Table 2. We show the impact of *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

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

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. What's 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.

Table 2: 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**

4.4 Influence of Training Data Size

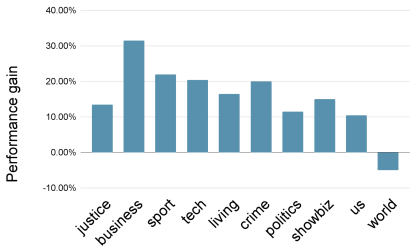

Fig. 3: Difference of performance gain of each client on Sentence R@1

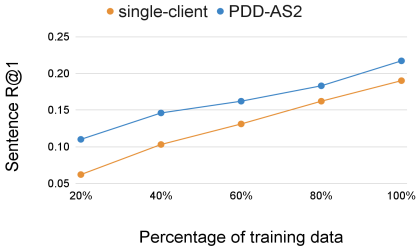

Fig. 4: Difference of performance in training dataset size on Sentence R@1

In this section, we first explore whether our PDD-AS2 can effectively handle the data scarcity problem in each client by leveraging data from 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 4. Compared with single-client training, the PDD-AS2 can achieve higher accuracy in all data ratio settings. Moreover, as the ratio of training data in each client decreases, the data scarcity

problem in single-client training becomes more serious. As a consequence, PDD-AS2 can bring about a more significant performance improvement over single-client training.

We also 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 3. We found that clients with less 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 decreased when applying our framework, implying the need for personalization in this scenario.

Table 3: 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 4: Perplexity of gpt-2 on our dataset.

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

4.5 Privacy

When transferring sentence embeddings between clients, one key concern is whether the user's privacy would be leaked. However, no work has been dedicated to restoring private information from mere sentence embeddings. In order to measure the risk involved, we conducted an experiment to detect whether our transmitted sentence embeddings 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 the experiment, we trained GPT-2 on the language modeling task using our dataset and measured its perplexity on the test set. In the second part of the experiment, we added the sentence embeddings 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 4 shows no significant difference in the perplexity between the two groups of experiments. The group with sentence embeddings has slightly lower perplexity on the test set. 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 Table 4.5.

We found that GPT-2 could not restore the actual sentence using only the sentence embeddings. Sentence embeddings did have an impact on the generated

results. However, these effects are seemingly random and irrelevant to the actual sentence.

Table 5: 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 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
Read the story at the WRTV web site	CNN's a great-school program that's not

5 Conclusion

In this paper, we propose a 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 enhance the training. We further test our method on a new Federated Open-domain Sentence Selection benchmark based on NewsQA. This benchmark better mimics real-world cases than other benchmarks in terms of 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 Federated learning, which indicates the need for personalization in such a scenario. As a solution, we provide each client with a client-customized query encoder which handles miscellaneous queries.

References

1. Allam, A.M.N., Haggag, M.H.: The question answering systems : A survey . (2016)

2. Berant, J., Chou, A., Frostig, R., Liang, P.: Semantic parsing on Freebase from question-answer pairs. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1533–1544. Association for Computational Linguistics, Seattle, Washington, USA (Oct 2013), https://aclanthology.org/D13-1160
3. Bird, S., Klein, E., Loper, E.: Natural language processing with Python: analyzing text with the natural language toolkit. " O'Reilly Media, Inc." (2009)
4. Chen, D., Fisch, A., Weston, J., Bordes, A.: Reading wikipedia to answer open-domain questions. In: ACL (2017)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019). https://doi.org/10.18653/v1/N19-1423, https://aclanthology.org/N19-1423
6. Gao, L., Callan, J.: 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). pp. 2843–2853. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.acl-long.203, https://aclanthology.org/2022.acl-long.203
7. Gao, L., Dai, Z., Chen, T., Fan, Z., Van Durme, B., Callan, J.: Complement lexical retrieval model with semantic residual embeddings. In: Hiemstra, D., Moens, M.F., Mothe, J., Perego, R., Potthast, M., Sebastiani, F. (eds.) Advances in Information Retrieval. pp. 146–160. Springer International Publishing, Cham (2021)
8. Garg, S., Vu, T., Moschitti, A.: Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 7780–7788 (2020)
9. Ge, S., Wu, F., Wu, C., Qi, T., Huang, Y., Xie, X.: Fedner: Privacy-preserving medical named entity recognition with federated learning. ArXiv **abs/2003.09288** (2020)
10. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.W.: Realm: Retrieval-augmented language model pre-training. In: Proceedings of the 37th International Conference on Machine Learning. ICML'20, JMLR.org (2020)
11. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.W.: Realm: Retrieval-augmented language model pre-training. ArXiv **abs/2002.08909** (2020)
12. Harabagiu, S.M., Maiorano, S.J., Pasca, M.: Open-domain textual question answering techniques. Natural Language Engineering **9**, 231 – 267 (2003)
13. Hardy, S., Henecka, W., Ivey-Law, H., Nock, R., Patrini, G., Smith, G., Thorne, B.: Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. ArXiv **abs/1711.10677** (2017)
14. Huang, J.T., Sharma, A., Sun, S., Xia, L., Zhang, D., Pronin, P., Padmanabhan, J., Ottaviano, G., Yang, L.: Embedding-based retrieval in facebook search. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. pp. 2553–2561 (2020)
15. Jiang, D., Song, Y., Tong, Y., Wu, X., Zhao, W., Xu, Q., Yang, Q.: Federated topic modeling. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management. p. 1071–1080. CIKM '19, Association for Computing Machinery,

New York, NY, USA (2019). https://doi.org/10.1145/3357384.3357909, https://doi.org/10.1145/3357384.3357909
16. Johnson, J., Douze, M., Jegou, H.: Billion-scale similarity search with gpus. IEEE Transactions on Big Data **7**(03), 535–547 (jul 2021). https://doi.org/10.1109/TBDATA.2019.2921572
17. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6769–6781. Association for Computational Linguistics, Online (Nov 2020). https://doi.org/10.18653/v1/2020.emnlp-main.550, https://aclanthology.org/2020.emnlp-main.550
18. Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Kelcey, M., Devlin, J., Lee, K., Toutanova, K.N., Jones, L., Chang, M.W., Dai, A., Uszkoreit, J., Le, Q., Petrov, S.: Natural questions: a benchmark for question answering research. Transactions of the Association of Computational Linguistics (2019)
19. Lee, J., Sung, M., Kang, J., Chen, D.: Learning dense representations of phrases at scale. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). pp. 6634–6647. Association for Computational Linguistics, Online (Aug 2021). https://doi.org/10.18653/v1/2021.acl-long.518, https://aclanthology.org/2021.acl-long.518
20. Lee, J., Yun, S., Kim, H., Ko, M., Kang, J.: Ranking paragraphs for improving answer recall in open-domain question answering. In: EMNLP (2018)
21. Lee, K., Chang, M.W., Toutanova, K.: Latent retrieval for weakly supervised open domain question answering. ArXiv **abs/1906.00300** (2019)
22. Lin, Y., Ji, H., Liu, Z., Sun, M.: Denoising distantly supervised open-domain question answering. In: ACL (2018)
23. Lu, S., He, D., Xiong, C., Ke, G., Malik, W., Dou, Z., Bennett, P., Liu, T.Y., Overwijk, A.: 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. pp. 2780–2791. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (Nov 2021). https://doi.org/10.18653/v1/2021.emnlp-main.220, https://aclanthology.org/2021.emnlp-main.220
24. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.y.: Communication-Efficient Learning of Deep Networks from Decentralized Data. In: Singh, A., Zhu, J. (eds.) Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 54, pp. 1273–1282. PMLR (20–22 Apr 2017), https://proceedings.mlr.press/v54/mcmahan17a.html
25. Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., Deng, L.: Ms marco: A human generated machine reading comprehension dataset (November 2016)
26. Paca, M.: Open-domain question answering from large text collections. Computational Linguistics **29**, 665–667 (2003)
27. Qu, Y., Ding, Y., Liu, J., Liu, K., Ren, R., Zhao, W.X., Dong, D., Wu, H., Wang, H.: 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. pp. 5835–5847. Association for Computational Linguistics, Online (Jun 2021). https://doi.org/10.18653/v1/2021.naacl-main.466, https://aclanthology.org/2021.naacl-main.466
28. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: SQuAD: 100,000+ questions for machine comprehension of text. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 2383–2392. Association for Computational Linguistics, Austin, Texas (Nov 2016). https://doi.org/10.18653/v1/D16-1264, https://aclanthology.org/D16-1264
29. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108 (2019)
30. Seo, M., Lee, J., Kwiatkowski, T., Parikh, A.P., Farhadi, A., Hajishirzi, H.: Real-time open-domain question answering with dense-sparse phrase index. ArXiv **abs/1906.05807** (2019)
31. Shen, G., Yang, Y., Deng, Z.H.: Inter-weighted alignment network for sentence pair modeling. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 1179–1189. Association for Computational Linguistics, Copenhagen, Denmark (Sep 2017). https://doi.org/10.18653/v1/D17-1122, https://aclanthology.org/D17-1122
32. Tran, Q.H., Lai, T., Haffari, G., Zukerman, I., Bui, T., Bui, H.: 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). pp. 1274–1283. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018). https://doi.org/10.18653/v1/N18-1115, https://aclanthology.org/N18-1115
33. Trischler, A., Wang, T., Yuan, X., Harris, J., Sordoni, A., Bachman, P., Suleman, K.: NewsQA: A machine comprehension dataset. In: Proceedings of the 2nd Workshop on Representation Learning for NLP. pp. 191–200. Association for Computational Linguistics, Vancouver, Canada (Aug 2017). https://doi.org/10.18653/v1/W17-2623, https://aclanthology.org/W17-2623
34. Wang, M., Smith, N.A., Mitamura, T.: 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). pp. 22–32. Association for Computational Linguistics, Prague, Czech Republic (Jun 2007), https://aclanthology.org/D07-1003
35. Xiong, L., Xiong, C., Li, Y., Tang, K.F., Liu, J., Bennett, P., Ahmed, J., Overwijk, A.: Approximate nearest neighbor negative contrastive learning for dense text retrieval. arXiv preprint arXiv:2007.00808 (2020)
36. Yoon, S., Dernoncourt, F., Kim, D.S., Bui, T., Jung, K.: A compare-aggregate model with latent clustering for answer selection. Proceedings of the 28th ACM International Conference on Information and Knowledge Management (2019)
37. Zhan, J., Mao, J., Liu, Y., Guo, J., Zhang, M., Ma, S.: Optimizing dense retrieval model training with hard negatives. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval. p. 1503–1512. SIGIR '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3404835.3462880, https://doi.org/10.1145/3404835.3462880