FedWeave: Federated Parametric RAG for Multi-Evidence Reasoning

Yuxiang Wang[†], Yawen Li[‡*], Yongxin Tong[†*], Zhilin Liang[†], Zimu Zhou[#], Zhe Xue[‡], Qiang Yang[§]
[†] Beihang University, Beijing Key Laboratory of AI-Native Data Systems, Beijing, China
[‡] Beijing University of Posts and Telecommunications, Beijing, China
[#] City University of Hong Kong, Hong Kong, China
[§] The Hong Kong Polytechnic University, Hong Kong, China
{yuxiangwang, yxtong, zlliang}@buaa.edu.cn, warmly0716@126.com,
xuezhe@bupt.edu.cn, zimuzhou@cityu.edu.hk, profqiang.yang@polyu.edu.hk

***Abstract*—Knowledge-intensive applications increasingly require retrieval-augmented generation (RAG) over documents distributed across organizations, while privacy, ownership, and regulatory constraints prevent raw documents from being centralized. This setting is especially challenging for multi-evidence reasoning, where answering a query requires complementary evidence scattered across silos. Federated in-context RAG coordinates retrieval across silos but still exposes retrieved content to the coordinator during generation. Federated parametric RAG preserves document locality by transferring lightweight document adapters instead of text, but existing methods retrieve and merge these adapters as independent knowledge units, leaving complementary evidence insufficiently coordinated. We propose FedWeave, a federated parametric RAG framework that coordinates retrieval and generation through evidence coverage. FedWeave uses compact query-facet support signals to select complementary document sets without exposing raw content, and reuses the same signals to guide interactions among selected adapters before generation. Experiments on open-domain QA benchmarks show that FedWeave improves average F1 score by 22.0% over the strongest baseline and achieves up to 54.3% gain on multi-evidence reasoning while keeping raw documents local.**

I. Introduction

Retrieval-Augmented Generation (RAG) [1]–[4] prevails in improving the factual reliability of Large Language Models (LLMs) by grounding generation in external knowledge. In many real-world domains, however, the knowledge needed by RAG is distributed across multiple data owners [5], [6]. Healthcare records may be maintained by different hospitals, financial evidence may reside in separate institutions, and legal documents may be controlled by different organizations. Due to privacy, regulatory, and ownership constraints, these raw documents cannot be simply centralized for retrieval and generation. This motivates federated retrieval-augmented generation (FedRAG) [7]–[9], where a central coordinator deploys the LLM and coordinates inference across multiple data silos, while each silo keeps its raw documents local.

An important but underexplored setting is *multi-evidence reasoning* in FedRAG (see Fig. 1). Unlike single-evidence queries [10], where one retrieved document may contain sufficient information for answering, multi-evidence queries [11]–[13] require identifying and integrating *complementary* evidence from multiple documents. These documents may be distributed across different silos and each supports only part of the query. For example, medical diagnosis may require combining pathology reports, imaging records, and genomic evidence stored at different hospitals, while financial risk assessment may require jointly analyzing transactions, compliance reports, and credit histories held by different institutions. In such cases, the system must go beyond retrieving individually relevant documents. It must select a document set that jointly covers the evidence need of the query and integrate these evidence during generation, all while preserving *document locality*.

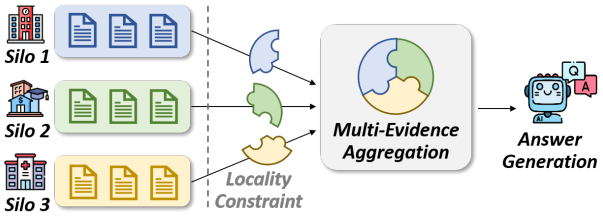


Fig. 1: Federated RAG for multi-evidence reasoning. A coordinator answers a query by collaborating with multiple silos whose raw documents remain local. The challenge is to retrieve and integrate complementary evidence distributed across silos without exposing document contents.

Existing FedRAG methods are insufficient for this setting. Most federated retrieval methods [14]–[16] focus on retrieving relevant documents or routing queries across silos, but still rely on in-context generation by sending retrieved passages or passage-derived contexts to the coordinator. This *weakens document locality* because generation still depends on exposing document content or its close substitutes. Federated parametric RAG [9] provides a more locality-preserving alternative by encoding local documents into lightweight adapters and transmitting selected adapters instead of raw text. However, existing federated parametric RAG methods are mainly designed to inject *independently* retrieved document knowledge. They do not explicitly address the *evidence coordination* required when the answer depends on multiple complementary documents.

* Corresponding authors.

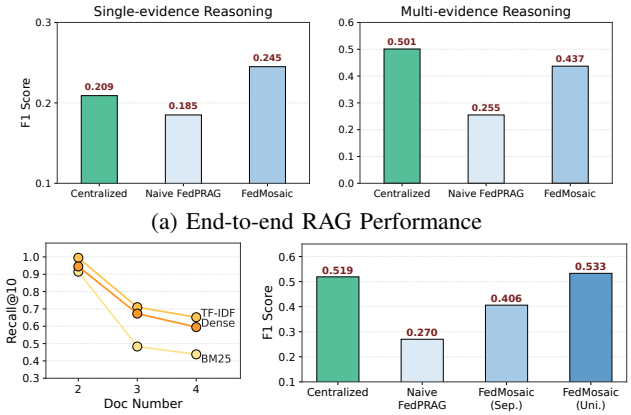
In this paper, we show that multi-evidence reasoning exposes a new coordination gap in federated parametric RAG. *(i)* On the *retrieval* side, existing methods rank candidate documents by *scalar* relevance, which may select individually relevant but redundant documents while missing complementary evidence. As the number of required supporting documents increases, relevance-driven retrieval becomes less effective at covering the full evidence need of the query. *(ii)* On the *generation* side, existing methods aggregate document adapters in parameter space before modeling their cross-document dependencies. This prematurely collapses document-specific memories and leaves evidence coordination largely to the frozen backbone LLM, preventing explicit information exchange among adapters containing complementary evidence.

To address this gap, we propose FedWeave, a federated parametric RAG framework for multi-evidence reasoning. The core idea is to introduce *evidence coverage* as a shared abstraction across retrieval and generation. In retrieval, each silo exposes only fine-grained query-facet support through coverage vectors, allowing the coordinator to select documents by *set-level complementarity* rather than independent relevance without accessing raw documents. In generation, the same coverage signal guides *cross-adapter interaction* before the selected adapters are aggregated into the backbone LLM. Thus, FedWeave goes beyond locality-preserving adapter transfer by enabling explicit coordination of complementary evidence across both retrieval and generation.

We evaluate FedWeave on open-domain QA benchmarks spanning single-evidence and multi-evidence reasoning scenarios [10]–[13]. FedWeave consistently outperforms local RAG, federated in-context RAG [8], [14], [15], federated fine-tuning [17], [18], and existing federated parametric RAG methods [9], [19]. The gains are especially notable on multi-evidence reasoning tasks, where complementary retrieval and cross-adapter interaction are critical. FedWeave outperforms the best baseline by 22.0% in average F1, with the lead reaching 54.3% on the most demanding multi-evidence queries.

Our main contributions are summarized as follows.

- We study the underexplored problem of federated parametric RAG for multi-evidence reasoning, where complementary documents are distributed across silos and raw documents must remain local.
- We propose FedWeave, a novel federated parametric RAG framework that uses evidence coverage to coordinate both retrieval and generation. It performs locality-preserving set-level retrieval to select complementary documents across silos, and enables coverage-guided cross-adapter interaction to integrate complementary evidence during generation.
- We experiment on multiple QA benchmarks and show that FedWeave improves answer accuracy over existing FedRAG solutions, while preserving document locality.



(a) End-to-end RAG Performance

(b) Retrieval side analysis

(c) Generation side analysis

Fig. 2: Challenges of federated parametric RAG for multi-evidence reasoning. (a) Centralized in-context RAG *vs.* federated parametric RAG on single-evidence and multi-evidence reasoning tasks. (b) Retrieval recall as the number of required supporting documents increases. (c) Generation with separated document adapters *vs.* generation with a unified adapter.

II. Problem and Motivation

A. Problem Statement

RAG under Locality Constraint. Retrieval-Augmented Generation (RAG) enhances Large Language Models (LLMs) by retrieving external knowledge during inference [1]–[3]. In privacy-sensitive domains such as healthcare, finance, and legal services, the knowledge needed for answering user queries is often distributed across multiple organizations and cannot be centrally shared. This motivates federated retrieval-augmented generation (FedRAG) [8], [9], [14], where a system collaboratively utilizes distributed knowledge while keeping raw documents inside their original institutions.

Multi-Evidence Reasoning. This paper focuses on *multi-evidence reasoning* in FedRAG. In this setting, answering a query requires integrating multiple complementary evidence, which may be distributed across different silos. For example, medical diagnosis may require jointly analyzing pathology reports, imaging records, and genomic evidence from different hospitals, while financial risk analysis may require reasoning over compliance reports and credit histories across institutions. Thus, the system must not only identify individually relevant documents, but also select and integrate documents providing complementary evidence under locality constraints.

Problem Formulation. Formally, we consider a federated RAG system consisting of a central coordinator C deploying an LLM and N data silos $\{S_i\}_{i=1}^{N}$. Each silo S_i maintains a local document corpus D_i, and raw documents are not allowed to leave their original silos. Given a query q, the answer may depend on a set of supporting documents $\mathcal{D}_q = \{d_1, d_2, \ldots\}$. These documents may come from different silos and provide distinct evidence needed for the final answer. The goal is to generate the answer to q by retrieving and integrating com-

plementary evidence across silos, while preserving document locality throughout both retrieval and generation.

B. Baseline Paradigm: Federated Parametric RAG

A promising paradigm for FedRAG under locality constraints is *federated parametric RAG*, which transfers knowledge through lightweight adapters rather than raw documents [9]. Under this paradigm, each silo encodes local document knowledge into document-specific adapters, and the coordinator uses the retrieved adapters for answer generation.

1) **Local Parametric Knowledge Encoding:** For each local document $d \in D_i$ stored in silo S_i, the silo trains a lightweight document adapter:

$$\mathcal{A}_d = \arg\min_{\Delta W} \mathcal{L}(d; W_{\text{base}} + \Delta W) \tag{1}$$

where W_{base} is the frozen backbone LLM parameters, ΔW is trainable adapter parameters, and $\mathcal{L}$ is the language modeling objective. The adapter $\mathcal{A}_d$ then serves as a *parametric representation* of the document knowledge.

2) **Query-Time Adapter Aggregation:** Given a query q, the coordinator selects relevant document and retrieves k adapters $\{\mathcal{A}_i\}_{i=1}^{k}$ for generation. Existing methods typically aggregate these adapters in *parameter space*:

$$x_{\text{out}} = W_{\text{base}} x_{\text{in}} + \sum_{i=1}^{k} w_i \cdot \mathcal{A}_i x_{\text{in}} \tag{2}$$

where x_{in} and x_{out} are the input and output hidden representations of a backbone LLM layer, and w_i is the aggregation weight for adapter $\mathcal{A}_i$. This design preserves document locality because raw documents are never transmitted to the coordinator. However, as shown next, this paradigm still faces important limitations when the query requires multiple *complementary* evidence sources.

C. Empirical Limitations and Design Objectives

We conduct a measurement study to examine where federated parametric RAG falls short in multi-evidence reasoning. As shown in Fig. 2(a), we compare federated parametric RAG methods, including the naive FedPRAG [19] and the state-of-the-art FedMosaic [9], against centralized in-context RAG. While federated parametric RAG remains comparable to centralized in-context RAG on single-evidence reasoning (0.245 *vs.* 0.209), its performance drops markedly on multi-evidence reasoning (0.437 *vs.* 0.501), which requires aggregating complementary evidence. This suggests that preserving document locality through adapters addresses only the privacy constraint, but not the evidence coordination required by multi-evidence reasoning. We further analyze the *retrieval* and *generation* stages and identify two key limitations.

- **Retrieval Limitation: Incomplete Query-Facet Coverage.** Multi-evidence reasoning requires the retrieved document set to cover different facets of the query. However, standard retrieval pipelines used in RAG mainly rank documents by *scalar* relevance [1], [2]. Such ranking can select individually relevant but mutually redundant documents, while missing documents providing complementary evidence. As shown in Fig. 2(b), retrieval recall on 2WikiMultihopQA [12] drops substantially as the number of required supporting documents increases, indicating that relevance-driven retrieval is insufficient for covering the full evidence need of complex queries. Although recent re-ranking methods [20], [21] improve document selection in centralized settings, they require access to full documents and are therefore incompatible with locality constraints in federated environments. This creates the need for a retrieval signal that supports set-level complementarity without exposing raw documents.
- **Generation Limitation: Insufficient Cross-Document Information Flow.** After complementary documents are selected, generation must integrate the evidence encoded in their corresponding adapters. However, existing federated parametric RAG methods [9], [19] aggregate adapters before modeling cross-document interactions, leaving evidence coordination largely to the frozen backbone LLM. As shown in Fig. 2(c), on 2WikiMultihopQA [12], separately trained document adapters perform notably worse than a unified adapter trained on all supporting documents, leading to a 32% performance gap for FedMosaic. This gap indicates that retrieved adapters do not sufficiently exchange information during generation, therefore calling for a generation mechanism that enables explicit interaction among document adapters.

Design Objectives. These limitations motivate us to design a federated parametric RAG framework for multi-evidence reasoning under locality constraints. The framework should meet two objectives: *(i)* retrieve complementary documents through locality-preserving set-level selection, and *(ii)* enable explicit information exchange among retrieved document adapters during generation.

III. FedWeave Overview

We propose FedWeave, a federated parametric RAG framework for multi-evidence reasoning under locality constraints (see Fig. 3). FedWeave keeps raw documents inside their original silos and transfers document knowledge through lightweight adapters. Following the two design objectives in Sec. II-C, FedWeave uses **evidence coverage** as the common signal for both retrieval and generation. During retrieval, evidence coverage enables the coordinator to select documents that jointly cover different query facets without accessing raw content. During generation, the same coverage signal guides cross-adapter interaction before document-specific adapter contributions are aggregated into the backbone LLM.

Architecture. FedWeave consists of two coordinated designs.

- **Coverage-Aware Federated Retrieval** (Sec. IV): Each silo performs local candidate retrieval and represents each candidate document with a *coverage vector*, which summarizes its query-facet support without exposing raw document content. The coordinator then performs set-

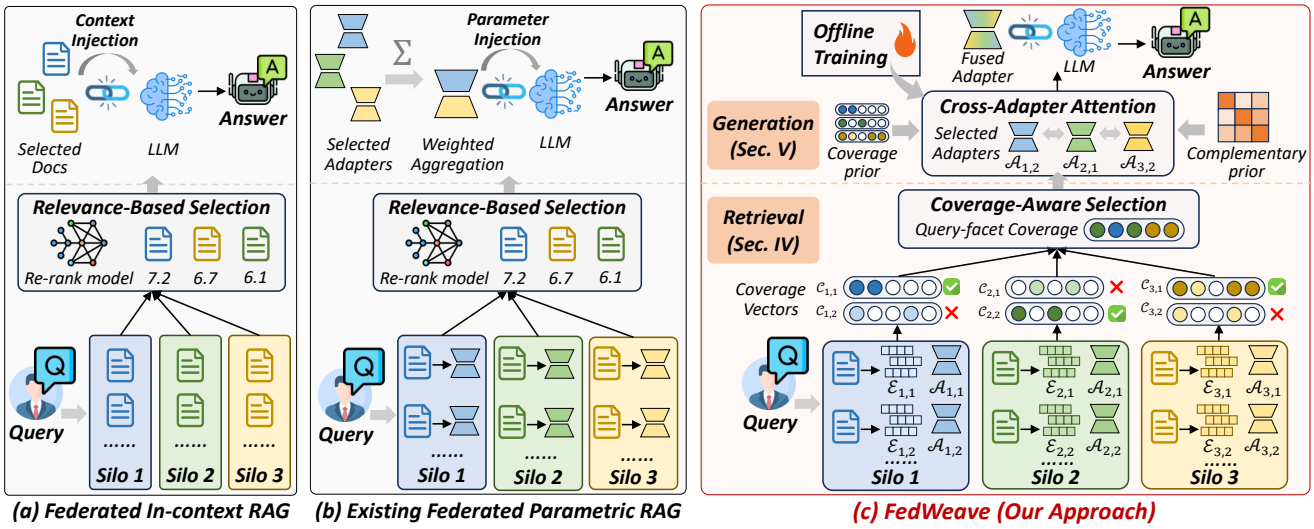


Fig. 3: Comparison of FedRAG solutions. (a) *Federated In-context RAG*: transfers retrieved documents to the coordinator, violating the locality constraint. (b) *Existing Federated Parametric RAG*: retrieves and aggregates document adapters independently, lacking coordination of complementary evidence. (c) *FedWeave (Our Approach)*: preserves document locality while enabling complementary evidence retrieval and cross-adapter collaboration for multi-evidence reasoning.

level selection over the uploaded vectors to choose documents that jointly provide complementary evidence.

- **Cross-Adapter Collaborative Generation** (Sec. V): After retrieval, the selected document adapters are uploaded to the coordinator for generation. Instead of merging adapters directly in parameter space, FedWeave keeps their document-conditioned activations separate and performs *coverage-guided cross-adapter attention* in activation space. This enables complementary adapters to exchange information before their contributions are aggregated into the backbone LLM.

Workflow. Algorithm 1 shows the workflow of FedWeave.

- In the offline stage, each silo prepares the local retrieval structures needed for computing query-conditioned coverage vectors and trains a lightweight adapter for each local document (lines 2-3). Meanwhile, the coordinator trains the cross-adapter collaborative generation module using public multi-evidence QA datasets, without accessing any silo documents (line 4).
- In the online stage, each silo first retrieves local candidate documents for the query, computes their coverage vectors, and uploads only the vectors with document identifiers to the coordinator (lines 5-8). The coordinator performs coverage-aware set-level selection based on these vectors (line 9). The silos then upload the selected adapters, and the coordinator generates the answer via the cross-adapter collaborative generation (lines 10-12).

IV. Coverage-Aware Federated Retrieval

This section presents the *retrieval* design of FedWeave. As discussed in Sec. II-C, multi-evidence reasoning requires the retrieved document set to jointly cover complementary query facets, while standard federated retrieval [1], [2] mainly ranks documents by individual relevance. To address this limitation, FedWeave introduces a locality-preserving *coverage vector* for each candidate document. The coverage vector summarizes how a document supports different parts of the query, allowing the coordinator to select documents by their marginal contribution to the whole evidence set rather than by scalar relevance. Each silo first performs local candidate retrieval and computes coverage vectors for its candidates while keeping raw documents private. The coordinator then receives only document identifiers and coverage vectors, and performs coverage-aware set selection to choose documents that jointly maximize query evidence coverage. We first define the coverage vector and the retrieval objective (Sec. IV-A), and then present the federated selection algorithm (Sec. IV-B).

A. Coverage Vector and Retrieval Objective

Coverage Vector. We construct the coverage vector using a late-interaction retriever such as ColBERT [22]. Given a query q and a document d, the retriever represents them as sets of token-level embeddings $\mathcal{E}_q = \{e_q^1, e_q^2, \ldots, e_q^{|\mathcal{E}_q|}\}$ and $\mathcal{E}_d = \{e_d^1, e_d^2, \ldots, e_d^{|\mathcal{E}_d|}\}$, where query and document embeddings lie in the same representation space.

Standard late-interaction retrieval aggregates token-level matching signals into a *scalar* relevance score. In contrast, FedWeave keeps the matching profile as a query-conditioned coverage *vector* of dimension $|\mathcal{E}_q|$. For document d and query q, the i-th dimension of its coverage vector $\mathcal{C}_d$ is defined as

$$\mathcal{C}_d[i] = \max_{e \in \mathcal{E}_d} \langle e_q^i, e \rangle \tag{3}$$

where $\langle \cdot, \cdot \rangle$ denotes inner-product similarity.

The value $\mathcal{C}_d[i]$ measures the strongest match between the query embedding e_q^i and document d. Thus, it indicates how

Algorithm 1: FedWeave

Input: user query q, coordinator C, silos $\{S_i\}_{i=1}^{N}$
Output: answer a

// Offline Stage
foreach *silo* S_i **do**
  Compute ColBERT representations for documents;
  Train LoRA adapter for each document via Equ. (1);
Coordinator C trains parameters for cross-adapter attention via Equ. (14);
// Online Stage (Retrieval)
foreach *silo* S_i **do**
  Retrieve local top-k documents $\widehat{D}_i$ based on q;
  Compute coverage vectors $\{\mathcal{C}_d \mid d \in \widehat{D}_i\}$ via Equ. (3);
  Upload coverage vectors to coordinator C;
Coordinator C selects document set $\mathcal{D}$ via Equ. (5);
// Online Stage (Generation)
Silos upload corresponding LoRA adapters $\{\mathcal{A}_d \mid d \in \mathcal{D}\}$;
Coordinator generates answer $a \leftarrow \mathrm{LLM}(q, \{\mathcal{A}_d\}, \{\mathcal{C}_d\})$;
return a;

well d supports the corresponding query facet. Compared with a scalar relevance score, the coverage vector preserves fine-grained evidence support and makes it possible to compare candidate documents by their complementarity.

Coverage-Aware Retrieval Objective. Given the coverage vectors, the coordinator selects k documents that jointly maximize query evidence coverage. For a selected document set $\mathcal{D}$, we define the utility function as:

$$\mathcal{U}(\mathcal{D}) = \sum_{i=1}^{|\mathcal{E}_q|} \max_{d \in \mathcal{D}} \mathcal{C}_d[i] \tag{4}$$

This objective keeps the strongest support for each query facet among all selected documents. A candidate that mainly supports already covered facets yields limited marginal gain. In contrast, a candidate that supports under-covered facets contributes more to the utility. Thus, maximizing Equ. (4) encourages the selected document set to contain complementary evidence across documents and silos.

B. Coverage-Aware Federated Document Selection

Given the coverage-aware objective, FedWeave adopts a two-stage selection procedure. Each silo first retrieves local candidate documents and computes their coverage vectors. The coordinator then re-ranks candidates from all silos using only the uploaded vectors.

Local Candidate Search. Given a query q, each silo independently searches its local corpus D_i for a local candidate set $\widehat{D}_i$. This step follows the standard late-interaction retrieval pipeline [22], optionally using sparse retrieval methods such as BM25 [23] for efficient candidate filtering. For each candidate document $d \in \widehat{D}_i$, the silo computes its coverage vector according to Equ. (3).

The silo uploads only document identifiers and their corresponding coverage vectors to the coordinator. Raw documents, local indexes, and document embeddings remain inside the silo. Thus, the coordinator obtains enough information for evidence selection without accessing local document content.

Global Coverage-Aware Re-ranking. After receiving candidate coverage vectors from all silos, the coordinator forms the global candidate pool $\widehat{D} = \cup_{i=1}^{N} \widehat{D}_i$ and selects k documents to maximize Equ. (4). As this objective is NP-hard (see Lemma 1 in the Appendix), we adopt a greedy approximation algorithm.

Starting from an empty set $\mathcal{D}$, the coordinator repeatedly selects the document with the largest marginal coverage gain:

$$d^* = \arg \max_{d \in \widehat{D} \setminus \mathcal{D}} \left(\mathcal{U}(\mathcal{D} \cup \{d\}) - \mathcal{U}(\mathcal{D})\right) \tag{5}$$

The selected document is added to $\mathcal{D}$, and the process continues until k documents are selected. The selected document identifiers are then sent back to the corresponding silos, which upload only the associated document adapters for generation.

Performance Guarantee. The utility function $\mathcal{U}(\cdot)$ is monotone because adding a document cannot decrease the maximum coverage of any query facet. It is also submodular because the marginal gain of a document decreases as more query facets are covered. Thus, the greedy algorithm enjoys the standard approximation guarantee for cardinality-constrained monotone submodular maximization [24].

Theorem 1: Let $\mathcal{D}^*$ denote the document set returned by the greedy algorithm and $\mathcal{D}_{OPT}$ denote the optimal solution. Then, $\mathcal{U}(\mathcal{D}^*) \geq (1 - 1/e) \cdot \mathcal{U}(\mathcal{D}_{OPT})$.

The proof is provided in the Appendix. If each silo returns k local candidates, the candidate pool contains kN documents. The greedy algorithm performs k iterations over this pool, yielding a time complexity of $O(k^2 N)$. This makes coverage-aware selection efficient while preserving document locality.

V. Cross-Adapter Collaborative Generation

As discussed in Sec. II-C, multi-evidence reasoning requires the model to integrate complementary evidence encoded in different document adapters. However, direct parameter-space aggregation collapses document-specific memories before their dependencies are explicitly modeled. To address this limitation, FedWeave keeps adapter outputs as separate document-conditioned activations and refines them through cross-adapter interaction before aggregation. To make this interaction evidence-aware, FedWeave reuses the coverage vectors from retrieval to derive complementarity priors among selected documents. These priors guide cross-adapter attention toward adapters that provide complementary query-facet support. We first present activation-space cross-adapter conditioning (Sec. V-A), then introduce coverage-guided cross-adapter attention (Sec. V-B), and finally describe the training strategy of the collaborative generation (Sec. V-C).

A. Activation-Space Cross-Adapter Conditioning

Given a query, the retrieval stage returns a document set $\mathcal{D} = \{d_1, \ldots, d_k\}$ and loads the corresponding adapters $\{\mathcal{A}_1, \ldots, \mathcal{A}_k\}$ for generation. Each adapter encodes knowledge from one retrieved document and contributes a document-conditioned update to the backbone LLM W_{base}.

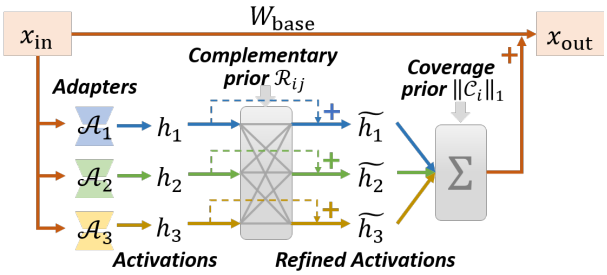


Fig. 4: Coverage-guided cross-adapter conditioning in activation space.

Conditioning in Activation Space. Instead of merging adapters directly in the parameter space [9], [19], FedWeave keeps their outputs separate in the *activation space*. Given the hidden representation x_{in}, each document adapter first produces a document-conditioned activation:

$$h_i = \mathcal{A}_i x_{\text{in}}, \quad i = 1, \ldots, k \tag{6}$$

Rather than immediately summing these activations, FedWeave treats $\{h_i\}_{i=1}^k$ as separate document-conditioned representations and models their dependencies before aggregation. This is realized by a **Cross-Adapter Attention (CAA)** module (see Sec. V-B):

$$x_{\text{out}} = W_{\text{base}} x_{\text{in}} + \text{CAA}(\{h_i\}_{i=1}^k) \tag{7}$$

Compared with parameter-space aggregation in Equ. (2), Equ. (7) delays aggregation until document-conditioned activations have been refined through interactions with other retrieved adapters. This allows FedWeave to model dependencies among complementary document memories while preserving independent training and storage for each document adapter.

B. Coverage-Guided Cross-Adapter Attention

We now instantiate the CAA module introduced in Sec. V-A. The goal is to refine each document-conditioned activation using information from other retrieved adapters before final aggregation. To make this conditioning evidence-aware, FedWeave reuses the coverage vectors computed during retrieval and derives a complementarity prior over retrieved documents. This prior guides attention toward adapters whose source documents cover different query facets (see Fig. 4).

Coverage-Based Complementarity Prior. Given the coverage vectors $\{\mathcal{C}_i\}_{i=1}^k$ of the retrieved documents, where $\mathcal{C}_i$ is the coverage vector of document d_i, we construct a coverage-based complementarity matrix:

$$\mathcal{R}_{ij} = 1 - \frac{\langle \mathcal{C}_i, \mathcal{C}_j \rangle}{\|\mathcal{C}_i\| \cdot \|\mathcal{C}_j\|} \tag{8}$$

where $\mathcal{R}_{ij}$ measures how different documents d_i and d_j are in their coverage of query facets. A larger $\mathcal{R}_{ij}$ indicates that the two documents provide more complementary evidence. We use this matrix as a structural prior for cross-adapter attention, so that each document-conditioned activation can attend more strongly to complementary evidence sources.

Algorithm 2: Cross-Adapter Collaborative Generation

Input: input hidden representation x_{in}, LLM backbone W_{base}, adapters $\{\mathcal{A}_i\}_{i=1}^k$, coverage vectors $\{\mathcal{C}_i\}_{i=1}^k$
Output: output hidden representation x_{out}

```
Compute the coverage-based complementarity matrix R from {C_i} via Equ. (8);
Generate adapter-specific activations {h_i}_{i=1}^k via Equ. (6);
Compute cross-adapter attention weights α_ij with complementarity prior R_ij via Equ. (10);
Aggregate activations with weight α_ij and combine them with original activation to obtain refined activation h̃_i;
Compute aggregation weights β_i with coverage prior ||C_i||_1 via Equ. (12);
Aggregate refined activations with weight β_i and combine them with layer output to obtain x_out;
return x_out;
```

Cross-Adapter Attention. Given document-conditioned activations $\{h_i\}_{i=1}^k$, where $h_i \in \mathbb{R}^{d_h}$, CAA first computes query, key, and value projections:

$$Q_i = W_Q h_i, \quad K_i = W_K h_i, \quad V_i = W_V h_i \tag{9}$$

where $W_Q, W_K, W_V \in \mathbb{R}^{d_a \times d_h}$ are learnable projection matrices and d_a is the attention hidden dimension.

The attention weight from adapter i to adapter j combines activation-space similarity with the coverage-based prior:

$$\alpha_{ij} = \text{softmax}_j \left(\frac{\langle Q_i, K_j \rangle}{\sqrt{d_a}} + \log \mathcal{R}_{ij} \right) \tag{10}$$

The first term captures learned relatedness between adapter activations, while $\log \mathcal{R}_{ij}$ biases attention toward adapters whose source documents cover complementary query facets.

Using these attention weights, CAA refines each document-conditioned activation as:

$$\widetilde{h}_i = h_i + \gamma \cdot W_O \sum_{j=1}^{k} \alpha_{ij} V_j \tag{11}$$

where $W_O \in \mathbb{R}^{d_h \times d_a}$ projects the attended representation back to the adapter activation space, and γ controls the strength of cross-adapter conditioning. The residual form preserves the original document-conditioned activation while allowing it to incorporate complementary evidence from other adapters.

Coverage-Aware Adapter Aggregation. After cross-adapter conditioning, FedWeave aggregates the refined activations into the backbone hidden state. For each adapter, the aggregation weight is computed as:

$$\beta_i = \text{softmax}_i \left(\frac{\langle W_T x_{\text{in}}, W_D h_i \rangle}{\sqrt{d_a}} + \log \|\mathcal{C}_i\|_1 \right) \tag{12}$$

where $W_T, W_D \in \mathbb{R}^{d_a \times d_h}$ project the token representation and the refined activation into a shared attention space. The similarity term $\frac{\langle W_T x_{\text{in}}, W_D h_i \rangle}{\sqrt{d_a}}$ measures how relevant adapter i is to the current generation state. The coverage prior $\|\mathcal{C}_i\|_1$

favors documents with broader support over the query facets.

Finally, the aggregated collaborative activation is added to the backbone output:

$$x_{\text{out}} = W_{\text{base}} x_{\text{in}} + \sum_{i=1}^{k} \beta_i \widetilde{h}_i \tag{13}$$

This gives the concrete formulation of Equ. (7). By conditioning adapter activations before aggregation, FedWeave enables explicit information flow among complementary document memories, rather than independent parameter-space merging. The procedure of the complete cross-adapter collaborative generation is summarized in Algorithm 2.

C. Training of Collaborative Generation Module

The generation module is trained at the coordinator using public multi-evidence QA datasets such as HotpotQA [11] and 2WikiMultihopQA [12]. Its purpose is to learn how to condition and aggregate document-specific adapter activations for queries that require evidence from multiple sources.

During training, the backbone W_{base} and all document adapters $\{\mathcal{A}_i\}$ are frozen. Only the parameters of the collaborative generation module $\{W_Q, W_K, W_V, W_O, W_T, W_D\}$ are optimized. For each training instance, the coordinator retrieves a document set $\mathcal{D}$, and applies cross-adapter attention. Given a query-answer pair $\langle q, a \rangle$, the model is optimized with the standard autoregressive language modeling objective:

$$\mathcal{L} = -\sum_{t} \log P(a_t \mid a_{<t}, q, \mathcal{D}) \tag{14}$$

After training, the collaborative generation module is deployed at the coordinator for inference. Since the backbone and document adapters remain frozen, FedWeave improves multi-evidence reasoning without changing the locality-preserving adapter training and storage pipeline.

VI. Experiments

A. Experiment Setup

Baselines. We compare FedWeave against ten baselines from four categories: **(1) Local RAG:** performs retrieval and generation locally within each silo and synthesizes silo-level answers, including Standard in-context RAG [2], CoTRAG [25] and ReAct [26]. **(2) Federated In-context RAG:** retrieves documents across silos and injects the retrieved text into the coordinator-side prompt, including C-FedRAG [8], MKPQA [14] and RAGRoute [15]. **(3) Federated Fine-tuning:** trains the backbone via federated LoRA optimization and performs inference with the tuned model, including FedIT [17] and FLoRA [18]. **(4) Federated Parametric RAG:** Encode documents into LoRA adapters and inject them into the backbone at generation time, including naive FedPRAG [19] and state-of-the-art FedMosaic [9]. Our implementation of FedWeave are available at https://github.com/lewellin727/FedWeave.

Datasets. We test on four QA benchmarks: HotpotQA (HQA) [11], 2WikiMultihopQA (2WQA) [12], and ComplexWebQuestions (CWQ) [13] for multi-evidence reasoning, and PopQA (PQA) [10] for single-evidence reasoning. Besides, we use Enron Emails and WikiText for privacy evaluation [27].

Models and Configurations. We use LLaMA3.2-1B-Instruct as the backbone LLM. For each dataset, the document corpus is partitioned into $N = 6$ silos under a Dirichlet allocation with $\alpha = 0.1$ to simulate Non-IID cross-silo distributions, and the default retrieval size is set to $k = 5$.

Metrics. We assess the generation quality using the F1 score between the answer and the ground truth, which serves as the primary end-to-end evaluation metric. We also evaluate retrieval accuracy using the top-k recall of golden supporting documents, and privacy protection using the protection success rate against the targeted attack [28] and the prefix attack [29].

B. Main Results

Gains on Multi-Evidence Reasoning. As shown in Table I, FedWeave achieves the best F1 on all datasets, outperforming the strongest baseline FedMosaic by 22.0% on average. The gains are particularly pronounced on multi-evidence reasoning tasks, reaching 54.3%, 37.4%, 40.0% and 8.4% on four subtypes of 2WQA. These results demonstrate the effectiveness of coverage-aware retrieval and cross-adapter generation for cross-silo evidence coordination.

Generalization to Unseen Datasets. Although the cross-adapter generation module of FedWeave is trained only on HQA and 2WQA, its benefits transfer to unseen datasets. On CWQ, FedWeave achieves 0.4223 F1 score, outperforming FedMosaic by 9.9%. It also ranks first on the single-evidence PQA dataset with 0.2800 F1, indicating strong generalization across both datasets and task types.

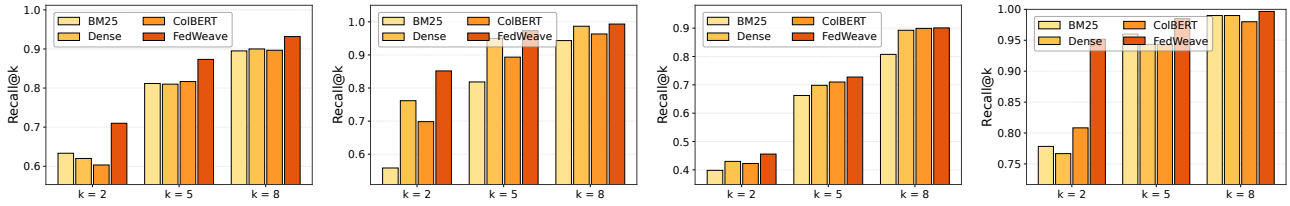
C. Micro-Benchmarks

Retrieval Accuracy. We compare FedWeave's coverage-aware retriever against three relevance-based baselines (BM25 [23], Dense [2], and ColBERT [22]). Fig. 5 reports the top-k recall across HQA and 2WQA. On HQA bridge, FedWeave achieves Recall@2 of 71.0%, compared to 60.3%–63.3% for the baselines, and maintains a clear advantage at larger retrieval size. The gain comes from coverage-aware selection, which favors documents that contribute previously uncovered query facets rather than redundant evidence. The coverage signal is also used by the cross-adapter attention at generation time, and Fig. 6 shows that on representative HQA and 2WQA queries, the pairwise complementarity scores within golden passages are clearly larger than within distractor pairs, providing helpful structural signal for CAA during generation.

Privacy Evaluation. We evaluate the privacy protection of FedWeave by comparing it with both in-context and parametric baselines, *i.e.*, C-FedRAG [8] and FedMosaic [9], using the average protection success rate under the targeted attack [28] and the prefix attack [29]. As shown in Table III, both parametric methods are immune to the targeted attack, since no PII context is present in the prompt. Under prefix attack,

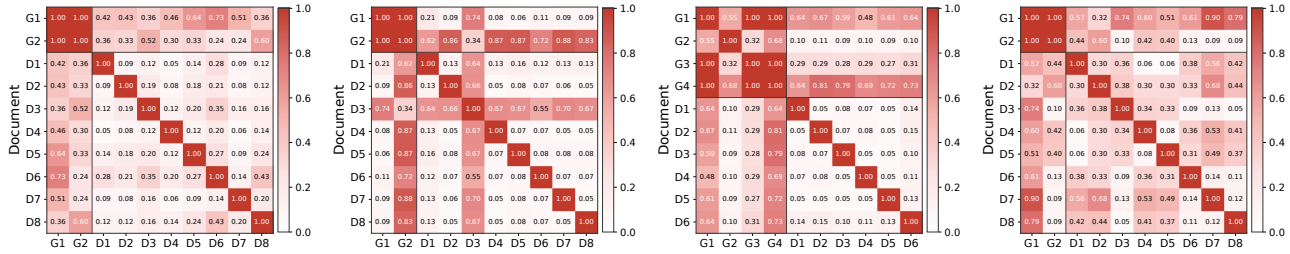
TABLE I: Overall accuracy for FedWeave and baseline methods.

Type	Method	HQA		2WQA				CWQ	PQA	AVG
		Bridge	Compar.	Bridge	Compar.	Compos.	Infer.	Total	Total	
Local RAG	Standard RAG	0.2415	0.3842	0.4631	0.4387	0.0599	0.1190	0.2804	0.1366	0.2654
	CoTRAG [25]	0.1087	0.3451	0.3619	0.4198	0.0366	0.1013	0.2378	0.0444	0.2069
	ReAct [26]	0.2192	0.4120	0.4361	0.5730	0.0431	0.0986	0.2443	0.1135	0.2675
Federated In-context RAG	C-FedRAG [8]	0.0331	0.1366	0.2201	0.2364	0.0014	0.0584	0.1946	0.1207	0.1252
	MKPQA [14]	0.2464	0.3767	0.4394	0.4325	0.0096	0.1062	0.2008	0.1329	0.2431
	RAGRoute [15]	0.2562	0.3677	0.3286	0.4045	0.0144	0.0655	0.1767	0.0380	0.2064
Federated Fine-tuning	FedIT [17]	0.1547	0.4649	0.5447	0.3478	0.0603	0.1323	0.3687	0.2152	0.2861
	FLoRA [18]	0.1605	0.4496	0.2507	0.3415	0.0802	0.1227	0.3497	0.1917	0.2433
Federated Parametric RAG	Naive FedPRAG [19]	0.0732	0.3358	0.5383	0.4804	0.0258	0.1306	0.2810	0.0759	0.2426
	FedMosaic [9]	0.2606	0.5276	0.3912	0.4710	0.0672	0.1758	0.3841	0.2368	0.3143
	FedWeave (Ours)	**0.2636**	**0.5679**	**0.6036**	**0.6471**	**0.0941**	**0.1905**	**0.4223**	**0.2800**	**0.3836**



(a) HQA Bridge (b) HQA Comparison (c) 2WQA Bridge (d) 2WQA Comparison

Fig. 5: Top-k recall of FedWeave and three relevance-based baselines.



(a) HQA Bridge (b) HQA Comparison (c) 2WQA Bridge (d) 2WQA Comparison

Fig. 6: Coverage-based complementarity prior $\mathcal{R}$ on multi-evidence queries (G* indicates golden document and D* indicates distractor document).

FedWeave reduces leakage by 26.2% to 94.4% relative to FedMosaic. This improvement stems from cross-adapter attention, which encourages reasoning across multiple documents rather than reliance on a single document, thereby reducing verbatim replay of sensitive content without privacy-specific training.

Varying Retrieval Size k. We vary the retrieval size k from 2 to 10 to assess its impact on generation quality. As shown in Table II, the F1 score is largely insensitive to k, with the maximum deviation from the mean being only 7.2%, indicating that FedWeave can utilize a varying number of documents without suffering from information overload.

D. Ablation Studies

Impact of Cross-Adapter Attention. We ablate the cross-adapter attention module by replacing it with a uniform mean-pool over the retrieved adapter outputs. As shown in Fig. 7, this degrades the average F1 by 28.5% across the datasets, confirming that CAA is the primary driver of FedWeave's gains. The drop is largest on bridge-style question subtypes, where evidence fusion is the most critical.

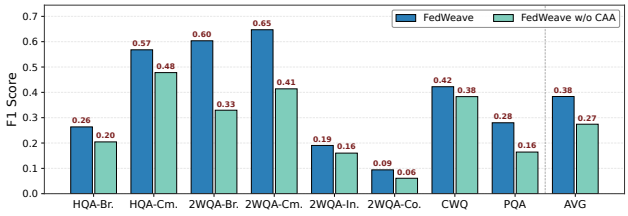


Fig. 7: Ablation of the cross-adapter attention module.

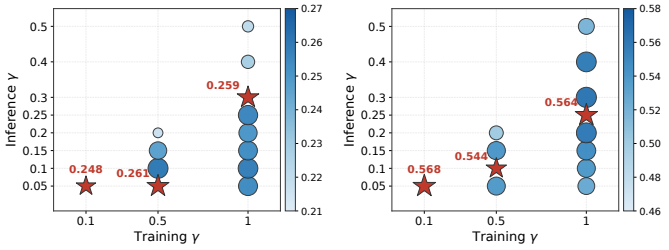
Impact of Conditioning Strength γ. To assess the sensitivity of the conditioning strength γ, we vary its values during

TABLE II: F1 score under varying retrieval size k on multi-evidence datasets.

Dataset	Type	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=8$	$k=10$
HQA	Bridge	$0.2556_{-1.4\%}$	$0.2565_{-1.0\%}$	$0.2530_{-2.4\%}$	$0.2612_{+0.8\%}$	$0.2602_{+0.4\%}$	$0.2632_{+1.5\%}$	$\mathbf{0.2647}_{+2.1\%}$
	Comparison	$0.5285_{-2.8\%}$	$0.5369_{-1.3\%}$	$0.5428_{-0.2\%}$	$\mathbf{0.5679}_{+4.4\%}$	$0.5346_{-1.7\%}$	$0.5483_{+0.8\%}$	$0.5477_{+0.7\%}$
2WQA	Bridge	$0.6238_{-0.6\%}$	$\mathbf{0.6361}_{+1.3\%}$	$0.6327_{+0.8\%}$	$0.6036_{-3.9\%}$	$0.6327_{+0.8\%}$	$0.6327_{+0.8\%}$	$0.6327_{+0.8\%}$
	Comparison	$0.6305_{-1.6\%}$	$0.6381_{-0.4\%}$	$0.6419_{+0.1\%}$	$0.6351_{-0.9\%}$	$0.6463_{+0.8\%}$	$0.6463_{+0.8\%}$	$\mathbf{0.6486}_{+1.2\%}$
	Inference	$0.1826_{-1.7\%}$	$0.1785_{-4.0\%}$	$0.1840_{-1.0\%}$	$0.1905_{+2.5\%}$	$0.1850_{-0.5\%}$	$\mathbf{0.1931}_{+3.9\%}$	$0.1872_{+0.7\%}$
	Composition	$0.0929_{+5.8\%}$	$0.0886_{+0.9\%}$	$0.0847_{-3.5\%}$	$\mathbf{0.0941}_{+7.2\%}$	$0.0847_{-3.5\%}$	$0.0847_{-3.5\%}$	$0.0847_{-3.5\%}$
CWQ	Total	$0.4010_{-3.4\%}$	$0.3987_{-4.0\%}$	$0.4190_{+0.9\%}$	$0.4223_{+1.7\%}$	$0.4204_{+1.3\%}$	$0.4218_{+1.6\%}$	$\mathbf{0.4231}_{+1.9\%}$

TABLE III: Privacy evaluation under target and prefix attacks.

Attack	Method	Target Prompt	Target Info.	Repeat Prompt	Repeat Length
Target	C-FedRAG [8]	138.93	102.20	131.33	21.34
	FedMosaic [9]	**0.00**	**0.00**	**0.00**	**0.00**
	FedWeave (Ours)	**0.00**	**0.00**	**0.00**	**0.00**
Prefix	C-FedRAG [8]	52.86	46.66	55.93	23.58
	FedMosaic [9]	44.00	29.53	22.86	22.22
	FedWeave (Ours)	**12.53**	**11.33**	**1.27**	**16.41**



(a) HQA Bridge (b) HQA Comparison

Fig. 8: Sensitivity of F1 score to the conditioning strength γ.

training and inference on HQA dataset. As shown in Fig. 8, peak F1 stays within a 5.1% relative range across training values, indicating that FedWeave's gains do not depend on fine-grained tuning.

VII. Related Work

Retrieval-Augmented Generation. RAG improves the factual reliability of LLMs by incorporating external knowledge during generation [1]–[4]. Existing RAG methods can generally be divided into *in-context* RAG and *parametric* RAG. In-context RAG directly injects retrieved documents into prompts, while parametric RAG encodes retrieved knowledge into trainable parameters such as lightweight adapters [19], [30]. Parametric RAG is especially attractive for federated settings since knowledge can be transferred through compact parameters instead of raw text. However, most parametric RAG methods are developed for centralized environments.

Multi-Evidence Reasoning in RAG. Multi-evidence reasoning has been widely studied through multi-hop or multi-document QA benchmarks, where answering a query requires integrating complementary information from multiple sources [11], [12], [31], [32]. Existing studies improve multi-evidence RAG through better retrieval, re-ranking, evidence decomposition, and cross-document reasoning. These methods typically assume that the retriever or generator can access raw document contents, which enables centralized evidence comparison and context construction. In contrast, we study multi-evidence reasoning when supporting documents are distributed across silos and raw documents cannot be exposed to the coordinator.

Federated RAG. FedRAG extends RAG to settings where knowledge is distributed across multiple silos [7]. Existing FedRAG studies mainly focus on federated retrieval, silo routing, or privacy-preserving search. For example, FANNS [16], [33] studies dense retrieval under heterogeneous retrieval models across silos, MKPQA [14] enables probabilistic multi-domain retrieval, RAGRoute [15] improves retrieval efficiency through query routing, and C-FedRAG [8] further considers confidential retrieval. However, these methods require transmitting raw documents, thereby violating locality constraints.

Federated Parametric RAG. Federated parametric RAG provides a locality-preserving alternative by transferring document knowledge through lightweight adapters instead of transmitting raw documents [9]. This paradigm is built upon parametric RAG [19], but differs in that documents remain inside their original silos and the coordinator performs generation using retrieved document adapters. Existing methods typically aggregate selected adapters in parameter space, which is effective for injecting independent document knowledge but less suitable for multi-evidence reasoning.

VIII. Conclusion

We present FedWeave, a federated parametric RAG framework for multi-evidence reasoning under document locality constraints. FedWeave uses evidence coverage as a shared abstraction across retrieval and generation. During retrieval, coverage vectors support locality-preserving selection of complementary document sets. During generation, the same coverage signal guides cross-adapter attention, allowing document-specific adapters to exchange complementary evidence before aggregation. Experiments on four open-domain QA benchmarks show that FedWeave consistently outperforms local RAG, in-context FedRAG, federated fine-tuning, and federated parametric RAG baselines, with strong gains on multi-evidence reasoning tasks.

ACKNOWLEDGEMENT

This work was partially supported by National Natural Science Foundation of China (NSFC) (Grant Nos. 62425202, 62532002, 62572061, 62336003), the Beijing Natural Science Foundation (Z230001), the Fundamental Research Funds for the Central Universities No. JKF-2026007844235, and the SKLCCSE. Zimu Zhou's work is supported in part by a Theme-based Research Scheme (TRS) Project from the Research Grant Council in Hong Kong, under project T43-101/26-N.

REFERENCES

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *NeurIPS*, 2020.
[2] V. Karpukhin, B. Oguz, S. Min, P. Lewis *et al.*, "Dense passage retrieval for open-domain question answering," in *EMNLP*, 2020.
[3] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua, and Q. Li, "A survey on rag meeting llms: Towards retrieval-augmented large language models," in *SIGKDD*, 2024.
[4] W. Jiang, S. Zhang, B. Han *et al.*, "Piperag: Fast retrieval-augmented generation via adaptive pipeline parallelism," in *SIGKDD*, 2025.
[5] Z. Zhang, X. Hu, J. Zhang, Y. Zhang *et al.*, "Fedlegal: The first real-world federated learning benchmark for legal nlp," in *ACL*, 2023.
[6] B. Pfitzner, N. Steckhan, and B. Arnrich, "Federated learning in a medical context: a systematic literature review," *ACM TOIT*, 2021.
[7] A. Chakraborty, C. Dahal, and V. Gupta, "Federated retrieval-augmented generation: A systematic mapping study," in *EMNLP*, 2025.
[8] P. Addison, M.-T. H. Nguyen, T. Medan, J. Shah, M. T. Manzari *et al.*, "C-fedrag: A confidential federated retrieval-augmented generation system," *arXiv preprint arXiv:2412.13163*, 2024.
[9] Z. Liang, Y. Wang, Z. Zhou, H. Zhang, B. Liu, and Y. Tong, "Fedmosaic: Federated retrieval-augmented generation via parametric adapters," in *SIGIR*, 2026.
[10] A. Mallen, A. Asai, V. Zhong, R. Das, D. Khashabi, and H. Hajishirzi, "When not to trust language models: Investigating effectiveness of parametric and non-parametric memories," in *ACL*, 2023.
[11] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, "Hotpotqa: A dataset for diverse, explainable multi-hop question answering," in *EMNLP*, 2018.
[12] X. Ho, D. Nguyen *et al.*, "Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps," in *COLING*, 2020.
[13] A. Talmor and J. Berant, "The web as a knowledge-base for answering complex questions," in *NAACL*, 2018.
[14] P. Shojaee, S. S. Harsha, D. Luo *et al.*, "Federated retrieval augmented generation for multi-product question answering," in *COLING*, 2025.
[15] R. Guerraoui, A.-M. Kermarrec, D. Petrescu, R. Pires, M. Randl, and M. de Vos, "Efficient federated search for retrieval-augmented generation," in *EuroMLSys*, 2025.
[16] Y. Wang, Y. Tong, Z. Zhou, Z. He, R. Hu, and K. Xu, "Federated retrieval over embedding-heterogeneous vector databases," in *ICDE*, 2026.
[17] J. Zhang, S. Vahidian, M. Kuo, C. Li *et al.*, "Towards building the federatedgpt: Federated instruction tuning," in *ICASSP*, 2024.
[18] Z. Wang, Z. Shen, Y. He, G. Sun, H. Wang, L. Lyu, and A. Li, "Flora: Federated fine-tuning large language models with heterogeneous low-rank adaptations," *NeurIPS*, 2024.
[19] W. Su, Y. Tang, Q. Ai, J. Yan, C. Wang, H. Wang, Z. Ye, Y. Zhou, and Y. Liu, "Parametric retrieval augmented generation," in *SIGIR*, 2025.
[20] M. Glass, G. Rossiello, M. F. M. Chowdhury, A. Naik, P. Cai, and A. Gliozzo, "Re2g: Retrieve, rerank, generate," in *NAACL*, 2022.
[21] J. Sun, X. Zhong, S. Zhou, and J. Han, "Dynamicrag: Leveraging outputs of large language model as feedback for dynamic reranking in retrieval-augmented generation," in *NeurIPS*, 2026.
[22] O. Khattab and M. Zaharia, "Colbert: Efficient and effective passage search via contextualized late interaction over bert," in *SIGIR*, 2020.
[23] S. Robertson and H. Zaragoza, *The probabilistic relevance framework: BM25 and beyond*. Foundations and Trends® in Information Retrieval, 2009.
[24] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical programming*, 1978.
[25] J. Wei, X. Wang, D. Schuurmans *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *NeurIPS*, 2022.
[26] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran *et al.*, "React: Synergizing reasoning and acting in language models," in *ICLR*, 2023.
[27] B. Klimt and Y. Yang, "The enron corpus: A new dataset for email classification research," in *ECML*, 2004.
[28] S. Zeng, J. Zhang, P. He, Y. Liu, Y. Xing, H. Xu, J. Ren, Y. Chang, S. Wang, D. Yin *et al.*, "The good and the bad: Exploring privacy issues in retrieval-augmented generation (rag)," in *ACL*, 2024.
[29] N. Carlini, D. Ippolito, M. Jagielski, K. Lee *et al.*, "Quantifying memorization across neural language models," in *ICLR*, 2022.
[30] X. Liu, S. Qiao, R. Nyange *et al.*, "Mergeprag: Orthogonal merging of passage-experts for multi-hop parametric rag," in *ICLR*, 2026.
[31] C. Zhao, C. Xiong, C. Rosset, X. Song *et al.*, "Transformer-xh: Multi-evidence reasoning with extra hop attention," in *ICLR*, 2020.
[32] V. Mavi, A. Jangra, and A. Jatowt, "Multi-hop question answering," *Foundations and Trends ® in Information Retrieval*, 2024.
[33] Y. Wang, R. Hu, Z. He, Z. Liang, X. Zhao, and Y. Tong, "Fedbridge: A federated query engine over embedding-heterogeneous vector databases," in *ICDE*, 2026.
[34] N. Megiddo, E. Zemel, and S. L. Hakimi, "The maximum coverage location problem," *Journal on Algebraic Discrete Methods*, 1983.

APPENDIX

Lemma 1: Given a candidate document collection $\widehat{D}$ and retrieval budget k, the problem of selecting a document subset $\mathcal{D} \subseteq \widehat{D}$ with $|\mathcal{D}| = k$ that maximizes $\mathcal{U}(\mathcal{D})$ is NP-hard.

Proof: We prove the lemma by reduction from the Maximum Coverage problem [34], which, given an element set $V = \{v_1, \ldots, v_n\}$ and a subset collection $\{B_1, \ldots, B_m\}$, aims to select at most k subsets such that the number of covered elements is maximized.

Given a Maximum Coverage instance, we construct one document d_j for each subset B_j. For each element $v_i \in V$, we define the i's dimension of coverage vector of d_j as:

$$\mathcal{C}_{d_j}[i] = \begin{cases} 1, & v_i \in B_j, \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Under this construction, the utility $\mathcal{U}(\mathcal{D})$ exactly equals the number of elements covered by the corresponding subset selection. Thus, maximizing $\mathcal{U}(\mathcal{D})$ is equivalent to solving the Maximum Coverage problem. Since Maximum Coverage is NP-hard [34], the lemma follows. ∎

Theorem 1: Let $\mathcal{D}^*$ denote the document set returned by the greedy algorithm and $\mathcal{D}_{OPT}$ denote the optimal solution. Then, $\mathcal{U}(\mathcal{D}^*) \geq (1 - 1/e) \cdot \mathcal{U}(\mathcal{D}_{OPT})$.

Proof: We prove the theorem by showing that $\mathcal{U}(\cdot)$ is monotone and submodular.

- **Monotonicity.** For any $\mathcal{D}_1 \subseteq \mathcal{D}2$, we have $\max_{d \in \mathcal{D}_1} \mathcal{C}d[i] \leq \max_{d \in \mathcal{D}_2} \mathcal{C}_d[i]$ for each dimension i. Summing over all dimensions gives $\mathcal{U}(\mathcal{D}_1) \leq \mathcal{U}(\mathcal{D}_2)$, proving monotonicity.
- **Submodularity.** Define the marginal utility of adding d to $\mathcal{D}$ as $\delta(d \mid \mathcal{D}) := \mathcal{U}(\mathcal{D} \cup d) - \mathcal{U}(\mathcal{D})$. For any $\mathcal{D}_1 \subseteq \mathcal{D}_2$ and $d' \notin \mathcal{D}_2$, the coverage of each dimension under $\mathcal{D}_2$ is no smaller than that under $\mathcal{D}_1$. Hence, adding d' yields no greater gain on $\mathcal{D}_2$, i.e., $\delta(d' \mid \mathcal{D}_1) \geq \delta(d' \mid \mathcal{D}_2)$. Thus, $\mathcal{U}(\cdot)$ is submodular.

Since $\mathcal{U}(\cdot)$ is monotone submodular under a cardinality constraint, the greedy algorithm achieves a $(1 - 1/e)$ approximation ratio according to [24]. ∎