DFAMS: Dynamic-flow guided Federated Alignment based Multi-prototype Search

Zhibang Yang[1,2*], Xinke Jiang[1,2,3*], Rihong Qiu[1,2,3*], Ruiqing Li[1,2], Yihang Zhang[1], Yue Fang[1,2], Yongxin Xu[1,3], Hongxin Ding[1,2], Xu Chu[2,3,4†], Junfeng Zhao[2,3†], Yasha Wang[1,5†]

[1]National Engineering Research Center for Software Engineering, Peking University, Beijing, China
[2]School of Computer Science, Peking University, Beijing, China
[3]Key Laboratory of High Confidence Software Technologies, Ministry of Education, Beijing, China
[4]Center on Frontiers of Computing Studies, Peking University, Beijing, China
[5]Peking University Information Technology Institute (Tianjin Binhai), Tianjin, China

{yangzb, xinkejiang, rihongqiu}@stu.pku.edu.cn, {chu_xu, zhaojf, wangyasha}@pku.edu.cn

Abstract

Federated Retrieval (FR) routes queries across multiple external knowledge sources, to mitigate hallucinations of LLMs, when necessary external knowledge is distributed. However, existing methods struggle to retrieve high-quality and relevant documents for ambiguous queries, especially in cross-domain scenarios, which significantly limits their effectiveness in supporting downstream generation tasks. Inspired by Dynamic Information Flow (DIF), we propose DFAMS, a novel framework that leverages DIF to identify latent query intents and construct semantically aligned knowledge partitions for accurate retrieval across heterogeneous sources. Specifically, DFAMS probes the DIF in LLMs by leveraging gradient signals from a few annotated queries and employing Shapley value-based attribution to trace neuron activation paths associated with intent recognition and subdomain boundary detection. Then, DFAMS leverages DIF to train an alignment module via multi-prototype contrastive learning, enabling fine-grained intra-source modeling and inter-source semantic alignment across knowledge bases. Experimental results across five benchmarks show that DFAMS outperforms advanced FR methods by up to 14.37% in knowledge classification accuracy, 5.38% in retrieval recall, and 6.45% in downstream QA accuracy, demonstrating its effectiveness in complex FR scenarios. Our code is publicly available at https://github.com/Artessay/DFAMS.

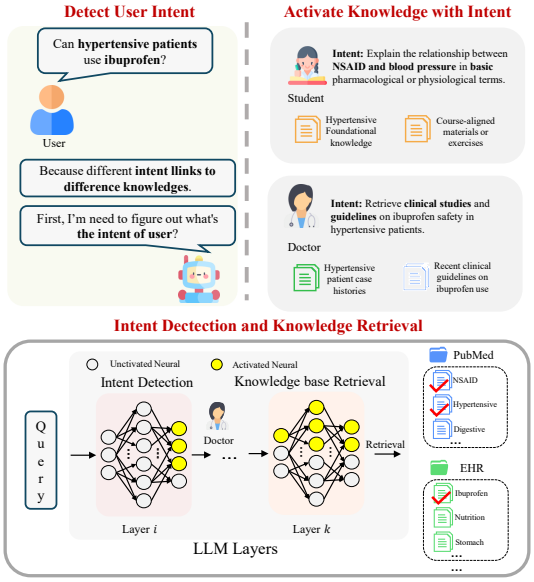

Figure 1: Hypothesized process of dynamic information flow (DIF) within LLMs for knowledge base selection. When a user asks "Can hypertensive patients use ibuprofen?", the LLM first infers the latent intent—where a student seeks basic pharmacological understanding, while a doctor requires clinical evidence. The identified intent (e.g., as a doctor) triggers distinct neural and knowledge activations, forming DIF signals that guide retrieval: clinical pathways access ibuprofen records in EHR, whereas conceptual pathways retrieve NSAID-related information from PubMed.

1 Introduction

Retrieval-Augmented Generation (RAG) leverages external knowledge documents (Edge et al., 2024; Asai et al., 2023) to effectively enhance the factuality and verifiability of outputs from Large Language Models (LLMs) (OpenAI, 2022, 2023; Vu et al., 2024; Liao et al., 2024; Yang et al., 2026b; Wu et al., 2025b), significantly mitigating issues such as hallucination and knowledge obsolescence (Ji et al., 2023; Cao et al., 2020; Jiang et al., 2024b,a; Asai et al., 2024; Su et al., 2024; Jeong et al., 2024; Liao et al., 2025; Baek et al., 2025). However, mainstream RAG approaches typically rely on a single, centralized vector database for knowledge storage and retrieval (Kukreja et al., 2024; Bhavnani and Wilson, 2009). In reality, knowledge is inherently distributed across multiple heterogeneous data sources — for instance, in medical scenarios, retrieval may need to simultane-

*Equal contribution.
†Corresponding authors.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 24728–24752

ously access electronic health records (EHR) (Yuan et al., 2023), textbooks, and the latest research papers (Zhao et al., 2025; Liao et al., 2025). Forcibly aggregating all documents into a unified index not only incurs high retrieval costs but also raises concerns around data sovereignty (Jiang, 2024; Shokouhi et al., 2011; Kairouz et al., 2021; Yang et al., 2026a). To address this, **Federated Retrieval (FR)** has emerged as a solution, enabling efficient cross-knowledge-base routing decisions that directly and precisely direct queries to the most relevant sources of knowledge (Schölkopf, 2019; Guerraoui et al., 2025; Wang et al., 2024c; Ryan et al., 2025; Shojaee et al., 2025).

Most existing FR methods are primarily designed with an emphasis on routing efficiency, privacy preservation, and downstream task integration (Chakraborty et al., 2025). However, in real-world scenarios with complex semantics (Clarke et al., 2008), users are more concerned with whether the system can accurately retrieve highly relevant documents to effectively support downstream generation tasks (Shokouhi et al., 2011; Huang and Huang, 2024), which does not receive sufficient attention in existing research. And due to limitations in current modeling strategies (Wang et al., 2024c), existing approaches exhibit notable shortcomings in addressing this issue(Guerraoui et al., 2025). On one hand, user queries often suffer from semantic ambiguity or compression, leading to a gap between surface expressions and underlying intent (Yuan et al., 2025). In different contexts, the same question may require different knowledge sources to answer (as illustrate in Figure 1). In such scenarios, user queries often fail to align with the structured and detailed content in the knowledge base, limiting the accuracy and coverage of semantic matching in traditional FR methods (Huang et al., 2021). Although some approaches leverage LLMs for prompt-based query rewriting (Gao et al., 2022) to reduce ambiguity, they often struggle to capture fine-grained semantics due to limited prompt expressiveness, leading to suboptimal performance in semantically complex scenarios. On the other hand, in real-world applications knowledge bases are typically partitioned by data sources, forming multiple structured yet interrelated knowledge subsystems (Wu et al., 2025a). The semantic boundaries between these subsystems are often flooded and overlapping. However, some existing methods overlooking underlying semantic connections, which struggle to support cross-source recall and dynamic integration (Wang et al., 2024c).

Recent research has shown that when LLMs process tasks of different fields, the contribution of each parameter in the LLM model varies (Dhamdhere et al., 2018; Yu et al., 2018; LeCun et al., 1989; Zhang et al., 2025). Building on this line of work, further investigations into the structural and functional mechanisms of LLMs have revealed that, during inference, LLMs naturally form a Dynamic Information Flow (DIF)—a latent path through which information propagates dynamically across transformer layers, activating neural substructures associated with semantics, knowledge, and reasoning (Zheng et al.; Yu and Ananiadou, 2024; Wang et al., 2024d). These findings suggest that LLMs may already possess an implicit capability to recognize user intent and organize knowledge into latent subdomains when processing complex queries. Inspired by this, we raise a central research question: Can DIF within LLMs be explicitly modeled to (1) more accurately identify users' latent query intents, and (2) structurally segment and dynamically organize overlapping or fuzzy-boundary knowledge subdomains to mitigate semantic misalignment and cross-source retrieval failures in FR settings? As shown in Figure 1, we hypothesize that, when presented with a complex query, the LLM first identifies the user's intent. If the model determines that the user is likely a doctor with an intent to retrieve clinical studies and guidelines, it will activate the corresponding neurons, forming a reasoning pathway that generates perception signals related to the target knowledge. These signals may be used to retrieve and align content from heterogeneous sources such as PubMed and EHR, integrating cross-subdomain knowledge for downstream generation.

To validate the above hypothesis, we need to address two core challenges: (C1) how to accurately detect the relevant DIF within LLMs; (C2) how to leverage the DIF-based internal semantic organization to support fine-grained knowledge base modeling while preserving semantic associations across multiple sources. To tackle these challenges, we propose a novel framework named DFAMS (**D**ynamic-flow guided **F**ederated **A**lignment based **M**ulti-prototype **S**earch). DFAMS explicitly models the internal DIF of LLMs and constructs knowledge base partitions aligned with the model's activation patterns, thereby preserving rich semantic signals. For **Challenge C1**, we utilize gradient signals under a small number of annotated

DIF-probing samples, and apply Shapley value-based attribution methods to identify neuron flow paths associated with query intent recognition and subdomain boundary detection. For **Challenge C2**, during training, we extract DIF flows induced by queries over each knowledge base. These flows are then used to train an alignment module via multi-prototype contrastive learning, achieving fine-grained intra-source modeling and inter-source alignment. The goal is to maintain semantic continuity across sources while enabling effective knowledge base classification. In summary, our contributions are as follows:

- We reveal a high-dimensional, information-rich DIF in LLMs that encodes both user intent and subdomain knowledge, enabling more faithful query understanding. To our knowledge, this is the first work to exploit DIF for intent-aware, domain-sensitive retrieval modeling.
- We propose the DFAMS framework. By modeling DIF, we construct knowledge partitions that preserve inter-source semantic associations. DFAMS integrates multi-prototype contrastive learning during training and employs Adaptive Prototype-Guided Routing at inference time, significantly improving the performance.
- We develop an enhanced FR benchmark to encompass realistic and diverse query types, ranging from knowledge-free queries, multi-fragment queries within a single source, to cross-source retrieval. The benchmark integrates structured taxonomy, associated documents, user queries, and ground-truth answers, offering a solid foundation for evaluating FR in complex settings.

2 Related Work

2.1 Federated Retrieval

Federated search (Shokouhi et al., 2011), extended into RAG by combining privacy-preserving federated learning (FL) (Zhang et al., 2021) with RAG (Lewis et al., 2020), enables retrieval across decentralized sources without sharing raw data (Chakraborty et al., 2025), which has seen widespread adoption in privacy-critical domains such as healthcare (Jiang, 2024; Jung et al., 2025; Xiong et al., 2024), finance, and legal services (Addison et al., 2024). Prior work in federated search mainly targets three aspects: (i) privacy and security through secure retrieval and encryption (Jeon et al., 2021; Peng et al., 2021), (ii) retrieval efficiency via query routing (Wang et al., 2024c), and (iii) integration of FL and RAG tailored to domain-specific tasks (Wang et al., 2024a; Zeng et al., 2024; Shojaee et al., 2025). These advances have proven effective (Zhao, 2024; Xu et al., 2022; Wang et al., 2024a; Zeng et al., 2024; Shojaee et al., 2025); however, retrieving high-quality results in complex semantic scenarios remains challenging (Wang et al., 2024c). Existing FR methods, prompt-based and embedding-based, struggle in this setting because queries often misalign with knowledge structures, which reduces dense vector accuracy (Huang et al., 2021), and LLM-based prompt rewriting lacks fine-grained semantic precision (Gao et al., 2022).

2.2 Neural Information Flow

Recent studies have shown that LLMs, like the human cortex (Arbib, 2003; Hawrylycz et al., 2012; Zador, 2019; Wang et al., 2024b), exhibit functional partitioning across their architecture (Dhamdhere et al., 2018; Yu et al., 2018; LeCun et al., 1989; Chu et al., 2025). Such functional partitions may emerge in the form of attention heads (Zheng et al., 2024; Yin and Steinhardt, 2025; Wu et al., 2024), feed-forward networks (Bandarkar et al., 2024; Wendler et al., 2024; Sun et al., 2025), or neurons (Huo et al., 2024; Tang et al., 2024), which are shaped during training and contribute differently across tasks (Dhamdhere et al., 2018; Yu et al., 2018). Building on this partitions, information dynamically flows among these functional modules, forming a Dynamic Information Flow (DIF) (Stolfo et al.; Yu and Ananiadou, 2024). To leverage DIF for downstream tasks, researchers attempt to detect DIF by quantifying the attribution of parameters in the LLM with respect to input query. Quantitative attribution methods have been extensively explored using various techniques, including forward-based methods (Liang et al.; Todd et al., 2023; Jiang et al., 2025; Dai et al., 2021) and backward-based methods (Feng et al., 2025, 2024) or their combinations (Xu et al., 2024). Among these methods, Shapley value-based approaches (Ghorbani and Zou, 2020; Adamczewski et al., 2024) have gained wide adoption. Although there has been extensive research on DIFs in large models, leveraging their powerful capabilities for modeling user intent and knowledge bases remains largely unexplored.

3 Method

We introduce DFAMS, a framework designed to enable LLM to perform semantically grounded FR

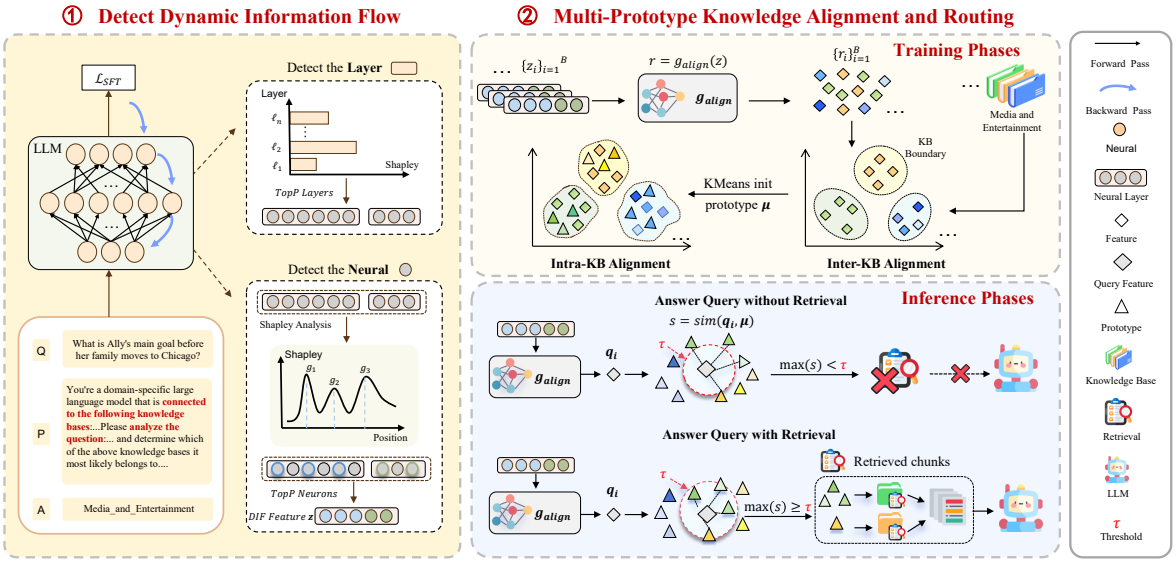

Figure 2: DFAMS dynamically detects relevant information flow in LLMs and employs multi-prototype alignment and routing to accurately associate queries with domain-specific knowledge bases.

across distributed knowledge bases, as illustrated in Figure 2. DFAMS operates in three stages: (1) formalizing the FR setting (Section 3.1); (2) extracting query-specific internal representations through Dynamic Information Flow (DIF) modeling (Section 3.2); and (3) aligning these representations with structured knowledge prototypes for adaptive, multi-source routing (Section 3.3). All key notations are summarized in Appendix A.

3.1 Problem Definition

Federated Retrieval Formalization. We formulate FR as a distributed retrieval problem over I isolated data sources, where each source $i \in \{1, \ldots, I\}$ privately hosts a knowledge base $\mathcal{K}_i = \{d_{i\ell}\}_{\ell=1}^{M_i}$ with M_i documents. Due to strict privacy constraints, sources cannot exchange raw documents or intermediate representations. Given a user query x, the system must determine a routing vector:

$$f_{\text{route}} : x \mapsto \mathbf{w} = [w_1, w_2, \ldots, w_I], \quad w_j \in \mathbb{N}_0,$$

where w_j specifies how many documents to retrieve from $\mathcal{K}_j$. The challenge lies in selecting the most relevant sources adaptively while avoiding unnecessary retrieval.

RAG in FR. Following RAGRoute (Guerraoui et al., 2025), we combine two types of knowledge: (*i*) parameterized knowledge Θ stored in the model weights, and (*ii*) non-parameterized knowledge $\mathcal{D} = \{\mathcal{K}_1, \ldots, \mathcal{K}_I\}$, representing distributed, domain-specific corpora. Given a query x, the goal is to generate a reliable response: $\texttt{Response} \leftarrow \Theta(x, R \mid \mathcal{P})$, where $\mathcal{P}$ is a task-specific prompt, and R is the subset of knowledge bases deemed relevant. To handle queries answerable without external retrieval, we include an `Others` category solved solely by Θ (Su et al., 2024). During training, supervision is single-source: each $(x, \mathcal{K}_i, y)$ is paired with a single knowledge base or labeled as `Others`. At inference, the model generalizes to *no-source*, *single-source*, or *multi-source* retrieval by predicting per-source relevance scores and selecting those above a dynamic threshold δ.

3.2 Dynamic Information Flow Modeling

To accurately interpret user intent and uncover domain-relevant semantics for routing (**C1**), we detect DIF capturing which neurons contribute most to domain-sensitive behavior, producing a compact, semantically grounded representation.

The process consists of two steps: (1) constructing a controlled probing dataset to isolate domain-selection behaviors; (2) identifying key layers and neurons via gradient-based Shapley attribution, and aggregating their activations into a DIF embedding.

Probing Dataset Construction. Using benchmark training or test sets for attribution often yields spurious signals, as they contain mixed user specific intents. To address this, we construct a dedicated probing dataset $\mathcal{D}_{\text{probe}} = \{(x_i, \mathcal{K}_i)\}_{i=1}^{n_{\text{probe}}}$, where each query x_i is synthesized with fixed

instruction-style prompts explicitly designed to elicit domain-selection (e.g., asking the model to identify the most relevant knowledge base). Each label $\mathcal{K}_i \in \{1, \ldots, I\}$ specifies the correct knowledge base. This dataset is disjoint from $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ to ensure no leakage.

Neuron Attribution and DIF Embedding. Using $\mathcal{D}_{\text{probe}}$, we estimate the importance of each neuron using Shapley-based attribution. For a transformer block at layer t, the feed-forward network (FFN) computes each neuron activation as: $\theta_{t,j} = \textsc{Act}([h_t W_{t1} + b_{t1}]_j)$, where $h_t \in \mathbb{R}^d$ is the output of the attention sublayer, $W_{t1} \in \mathbb{R}^{d \times 4d}$ and $b_{t1} \in \mathbb{R}^{4d}$ are projection weights and biases, and $\textsc{Act}(\cdot)$ denotes the nonlinearity. For each parameter $\theta_j \in \Theta$, its Shapley value ϕ_j is:

$$\phi_j = -g_j^{(\gamma)}\theta_j - \frac{1}{2}\omega_{jj}^{(j)}\theta_j^2 H_{jj}^{(\gamma)} - \frac{1}{2}\theta_j \sum_{k \neq j} \omega_{jk}^{(\mathcal{S})} H_{jk}^{(\gamma)} \theta_k, \tag{1}$$

where $g_j^{(\gamma)} = \partial \mathcal{L}_{\text{SFT}} / \partial \theta_j$ is the supervised loss gradient, and $H_{jk}^{(\gamma)}$ the Hessian for second-order interactions. The coefficients $\omega_{jj}^{(j)}$ and $\omega_{jk}^{(\mathcal{S})}$ weight self and pairwise contributions, respectively.

We then: ❶. Select the top layers $\mathcal{L}_{\text{top}}$ based on aggregated Shapley scores. ❷. Within each selected layer $\ell \in \mathcal{L}_{\text{top}}$, identify the most informative neuron groups $\mathcal{G}_\ell$ (adjacent units with the highest ϕ_j). ❸. For a query x, collect the activations of these groups and concatenate them to form the DIF representation: $\mathbf{z}(x) = \textsc{Concat}\Big(\big\{h_\ell^{(g)}(x) \;\big|\; g \in \mathcal{G}_\ell\big\}_{\ell \in \mathcal{L}_{\text{top}}}\Big)$ captures both semantic intent and domain cues, and serves as the input for both inter- and intra-knowledge-base modeling (Section 3.3).

Computational Complexity. Although Shapley-value estimation is typically expensive, we use two approximations: (i) gradient-based Model Shapley (Chu et al., 2025) instead of combinatorial Shapley, and (ii) blockwise rather than full Hessian approximation. The time complexity becomes $O(I \cdot B \cdot M + M \cdot d)$, where I is the number of iterations, B the batch size, M the number of parameters, and d the block size. Memory complexity is reduced from $O(M^2)$ to $O(M + d^2)$. For a 7B-parameter model, this lowers Hessian memory from $\sim$196 TB to $\sim$28 GB, making DIF extraction feasible on standard multi-GPU hardware. Detailed runtime measurements are in Appendix L.

3.3 Multi-Prototype Knowledge Alignment & Routing

To bridge internal DIF representations and external distributed knowledge structure (**C2**), we map $\mathbf{z}$ to a semantic space using a projection g_{align}, producing $\mathbf{r} = g_{\text{align}}(\mathbf{z})$. We train it with two contrastive stages and use it for prototype-guided routing.

Inter-KB Alignment. We apply supervised contrastive learning to model the boundary between different knowledge bases:

$$\ell_{i,p}^{\text{CL}} = -\log \frac{\exp(\mathbf{r}_i^\top \mathbf{r}_p / \tau_{cl})}{\sum_{a \in A(i)} \exp(\mathbf{r}_i^\top \mathbf{r}_a / \tau_{cl})} \tag{2}$$

The overall objective is defined as $\mathcal{L}_{\text{CL}} = \sum_i \frac{1}{|P(i)|} \sum_{p \in P(i)} \ell_{i,p}^{\text{CL}}$, where, $P(i)$ denotes all in-batch positive samples that share the same knowledge base as i, excluding i itself, and $A(i)$ includes all other in-batch samples except i. $\tau_{cl} \in \mathbb{R}^+$ is a temperature parameter.

Intra-KB Alignment. Besides inter-KB modeling, we perform intra-KB modeling to capture fine-grained variations. Specifically, we cluster embeddings within each class into prototypes $\{\boldsymbol{\mu}_m\}_{m=1}^M$ using KMeans, which are used to initialize and optimize the prototype contrastive loss.

$$\ell_{i,m}^{\text{PCL}} = -\log \frac{\exp(\mathbf{r}_i^\top \boldsymbol{\mu}_m / \tau_{pcl})}{\sum_{j \in A(i)} \exp(\mathbf{r}_i^\top \boldsymbol{\mu}_j / \tau_{pcl})} \tag{3}$$

The full objective is computed by averaging over all queries and their positive prototype sets: $\mathcal{L}_{\text{PCL}} = \sum_i \frac{1}{|C(i)|} \sum_{m \in C(i)} \ell_{i,m}^{\text{PCL}}$, where $C(i)$ denotes the set of nearest prototypes to $\mathbf{r}_i$ based on cosine similarity, and $A(i)$ includes all prototypes excluding those in $C(i)$. During inference, given a query embedding $\mathbf{q}$, we compute its similarity scores $s_i = \text{sim}(\mathbf{q}, \boldsymbol{\mu}_i)$ to all prototypes $\boldsymbol{\mu}_i$. The routing function then proceeds in two stages:

Adaptive Triggering. If the maximum similarity falls below a threshold τ, i.e., $\max_i s_i < \tau$, the system abstains from retrieval.

$$f_{\text{route}}(\mathbf{q}) = \begin{cases} \mathbf{0}, & \max_i s_i < \tau, \\ [w_1, \ldots, w_K], & \text{otherwise} \end{cases} \tag{4}$$

Semantic Routing. Otherwise, it identifies the top-N most similar prototypes $\mathcal{I}$ and allocates a total of T retrieval slots across knowledge bases. The

number of documents assigned to each knowledge base k is computed as:

$$w_k = \left\lfloor \frac{\sum_{i \in \mathcal{I},\, k_i = k} s_i}{\sum_{k'} \sum_{i \in \mathcal{I},\, k_i = k'} s_i} \cdot T \right\rfloor \tag{5}$$

This two-step mechanism ensures retrieval occurs only when necessary, with resources allocated by prototype-level semantic relevance.

4 Experiments

We conduct extensive experiments across multiple datasets to evaluate the effectiveness of DFAMS in Federated Retrieval settings and answer the following key research questions:

- **RQ1 (Section 4.2):** Does DFAMS consistently outperform existing advanced methods?
- **RQ2 (Section 4.3):** How do the proposed component contribute to performance improvements?
- **RQ3 (Section 4.4):** How sensitive is DFAMS to variations in model configurations?

4.1 Experimental Setup

LLM Backbones. We implement DFAMS on four open-weight LLMs with varying scales to evaluate scalability and generalization: *Qwen2.5-0.5B*, *Qwen2.5-3B*, *Qwen2.5-7B* (Yang et al., 2024), and *Llama3.1-8B* (Grattafiori et al., 2024).

Retrieval Configuration. Following Wu et al. (2025a) and Guerraoui et al. (2025), DFAMS utilizes FAISS (Douze et al., 2024) for dense vector retrieval across three FR scenarios, each comprising multiple knowledge bases. For each query, top-10 chunks are retrieved from selected sources across all relevant knowledge bases within the same scenario. Further details are provided in Appendix C

Datasets. We construct three in-domain evaluation benchmarks, *Wiki*, *Med*, and *PEP*. To further evaluate out-of-domain generalization, we test models trained on *Wiki* and *Med* using a subset of *MMLU* (Hendrycks et al., 2020) and a subset of *MIRAGE* (Xiong et al., 2024), following the setup of (Guerraoui et al., 2025). Further construction details are provided in Appendix D.

Baselines. We compare DFAMS with several representative retrieval and routing strategies: *No-RAG*, *Merged-RAG*, and two prompt-based methods *Prompt* and *CoT Prompt* (Wei et al., 2023). We further compare with a supervised fine-tuning baseline (*SFT*) (Hu et al., 2021). In addition, we consider two recent multi-source retrieval approaches: *RAGRoute* (Guerraoui et al., 2025), which employs a binary classifier for each knowledge base to select the Top-k sources, and *RopMura* (Wu et al., 2025a), a prototype-based multi-agent routing method. Implementation details are provided in Appendix E.

Evaluation Metrics. We report three complementary metrics: (1) **Cls Acc**, which measures whether the predicted knowledge base(s) match the ground truth, including correct `Others` predictions for no-retrieval cases; (2) **Recall**, computed over Top-10 retrieved documents for retrieval-triggering queries, measuring the proportion of gold documents successfully retrieved excluding no-retrieval cases; (3) **QA**, the end-to-end answer accuracy, using accuracy(QA ACC) for multiple-choice and LLM-based scoring (QA Score) (Zheng et al., 2023) for open-ended questions. More metric details are available in Appendix F.

4.2 Main Result Analysis (RQ1)

To address *RQ1*, we train DFAMS on the *Wiki*, *Med*, and *PEP* datasets, and evaluate it on both in-domain (Wiki, Med, PEP) and out-of-domain (MMLU, MIRAGE) benchmarks using *Qwen2.5-7B* and *LLaMA3.1-8B*, comparing against baselines from retrieval, prompt-based, and multi-source routing methods.

Comparison with Baselines. Table 1 summarizes the performance of DFAMS compared with a range of baseline methods across three key metrics: Cls Acc, Recall, and QA Acc/Score. DFAMS consistently outperforms all baselines on both in-domain and out-of-domain datasets.

Comparison of Advanced Multi-Source Retrieval and Naive Methods. In most cases, multi-source retrieval methods (*RopMura*, *RAGRoute*) achieve higher Cls Acc and Recall than *Prompt*-based or *Merged-RAG* baselines in most cases. For example, on the Wiki dataset with *Qwen2.5-7B*, RAGRoute achieves a Cls Acc of 76.07% and recall of 50.04%, compared to just 32.47% and 25.93% from Prompt. Similarly, RopMura outperforms Merged-RAG on Med in Recall (41.59% vs. 33.89%). While multi-source retrieval methods (*RopMura*, *RAGRoute*) generally achieve higher recall through broader source coverage, these gains often come at the cost of increased cross-domain noise. For instance, although RAGRoute obtains higher recall on the Med dataset, its QA accuracy (73.64%) falls short of that of Merged-RAG

Method	In-Domain								Out-of-Domain	
	Wiki			Med			PEP		MMLU	MIRAGE
	Cls Acc (↑)	**Recall (↑)**	**QA Acc (↑)**	**Cls Acc (↑)**	**Recall (↑)**	**QA Acc (↑)**	**Cls Acc (↑)**	**QA Score (↑)**	**QA Acc (↑)**	**QA Acc (↑)**
Qwen2.5-7B										
No RAG	/	/	59.30	/	/	75.20	/	7.56	79.05	68.05
Merged-RAG	/	48.49	77.20	/	33.89	79.60	/	8.33	77.74	65.78
Prompt	32.47	25.93	63.66	48.82	34.68	79.59	66.86	6.21	80.77	69.17
CoT Prompt	58.67	38.17	71.40	48.82	31.40	80.72	77.91	5.92	80.52	69.47
SFT	41.13	25.60	61.97	49.35	24.70	75.73	68.76	7.23	78.23	67.97
RopMura	62.83	48.92	77.47	53.09	41.59	82.01	75.92	7.18	80.38	69.97
RAGRoute	76.07	50.04	78.40	69.04	35.78	73.64	51.47	7.14	80.20	69.77
DFAMS	**85.03**	**53.83**	**78.94**	**71.81**	**42.82**	**82.82**	**82.85**	**8.39**	**86.17**	**79.88**
LLaMA3.1-8B										
No RAG	/	/	60.40	/	/	68.27	/	5.35	59.31	61.90
Merged-RAG	/	48.49	76.48	/	33.89	74.17	/	5.38	56.30	62.68
Prompt	36.18	25.93	64.67	49.57	34.72	71.65	65.99	5.17	63.90	61.51
CoT Prompt	58.20	38.14	68.98	48.92	31.59	74.46	76.75	4.82	63.55	63.43
SFT	42.61	26.12	63.78	49.31	26.41	70.41	63.95	4.86	58.31	62.11
RopMura	62.83	48.92	76.78	53.09	41.59	75.37	75.92	4.46	64.06	61.82
RAGRoute	76.07	50.04	76.83	69.04	35.78	74.55	51.47	4.58	64.37	63.16
DFAMS	**84.36**	**53.83**	**78.60**	**72.57**	**42.82**	**77.65**	**81.98**	**8.08**	**64.89**	**64.89**

Table 1: Performance comparison (%) on *Wiki*, *Med*, *PEP*, *MMLU*, and *MIRAGE*, where **bold** indicates the best result, and symbol slash "/" denotes inapplicable metrics .

(79.60%). In contrast, prompt-based methods adopt a more conservative source selection strategy, often retrieving from fewer knowledge bases, which helps reduce cross-domain noise. Merged methods, on the other hand, rely on dense semantic similarity across the entire corpus; while the retrieved chunks may not always be precisely grounded, they tend to be semantically coherent.

Comparison of DFAMS and Advanced Multi-Source Retrieval. Compared with advanced multi-source retrieval methods, DFAMS consistently achieves higher Cls Acc and Recall across all datasets and backbones. For instance, on Wiki with *Qwen2.5-7B*, DFAMS outperforms RAGRoute by +8.96% in Cls Acc (85.03% vs. 76.07%) and +3.79% in Recall (53.83% vs. 50.04%). These improvements stem from DFAMS's fine-grained modeling of DIF, enabling more accurate identification of query intent and relevant knowledge sources. This leads to better routing and more focused retrieval with less cross-domain noise. As a result, DFAMS consistently achieves the highest QA accuracy across all datasets. For example, on the *Qwen2.5-7B* backbone, it reaches 82.82% QA accuracy on Med and 78.94% on Wiki, outperforming both RopMura and RAGRoute.

Adaptive Retrieval Capability. DFAMS learns to decide whether external retrieval is needed for a given query. We evaluate its adaptive retrieval capability by grouping all sources into a single `Knowledge` class and using an `Others` class for queries answerable via parametric knowledge. As shown in Table 2, DFAMS achieves high accuracy on Wiki (99.95%) and Med (93.67%), closely matching *Probing RAG* and significantly outperforming the *Prompt*-based approach. These results highlight DFAMS's ability to avoid unnecessary retrieval while preserving high coverage when external information is required.

Method	**Wiki** Acc (↑)	**Med** Acc (↑)
Prompt	69.73	84.20
Probing RAG	**99.98**	92.80
DFAMS	99.95	**93.67**

Table 2: Adaptive retrieval accuracy comparison between different methods on Wiki and Med datasets.

Inference Efficiency. We assess DFAMS's efficiency by measuring average routing, retrieval, and total latency per sample on the Med dataset (Table 3). Despite relying on LLMs, DFAMS achieves a low end-to-end latency of 1.48s—substantially faster than *Prompt* (15.59s) and *Merged-RAG* (3.89s). Compared to *RAGRoute* (1.64s), DFAMS is slightly faster due to retrieving from fewer sources. In RAGRoute, its pursuit of higher recall often triggers more knowledge bases, and slower sources can increase latency despite parallel execution. Overall, DFAMS offers faster inference with precise routing.

Method	Routing (s)	Retrieval (s)	Total (s)
Prompt	14.25	1.35	15.59
Merged-RAG	**0**	3.89	3.89
RAGRoute	0.0023	1.64	1.64
DFAMS	0.13	**1.34**	**1.48**

Table 3: Comparison of routing, retrieval, and total processing times for different methods on the Med dataset.

4.3 Ablation Study (RQ2)

To answer **RQ2**, we conduct ablation studies on the two core components of DFAMS: (1) *Dynamic Information Flow Modeling* and (2) *Multi-Prototype Knowledge Alignment and Routing*, aiming to investigate their individual contributions to the overall system performance.

Dynamic Information Flow Modeling. We conduct ablation studies to test our central hypothesis: that DIF signals not only exist but can be effectively detected and exploited to improve model performance. We design two experimental settings: (1) Frozen LLM w/o Align-MLP: We remove the trainable `Align-MLP` and directly utilize the extracted DIF. This setting investigates whether native LLM activations inherently encode subdomain-aware signals—i.e., whether meaningful associations between user queries and knowledge subdomains can be inferred without explicit alignment. (2) Full DFAMS (w/ Align-MLP): We enable the trainable `Align-MLP` to utilize DIF to assess whether modeling the knowledge base on DIF leads to improved alignment and enhanced downstream performance.

Method	Wiki (↑)	Med (↑)
Frozen (no `Align-MLP`*)*		
Random	53.19	42.90
Full	64.05	52.67
DFAMS	**67.74**	**54.68**
Trained (with `Align-MLP`*)*		
Random	81.49	66.47
Full	82.42	68.36
DFAMS	**85.03**	**71.81**

Table 4: Ablation analysis of Dynamic Information Flow modeling on Wiki and Med datasets

We conduct experiments on both the *Wiki* and *Med* datasets. Table 4 shows that in setting (1), DFAMS (2000-dim) outperforms both random (2000-dim) and full-layer (3584-dim) baselines, achieving +14.6% accuracy gain on Wiki and +11.78% on Med over the random baseline. These results validate our hypothesis: DIF captures query-subdomain associations and can be directly leveraged, even without further training. Moreover, DFAMS also surpasses the Full baseline, yielding +3.69% and +2.01% accuracy improvements on the Wiki and Med datasets, respectively. In setting (2), with `Align-MLP` enabled, DFAMS achieves +3.54% and +2.61% higher accuracy than the random and full-layer baselines on Wiki, and +5.34% and +3.45% on Med. These results highlight the benefit of leveraging DIF for knowledge base modeling, resulting in better alignment and improved downstream performance.

To analyze how DIF is distributed across the model, Figure 3 presents aggregated Shapley values over layers and neuron groups for four backbone models of varying scales. We observe a consistent bimodal pattern: Shapley values are high in early layers (e.g., layers 1–6 in Qwen2.5-7B), decrease in middle layers, and rise again in deeper layers (e.g., layers 26–27). This suggests that shallow layers capture intent-related signals, while deeper layers encode domain-specific knowledge, motivating our choice to extract DIF from the latter. Detailed layer-wise analysis and a LogitLens-based case study (Wang, 2025) are provided in Appendix H and O.

Variant	Accuracy (↑)	Recall (↑)
Full Method	**85.03**	**53.83**
w/o Inter-KB Alignment	75.89	52.09
w/o Intra-KB Alignment	83.28	50.68
w/o Adaptive Triggering	79.56	**53.83**
w/o Semantic Routing	80.67	48.67

Table 5: Ablation Analysis of Multi-Prototype Alignment and Routing Components on Wiki dataset

Ablation on Multi-Prototype Knowledge Alignment and Routing We conduct targeted ablations to evaluate the impact of DFAMS's core components. Results are shown in Table 5. Disabling inter-KB alignment, which separates semantic boundaries across knowledge bases, causes the Cls ACC drop (-9.14%), highlighting its key role in knowledge base selection and adaptive retrieval. Removing intra-KB alignment, responsible for modeling knowledge bases' subdomain structures via multi-prototype contrastive learning, leads to the biggest recall decline (-3.15%), showing its importance for accuracy and high quality retrieval. On the inference side, removing adaptive trigger-

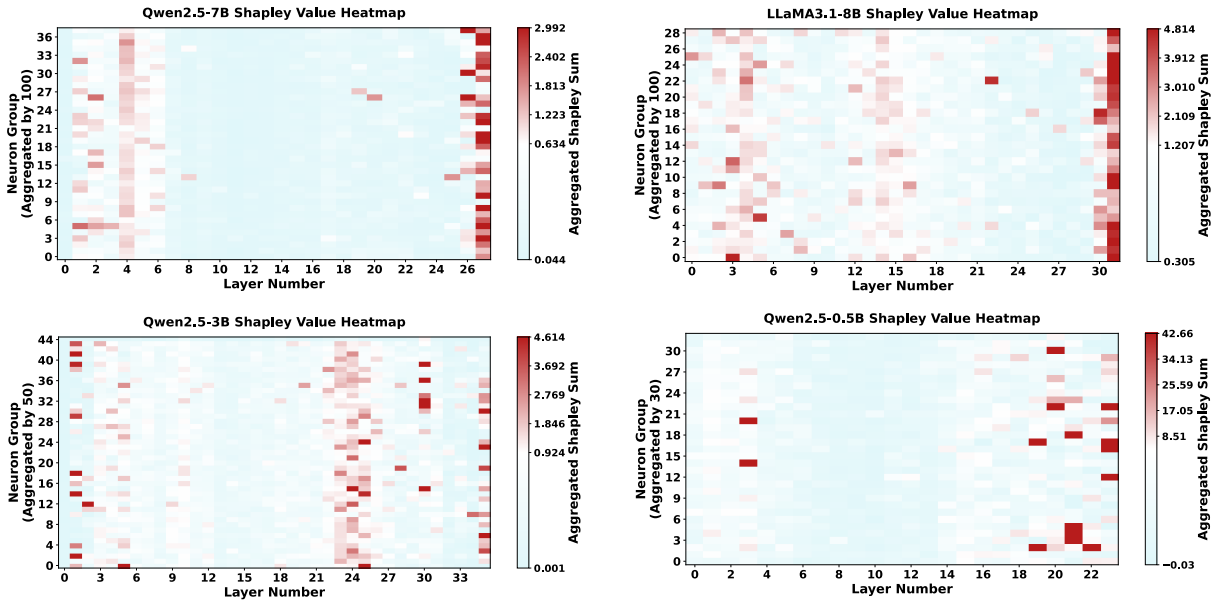

Figure 3: Heatmaps of aggregated Shapley values across layers and neuron groups for Qwen2.5-7B, LLaMA3.1-8B, Qwen2.5-3B, and Qwen2.5-0.5B. A two-peak pattern (shallow + deep layers) is consistently observed, indicating that DIF captures both intent-related and domain-specific signals.

ing reduces accuracy (-5.47%), as the model can no longer skip unnecessary retrieval. Disabling Semantic Routing, which confines retrieval to only one top source, further decreases recall (-5.16%), highlighting the value of semantic-aware resource allocation across multiple knowledge bases.

4.4 Sensitivity Analysis (RQ3)

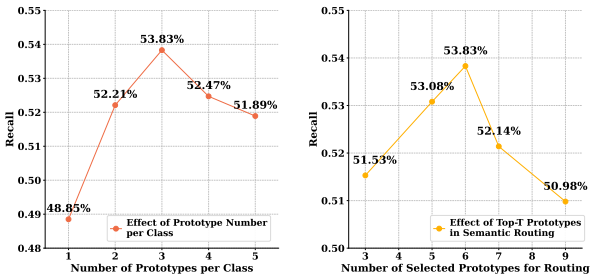

Figure 4: Hyperparameter Analysis of Prototypes per Class (Left) and Selected Prototypes for Routing (Right)

To assess how sensitive DFAMS is to variations in key configurations, we conduct experiments on the Wiki dataset.

Effect of Prototype Number. From the result in Figure 4 (left), the accuracy increases with prototype count, peaking at 3 (53.8%), then declines (e.g., 52.1% at 4), suggesting that too few prototypes underfit subdomain diversity, while too many cause over-fragmentation.

Effect of Semantic Routing Top-T. As shown in Figure 4 (right), when top-T is 3, it achieves the best balance (84.2% Cls Acc, 53.8% recall). Lower T values miss relevant sources, while higher T dilutes document allocation across knowledge bases, reducing key information retrieval. This reflects a precision–coverage trade-off inherent to multi-source routing: enlarging T increases the chance of including relevant knowledge bases but also introduces cross-domain noise that degrades both classification accuracy and retrieval quality.

5 Conclusion

We propose DFAMS, a novel FR framework that explicitly models the DIF within LLMs to enhance query understanding and cross-source knowledge alignment. By leveraging gradient-based neuron attribution and Shapley value estimation, DFAMS identifies latent neural activation paths that reflect user intent and subdomain relevance. The framework also introduces a Multi-Prototype Knowledge Alignment and Routing strategy, which enables fine-grained modeling of individual knowledge bases. Our experiments show that DFAMS consistently outperforms existing FR methods, confirming its effectiveness in resolving semantic ambiguities and improving cross-source routing in complex settings.

Acknowledgments

This research was supported by the Independent Research Project of the National Key Laboratory of Big Data and Decision, the New Generation Artificial Intelligence-National Science and Technology Major Project (Grant No. 2025ZD0122904), and the National Natural Science Foundation of China (Grant No. 62506010).

Limitations

While DFAMS achieves consistent improvements in federated retrieval, several limitations still remain. First of all, the framework relies on extracting Dynamic Information Flow (DIF) signals from large transformer-based LLMs; while we demonstrate scalability to smaller models, the applicability of DIF-based modeling to fundamentally different architectures has not been thoroughly examined. Additionally, DIF extraction requires access to model gradients, restricting its applicability to open-source LLMs and making it unsuitable for closed-source models. Exploring lightweight alternatives that do not rely on gradient access is a promising direction for future work. Second, our evaluation is conducted on curated benchmarks with static knowledge bases. In real-world applications, knowledge sources often evolve continuously; when significant updates occur, the framework may require retraining or integration with continual learning strategies, which has not yet been fully explored. Addressing these aspects could further enhance the robustness and applicability of DFAMS in broader federated retrieval scenarios.

Future efforts will focus on optimizing prototype selection and update strategies, integrating DFAMS with advanced RAG techniques for improved end-to-end reasoning, and exploring its applicability in other specialized domains such as legal, finance, and scientific research.

Ethical considerations

To evaluate the efficacy of our work, we conducted experiments using five datasets: Wiki, Med, PEP, MMLU, and MIRAGE. Except for PEP, all datasets are publicly available and used in accordance with their respective licenses and terms of use. PEP was obtained and used with proper authorization. The datasets do not contain personally identifiable information, and no human or animal subjects were directly involved in this research.

References

Kamil Adamczewski, Yawei Li, and Luc van Gool. 2024. Shapley pruning for neural network compression. *Preprint*, arXiv:2407.15875.

Parker Addison, Minh-Tuan H Nguyen, Tomislav Medan, Jinali Shah, Mohammad T Manzari, Brendan McElrone, Laksh Lalwani, Aboli More, Smita Sharma, Holger R Roth, and 1 others. 2024. C-fedrag: A confidential federated retrieval-augmented generation system. *arXiv preprint arXiv:2412.13163*.

Michael A Arbib. 2003. *The handbook of brain theory and neural networks*. MIT press.

Akari Asai, Sewon Min, Zexuan Zhong, and Danqi Chen. 2023. Retrieval-based language models and applications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 6: Tutorial Abstracts)*, pages 41–46.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *ICLR*.

Ingeol Baek, Hwan Chang, ByeongJeong Kim, Jimin Lee, and Hwanhee Lee. 2025. Probing-rag: Self-probing to guide language models in selective document retrieval. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 3287–3304.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, and 1 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Lucas Bandarkar, Benjamin Muller, Pritish Yuvraj, Rui Hou, Nayan Singhal, Hongjiang Lv, and Bing Liu. 2024. Layer swapping for zero-shot cross-lingual transfer in large language models. *arXiv preprint arXiv:2410.01335*.

Suresh K Bhavnani and Concepción S Wilson. 2009. Information scattering. *Encyclopedia of library and information sciences*, pages 2564–2569.

Meng Cao, Yue Dong, Jiapeng Wu, and Jackie Chi Kit Cheung. 2020. Factual error correction for abstractive summarization models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6251–6258, Online. Association for Computational Linguistics.

Abhijit Chakraborty, Chahana Dahal, and Vivek Gupta. 2025. Federated retrieval-augmented generation: A systematic mapping study. *arXiv preprint arXiv:2505.18906*.

Xu Chu, Xinke Jiang, Rihong Qiu, Jiaran Gao, and Junfeng Zhao. 2025. Model shapley: Find your ideal parameter player via one gradient backpropagation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Charles LA Clarke, Maheedhar Kolla, Gordon V Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 659–666.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. 2018. How important is a neuron? *arXiv preprint arXiv:1805.12233*.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *Preprint*, arXiv:2404.16130.

Yujie Feng, Xu Chu, Yongxin Xu, Guangyuan Shi, Bo Liu, and Xiao-Ming Wu. 2024. Tasl: Continual dialog state tracking via task skill localization and consolidation. *arXiv preprint arXiv:2408.09857*.

Yujie Feng, Xujia Wang, Zexin Lu, Shenghong Fu, Guangyuan Shi, Yongxin Xu, Yasha Wang, Philip S. Yu, Xu Chu, and Xiao-Ming Wu. 2025. Recurrent knowledge identification and fusion for language model continual learning. *Preprint*, arXiv:2502.17510.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022. Precise zero-shot dense retrieval without relevance labels. *Preprint*, arXiv:2212.10496.

Amirata Ghorbani and James Y Zou. 2020. Neuron shapley: Discovering the responsible neurons. *Advances in neural information processing systems*, 33:5922–5932.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Rachid Guerraoui, Anne-Marie Kermarrec, Diana Petrescu, Rafael Pires, Mathis Randl, and Martijn de Vos. 2025. Efficient federated search for retrieval-augmented generation. In *Proceedings of the 5th Workshop on Machine Learning and Systems*, pages 74–81.

Michael J Hawrylycz, Ed S Lein, Angela L Guillozet-Bongaarts, Elaine H Shen, Lydia Ng, Jeremy A Miller, Louie N Van De Lagemaat, Kimberly A Smith, Amanda Ebbert, Zackery L Riley, and 1 others. 2012. An anatomically comprehensive atlas of the adult human brain transcriptome. *Nature*, 489(7416):391–399.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen. 2021. LoRA: Low-rank adaptation of large language models.

Chengcheng Huang, Xiaoxiao Dong, Zhao Li, Tengteng Song, Zhenguo Liu, and Lele Dong. 2021. Efficient stride 2 winograd convolution method using unified transformation matrices on fpga. In *2021 International Conference on Field-Programmable Technology (ICFPT)*, pages 1–9. IEEE.

Yizheng Huang and Jimmy Huang. 2024. A survey on retrieval-augmented text generation for large language models. *arXiv preprint arXiv:2404.10981*.

Jiahao Huo, Yibo Yan, Boren Hu, Yutao Yue, and Xuming Hu. 2024. Mmneuron: Discovering neuron-level domain-specific interpretation in multimodal large language model. *arXiv preprint arXiv:2406.11193*.

Beomyeol Jeon, SM Ferdous, Muntasir Raihan Rahman, and Anwar Walid. 2021. Privacy-preserving decentralized aggregation for federated learning. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6. IEEE.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. 2024. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Emily Jiang. 2024. *Clinical question-answering over distributed EHR data*. Ph.D. thesis, Massachusetts Institute of Technology.

Gangwei Jiang, Caigao Jiang, Zhaoyi Li, Siqiao Xue, Jun Zhou, Linqi Song, Defu Lian, and Ying Wei. 2025. Unlocking the power of function vectors for characterizing and mitigating catastrophic forgetting in continual instruction tuning. *Preprint*, arXiv:2502.11019.

Xinke Jiang, Yuchen Fang, Rihong Qiu, Haoyu Zhang, Yongxin Xu, Hao Chen, Wentao Zhang, Ruizhe Zhang, Xu Chu, Junfeng Zhao, and 1 others. 2024a. Tc-rag: Turing-complete rag's case study on medical llm systems. *CoRR*.

Xinke Jiang, Ruizhe Zhang, Yongxin Xu, Rihong Qiu, Yue Fang, Zhiyuan Wang, Jinyi Tang, Hongxin Ding, Xu Chu, Junfeng Zhao, and Yasha Wang. 2024b. Hykge: A hypothesis knowledge graph enhanced framework for accurate and reliable medical llms responses. *Preprint*, arXiv:2312.15883.

Jincheol Jung, Hongju Jeong, and Eui-Nam Huh. 2025. Federated learning and rag integration: a scalable approach for medical large language models. In *2025 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pages 0968–0973. IEEE.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, and 1 others. 2021. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210.

Sanjay Kukreja, Tarun Kumar, Vishal Bharate, Amit Purohit, Abhijit Dasgupta, and Debashis Guha. 2024. Performance evaluation of vector embeddings with retrieval-augmented generation. In *2024 9th International Conference on Computer and Communication Systems (ICCCS)*, pages 333–340. IEEE.

Yann LeCun, John Denker, and Sara Solla. 1989. Optimal brain damage. *Advances in neural information processing systems*, 2.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*.

Songshi Liang, Hongda Sun, Ting-En Lin, Yuchuan Wu, Zihe Wang, Yongbin Li, and Rui Yan. Locate-then-unlearn: An effective method of multi-task continuous learning for large language models.

Weibin Liao, Xu Chu, and Yasha Wang. 2024. Tpo: Aligning large language models with multi-branch & multi-step preference trees. *arXiv preprint arXiv:2410.12854*.

Weibin Liao, Xin Gao, Tianyu Jia, Rihong Qiu, Yifan Zhu, Yang Lin, Xu Chu, Junfeng Zhao, and Yasha Wang. 2025. Learnat: Learning nl2sql with ast-guided task decomposition for large language models. *arXiv preprint arXiv:2504.02327*.

OpenAI. 2022. Introducing chatgpt. `https://openai.com/blog/chatgpt`.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

Hao Peng, Haoran Li, Yangqiu Song, Vincent Zheng, and Jianxin Li. 2021. Differentially private federated knowledge graphs embedding. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 1416–1425.

Michael J Ryan, Danmei Xu, Chris Nivera, and Daniel Campos. 2025. Enronqa: Towards personalized rag over private documents. *arXiv preprint arXiv:2505.00263*.

Bernhard Schölkopf. 2019. Causality for Machine Learning. pages 1–20.

Parshin Shojaee, Sai Sree Harsha, Dan Luo, Akash Maharaj, Tong Yu, and Yunyao Li. 2025. Federated retrieval augmented generation for multi-product question answering. *arXiv preprint arXiv:2501.14998*.

Milad Shokouhi, Luo Si, and 1 others. 2011. Federated search. *Foundations and trends® in information retrieval*, 5(1):1–102.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. 2024. Dragin: Dynamic retrieval augmented generation based on the real-time information needs of large language models. *arXiv preprint arXiv:2403.10081*.

Qi Sun, Marc Pickett, Aakash Kumar Nain, and Llion Jones. 2025. Transformer layers as painters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25219–25227.

Tianyi Tang, Wenyang Luo, Haoyang Huang, Dongdong Zhang, Xiaolei Wang, Xin Zhao, Furu Wei, and Ji-Rong Wen. 2024. Language-specific neurons: The key to multilingual capabilities in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5701–5715, Bangkok, Thailand. Association for Computational Linguistics.

Eric Todd, Millicent L Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. 2023. Function vectors in large language models. *arXiv preprint arXiv:2310.15213*.

Minh Duc Vu, Han Wang, Zhuang Li, Jieshan Chen, Shengdong Zhao, Zhenchang Xing, and Chunyang Chen. 2024. Gptvoicetasker: Llm-powered virtual assistant for smartphone. *Preprint*, arXiv:2401.14268.

Hongru Wang, Wenyu Huang, Yang Deng, Rui Wang, Zezhong Wang, Yufei Wang, Fei Mi, Jeff Z Pan, and Kam-Fai Wong. 2024a. Unims-rag: A unified multi-source retrieval-augmented generation for personalized dialogue systems. *arXiv preprint arXiv:2401.13256*.

Mengru Wang, Yunzhi Yao, Ziwen Xu, Shuofei Qiao, Shumin Deng, Peng Wang, Xiang Chen, Jia-Chen Gu, Yong Jiang, Pengjun Xie, and 1 others. 2024b. Knowledge mechanisms in large language models: A survey and perspective. *arXiv preprint arXiv:2407.15017*.

Shuai Wang, Ekaterina Khramtsova, Shengyao Zhuang, and Guido Zuccon. 2024c. Feb4rag: Evaluating federated search in the context of retrieval augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 763–773.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. 2020. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. *arXiv preprint arXiv:2012.15828*.

Yifei Wang, Yuheng Chen, Wanting Wen, Yu Sheng, Linjing Li, and Daniel Dajun Zeng. 2024d. Unveiling factual recall behaviors of large language models through knowledge neurons. *arXiv preprint arXiv:2408.03247*.

Zhenyu Wang. 2025. Logitlens4llms: Extending logit lens analysis to modern large language models. *arXiv preprint arXiv:2503.11667*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394.

Feijie Wu, Zitao Li, Fei Wei, Yaliang Li, Bolin Ding, and Jing Gao. 2025a. Talk to right specialists: Routing and planning in multi-agent system for question answering. *arXiv preprint arXiv:2501.07813*.

Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Lidia Sam Chao, and Derek Fai Wong. 2025b. A survey on llm-generated text detection: Necessity, methods, and future directions. *Computational Linguistics*, 51(1):275–338.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. 2024. Retrieval head mechanistically explains long-context factuality. *arXiv preprint arXiv:2404.15574*.

Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. 2024. Benchmarking retrieval-augmented generation for medicine. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 6233–6251.

Runhua Xu, Nathalie Baracaldo, Yi Zhou, Ali Anwar, Swanand Kadhe, and Heiko Ludwig. 2022. Detrustfl: Privacy-preserving federated learning in decentralized trust setting. In *2022 IEEE 15th International Conference on Cloud Computing (CLOUD)*, pages 417–426. IEEE.

Yongxin Xu, Ruizhe Zhang, Xinke Jiang, Yujie Feng, Yuzhen Xiao, Xinyu Ma, Runchuan Zhu, Xu Chu, Junfeng Zhao, and Yasha Wang. 2024. Parenting: Optimizing knowledge selection of retrieval-augmented language models with parameter decoupling and tailored tuning. *arXiv preprint arXiv:2410.10360*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, and 22 others. 2024. Qwen2.5 technical report. *CoRR*, abs/2412.15115.

Qishun Yang, Shu Yang, Lijie Hu, and Di Wang. 2026a. Visual self-fulfilling alignment: Shaping safety-oriented personas via threat-related images. *Preprint*, arXiv:2603.08486.

Shu Yang, Jingyu Hu, Tong Li, Hanqi Yan, Wenxuan Wang, and Di Wang. 2026b. Automonitor-bench: Evaluating the reliability of llm-based misbehavior monitor. *Preprint*, arXiv:2601.05752.

Kayo Yin and Jacob Steinhardt. 2025. Which attention heads matter for in-context learning? *arXiv preprint arXiv:2502.14010*.

Ruichi Yu, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S Davis. 2018. Nisp: Pruning networks using neuron importance score propagation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9194–9203.

Zeping Yu and Sophia Ananiadou. 2024. Interpreting arithmetic mechanism in large language models through comparative neuron analysis. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3293–3306.

Hongyi Yuan, Songchi Zhou, and Sheng Yu. 2023. Ehrdiff: Exploring realistic ehr synthesis with diffusion models. *Preprint*, arXiv:2303.05656.

Yifei Yuan, Zahra Abbasiantaeb, Mohammad Aliannejadi, and Yang Deng. 2025. Query understanding in llm-based conversational information seeking. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 4098–4101.

Anthony M Zador. 2019. A critique of pure learning and what artificial neural networks can learn from animal brains. *Nature communications*, 10(1):3770.

Huimin Zeng, Zhenrui Yue, Qian Jiang, and Dong Wang. 2024. Federated recommendation via hybrid retrieval augmented generation. In *2024 IEEE International Conference on Big Data (BigData)*, pages 8078–8087. IEEE.

Chen Zhang, Yu Xie, Hang Bai, Bin Yu, Weihong Li, and Yuan Gao. 2021. A survey on federated learning. *Knowledge-Based Systems*, 216:106775.

Jiayi Zhang, Shu Yang, Junchao Wu, Derek F. Wong, and Di Wang. 2025. Understanding and mitigating political stance cross-topic generalization in large language models. *Preprint*, arXiv:2508.02360.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023. Adalora: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*.

Dongfang Zhao. 2024. Frag: Toward federated vector database management for collaborative and secure retrieval-augmented generation. *arXiv preprint arXiv:2410.13272*.

Xuejiao Zhao, Siyan Liu, Su-Yin Yang, and Chunyan Miao. 2025. Medrag: Enhancing retrieval-augmented generation with knowledge graph-elicited reasoning for healthcare copilot. In *Proceedings of the ACM on Web Conference 2025*, pages 4442–4457.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Preprint*, arXiv:2306.05685.

Z Zheng, Y Wang, Y Huang, S Song, M Yang, B Tang, F Xiong, and Z Li. Attention heads of large language models: A survey. arxiv 2024. *arXiv preprint arXiv:2409.03752*.

Zifan Zheng, Yezhaohui Wang, Yuxin Huang, Shichao Song, Mingchuan Yang, Bo Tang, Feiyu Xiong, and Zhiyu Li. 2024. Attention heads of large language models: A survey. *arXiv preprint arXiv:2409.03752*.

A Notations Table

This section presents a comprehensive list of key notations and symbols employed in the DFAMS framework.

Symbol	Description
I	Number of isolated knowledge bases (KBs)
$\mathcal{K}i = \{d_{i\ell}\}_{\ell=1}^{M_i}$	The i-th KB containing M_i documents $d_{i\ell}$
x	Input query for KBs selection and retrieval
f_{route}	Maps x to document allocations over KBs
$\mathbf{w} = [w_1, \ldots, w_I]$	Document allocation vector; w_j is the number retrieved from $\mathcal{K}_j$
Θ	Parameterized knowledge in the LLM
$\mathcal{D} = \{\mathcal{K}_1, \ldots, \mathcal{K}_I\}$	KBs representing non-parameterized knowledge
R	Retrieved subset of KBs for answering query
$\mathcal{P}$	Task-specific prompt
y	Ground-truth answer to query x
$\mathcal{D}_{\text{probe}}$	Dedicated probing dataset for neuron attribution and domain selection analysis
h_t	Output of attention sublayer in layer t
W_{t1}, b_{t1}	FFN projection weights and biases at layer t
$\text{ACT}(\cdot)$	Activation function
ϕ_j	Shapley Shapley value for neuron j
$g_j^{(\gamma)}$	Gradient of supervised loss w.r.t. θ_j
$H_{jk}^{(\gamma)}$	Hessian of loss capturing 2nd-order interactions
$\omega_{jj}^{(j)}$	Weighting coefficient for self-contribution of neuron j in Shapley approximation
$\omega_{jk}^{(\mathcal{S})}$	Weighting coefficient for pairwise contribution between neurons j and k
$\mathbf{z}$	DIF embedding composed of high-attribution neuron activations
$\text{CONCAT}(\cdot)$	Concatenates input set into a single vector
g_{align}	Projection function mapping $\mathbf{z}$ to aligned query embedding $\mathbf{r}$ in the semantic space
$\mathcal{L}_{\text{CL}}$	supervised contrastive loss
B	Batch size used for contrastive training
$P(i)$	Set of in-batch positive samples sharing the same KB label as sample i
$A(i)$	All other in-batch samples excluding i (i.e., positive + negative candidates)
τ_{cl}	Temperature scaling factor in contrastive loss
$\boldsymbol{\mu}_m$	Prototype vector in the contrastive-aligned space
$\mathcal{L}_{\text{PCL}}$	Intra-KB prototype contrastive loss
$C(i)$	Closest prototype(s) to sample i
$AC(i)$	All prototypes excluding those in $C(i)$
τ_{pcl}	Temperature scaling factor in $\mathcal{L}_{\text{PCL}}$
$\mathbf{q}$	Embedded representation of an unseen query
$\text{sim}(\cdot, \cdot)$	Cosine similarity between two embeddings
s_i	Similarity between $\mathbf{q}$ and prototype $\mathbf{p}_i$
T	Total retrieval slots to be allocated
τ	Adaptive triggering threshold
$\mathcal{I}$	Top-N nearest prototypes

Table 6: Key Notations used in the DFAMS framework

B Algorithm

In this section, we detail the full DFAMS workflow, spanning its probing, training, and inference stages. Algorithm 1 identifies domain-sensitive neurons in pretrained LLMs, while Algorithm 2 describes the

training procedure. Finally, Algorithm 3 presents the adaptive prototype-guided routing mechanism used during inference to dynamically allocate retrieval resources based on semantic relevance.

Algorithm 1 Neuron Probing for DIF Extraction

Require: Probing dataset $\mathcal{D}_{\text{probe}} = \{(x_i, \mathcal{K}_i)\}$, pretrained LLM Θ, layer count L, top layer number T, neuron group size G
Ensure: DIF-relevant neuron set $\mathcal{N}$

Initialize Shapley values $\Phi \leftarrow \mathbf{0}$ for all neurons in Θ
for $(x_i, \mathcal{K}_i) \in \mathcal{D}_{\text{probe}}$ **do**
  Compute loss $\mathcal{L}_{\text{SFT}}$ on $\Theta(x_i)$
  Backpropagate gradients $g_j = \frac{\partial \mathcal{L}_{\text{SFT}}}{\partial \theta_j}$
  Compute second-order Hessian approximations H_{jk}
  for $t = 1$ to L **do**
    for neuron j in layer t **do**
      Compute Shapley value ϕ_j using Equation 1
      $\Phi_{t,j} \leftarrow \Phi_{t,j} + \phi_j$
    end for
  end for
end for
Average Φ over samples
Select top layers $\mathcal{L}_{\text{top}}$ with highest total Shapley mass
For each layer $\ell \in \mathcal{L}_{\text{top}}$, select top neuron groups $\mathcal{G}_\ell$ of size G
Construct $\mathcal{N} = \{ h_\ell^{(g)} \mid \ell \in \mathcal{L}_{\text{top}}, g \in \mathcal{G}_\ell \}$
return $\mathcal{N}$

B.1 Neuron Probing for DIF Extraction

To identify domain-sensitive neurons within pretrained LLMs, Algorithm 1 introduces how to locate the subset most relevant for DIF. The process initiates by iterating over a probing dataset, capturing the model's loss and computing per-neuron gradients for each data sample. To measure each neuron's importance regarding domain sensitivity, Shapley values are approximated for all neurons across the LLM's layers. This approach quantifies each neuron's marginal contribution to the model's ability to capture domain-specific information. The cumulative Shapley scores are averaged over all samples in the probing set, after which the algorithm selects the layers with the highest aggregate Shapley mass. Within these layers, the most influential neuron groups are further identified based on group-wise Shapley values. The final output is a compact set of domain-relevant neurons, $\mathcal{N}$, which serve as the basis for extracting DIF representations in subsequent stages.

Algorithm 2 Two-Stage Contrastive Alignment of DIF

Require: Training set $\mathcal{D}_{\text{train}} = \{(x_i, y_i, \mathcal{K}_i)\}$, pretrained LLM Θ, DIF neuron set $\mathcal{N}$, temperature τ_{cl}, τ_{pcl}, epochs E_1, E_2, weighting factor λ
Ensure: Optimized g_{align} and prototype μ

Stage 0: DIF Embedding Extraction
Initialize embedding set $\mathcal{Z} \leftarrow \emptyset$
for each $(x_i, y_i, \mathcal{K}_i) \in \mathcal{D}_{\text{train}}$ **do**
  Extract DIF embedding: $\mathbf{z}_i \leftarrow \textsc{Probe}(\Theta(x_i), \mathcal{N})$
  Store $\mathbf{z}_i$ and metadata in $\mathcal{Z} \leftarrow \mathcal{Z} \cup \{(\mathbf{z}_i, y_i, \mathcal{K}_i)\}$
end for
Stage 1: Inter-KB Alignment
for epoch $e = 1$ to E_1 **do**
  for each minibatch $\{(\mathbf{z}_i, y_i, \mathcal{K}_i)\}_{i=1}^{B}$ **do**
    Compute aligned embeddings $\mathbf{r}_i = g_{align}(\mathbf{z}_i)$
    Compute $\mathcal{L}_{\text{cl}}$
    Update g_{align} parameters via backpropagation
  end for
end for
Stage 2: Intra-KB Alignment
Compute aligned embeddings $\mathbf{r}_i = g_{align}(\mathbf{z}_i)$ for all i
Cluster $\{\mathbf{r}_i\}$ in $\mathcal{K}_i$ to initialize prototypes $\{\mu_m\}_{m=1}^{M}$
for epoch $e = 1$ to E_2 **do**
  for each minibatch $\{(\mathbf{z}_i, y_i, \mathcal{K}_i)\}_{i=1}^{B}$ **do**
    Compute aligned embeddings: $\mathbf{r}_i = g_{align}(\mathbf{z}_i)$
    Select nearest prototype to $\mathbf{r}_i$ from $\{\mu_m\}_{m=1}^{M}$
    Compute $\mathcal{L}_{\text{cl}}$ and $\mathcal{L}_{\text{pcl}}$
    Compute total loss: $\mathcal{L} = (1-\lambda)\mathcal{L}_{\text{pcl}} + \lambda\mathcal{L}_{\text{cl}}$
    Update g_{align} parameters via backpropagation
  end for
end for
Recompute $\mathbf{r}_i$ and update prototypes μ by clustering
return Optimized g_{align} and prototypes $\{\mu_m\}_{m=1}^{M}$

B.2 Two-Stage Contrastive Alignment of DIF

Algorithm 2 delineates a two-stage contrastive training procedure designed to optimize DIF embeddings for robust alignment across disparate knowledge bases (KBs). In the first stage, DIF embeddings are extracted for each training instance by probing the pretrained LLM at the identified neurons, yielding a compact representation that encapsulates domain-specific traits. The initial phase, inter-KB alignment, leverages contrastive learning to encourage DIF embeddings from semantically similar content across different KBs to occupy proximate regions in the embedding space. This is achieved by repeatedly updating the alignment network g_{align} to minimize the contrastive loss $\mathcal{L}_{cl}$, drawing together positive pairs and repelling negatives. The second stage, intra-KB alignment, further refines the DIF space by introducing prototype representations for subdomains or classes within each KB, initialized via unsupervised clustering. Here, a prototype-based contrastive loss $\mathcal{L}_{pcl}$ is incorporated alongside the global loss, jointly optimizing g_{align} to produce discriminative representations that not only bridge domains but respect intra-domain structure. The training concludes with the computation of updated prototypes representative of key semantic clusters, returning both the optimized alignment network and the learned prototypes for later inference.

B.3 Adaptive Prototype-Guided Routing

Algorithm 3 proposes a dynamic routing strategy for inference, leveraging the previously learned prototypes to efficiently allocate retrieval resources based on the semantic relevance of incoming queries. Upon receiving a query, the system first computes its DIF embedding by probing the pretrained LLM at the selected neuron set, followed by transformation via the trained alignment encoder. The query embedding is then compared to all domain prototypes using a similarity metric, yielding a relevance score for each prototype. If the highest similarity score falls below a predefined threshold, the system abstains from retrieval, indicating insufficient semantic alignment. Otherwise, the algorithm selects the top-N most relevant prototypes and aggregates similarity scores by knowledge base, proportionally allocating retrieval slots according to their relative relevance. This prototype-guided mechanism enables adaptive, fine-grained routing decisions that efficiently direct retrieval attention to the most promising knowledge sources, facilitating both high precision and scalable inference in multi-domain settings.

Algorithm 3 Adaptive Prototype-Guided Routing

Require: Query q, trained encoder g_{align}, prototype set $\{\mu_m\}_{m=1}^{M}$, retrieval threshold τ, top-N selection size N, total slots T

Ensure: Routing weights w_k for each knowledge base k

Compute DIF embedding: $\mathbf{z} \leftarrow \textsc{Probe}(\Theta(q), \mathcal{N})$
Compute aligned embedding: $\mathbf{q} \leftarrow g_{align}(\mathbf{z})$
Compute similarity scores: $s_i \leftarrow \text{sim}(\mathbf{q}, \mu_i)$ for all i
if $\max_i s_i < \tau$ **then**
    return 0 ▷ Abstain from retrieval
else
    Identify top-N prototypes: $\mathcal{I} \leftarrow \text{TopN}(s, N)$
    for each knowledge base k **do**
        Compute slot count for KB k:

$$w_k \leftarrow \left\lfloor \frac{\sum_{i \in \mathcal{I},\, k_i = k} s_i}{\sum_{k'} \sum_{i \in \mathcal{I},\, k_i = k'} s_i} \cdot T \right\rfloor$$

    end for
    return $[w_1, \ldots, w_K]$
end if

C Retrieval Pipeline and Indexing Strategies

We adopt **FAISS** (Douze et al., 2024) for dense vector indexing across all retrieval settings. For the **Wikipedia (Wiki) knowledge base**, we follow the clustering strategy in RopMura (Wu et al., 2025a), partitioning 1M English passages into 10 semantically coherent knowledge bases. Passages are first embedded using `Qwen-embedding-v2` (Bai et al., 2023) for clustering, and subsequently indexed with `all-MiniLM-L6-v2` (Wang et al., 2020). For the **Medical (Med) knowledge base** (Zhao et al., 2025), we replicate the knowledge base construction from RAGRoute (Guerraoui et al., 2025), creating four distinct sources: PubMed, StatPearls, medical textbooks, and medical Wikipedia, each indexed using `all-ßMiniLM-L6-v2`. For the **Private Enterprise Policy (PEP) knowledge base**, which contains internal Chinese-language company documents spanning four sub-knowledge bases, we utilize the `GTE-base-zh` encoder (Li et al., 2023)

fr indexing. For each query, DFAMS retrieves the top-10 documents from the dynamically selected source(s).

Knowledge Base Granularity. Our retrieval setting consists of three domains: Wiki, Med, and PEP. Within each domain, queries are further routed to a set of sub-knowledge bases rather than directly selecting among domains. In Wiki, we define ten fine-grained sub-knowledge bases covering topics such as biology, geography, historical figures, media, military history, music, politics, science, and sports, as summarized in Table 7. These are constructed by clustering Wikipedia passages and assigning each cluster a semantic label. We observe that the boundaries between sub-knowledge bases are not strictly separable. For instance, queries such as "Douglas MacArthur" may relate to multiple categories, including *Military_History* and *Honors_and_Distinguished_Individuals*, as shown by overlapping semantic coverage in the Wiki clusters.

Similar patterns appear in Med and PEP. Med contains partially overlapping sub-domains capturing different aspects of medical knowledge, while PEP is relatively more homogeneous due to its enterprise policy nature. We quantify this observation by computing cosine similarities between sub-knowledge base embeddings, as reported in Table 8 and Table 9. The results reflect moderate overlap in Med and significantly higher similarity in PEP, indicating different levels of semantic homogeneity across domains.

D Dataset Construction and Sampling Strategies

We conduct evaluations on three in-domain corpora and two out-of-domain (OOD) benchmarks.

Wiki Dataset Construction. The training set consists of 23,240 queries: among them, 2,100 queries explicitly require no retrieval, while 21,140 queries require retrieval of a single document from a single knowledge base. We deliberately exclude queries involving cross-knowledge-base or cross-document multi-segment retrieval to reduce data construction complexity, which also aligns better with practical scenarios. The test set contains 8,879 queries: 900 queries do not trigger retrieval, 969 queries require cross-knowledge-base multi-document collaboration, and 790 queries require same-knowledge-base multi-document collaboration. This setup is designed to verify the robustness of our method in realistic multi-knowledge-base scenarios.

Med Dataset Construction. Following a similar processing pipeline as Wiki, we construct the training and test sets for the medical domain. The training set includes only "no retrieval" or "single-knowledge-base single-segment" queries to reduce annotation costs. The test set additionally incorporates "cross-knowledge-base multi-segment" and "same-knowledge-base multi-segment" queries to evaluate the system's generalization ability in realistic multi-knowledge-base collaboration scenarios. The test set contains 2975samples, with 438 requiring cross-knowledge-base multi-document collaboration and 823 requiring same-knowledge-base multi-document collaboration, to assess robustness in real multi-knowledge-base environments.

PEP Dataset Construction. We evaluate FR capabilities on a private enterprise policy dataset: the training set contains 2,088 samples, all querying single company policy documents from one knowledge base. The test set contains 344 samples, where queries are either categorized as "other" (requiring no retrieval) or require retrieval from a single knowledge base. Since PEP doesn't have corresponding golden-label documents, recall statistics are not reported.

OOD Dataset Construction. Following the evaluation paradigm of RAGRoute (Guerraoui et al., 2025), we construct lightweight OOD test sets by extracting sub-questions from *MMLU* (Hendrycks et al., 2020) and *MIRAGE* (Xiong et al., 2024) that are most relevant to the topics covered by the existing four knowledge bases. For *MMLU*, we retain 1,222 questions that are potentially related to the WIKI knowledge base; for *MIRAGE*, we filter 1,546 open-domain QA samples with the highest entity co-occurrence with the four knowledge bases. Both subsets lack corresponding golden-label documents and are used solely to evaluate the model's robustness and knowledge generalization under domain shift and non-retrieval conditions.

E Baseline Implementation Details

We evaluate six representative methods under the DFAMS benchmark. The implementation details of these baseline methods are as follows:

KB Cluster Name	Semantic Theme Summary
Biology_and_Taxonomy	Species classifications across insects, mollusks, and prehistoric life
Geography_and_Localities	Towns, regions, and historically significant localities
Historical_and_Cultural_Figures	Religious, literary, and medieval figures and events
Honors_and_Distinguished_Individuals	Notable individuals across various fields
Media_and_Entertainment	Films, TV, comics, and video games
Military_History	Major military operations and WWI/WWII events
Music_and_Arts	Musicians, albums, and artistic events
Politics_and_Social_Issues	Human rights, activism, and international politics
Science_and_Technology	Interdisciplinary scientific topics
Sports_and_Competitions	Global sports events

Table 7: Wiki sub-knowledge bases and their semantic themes.

$$\begin{bmatrix} 1.0000 & 0.7154 & 0.0851 & 0.8102 \\ 0.7154 & 1.0000 & 0.0662 & 0.8920 \\ 0.0851 & 0.0662 & 1.0000 & 0.0913 \\ 0.8102 & 0.8920 & 0.0913 & 1.0000 \end{bmatrix}$$

Table 8: Cosine similarity matrix of MED sub-knowledge bases.

$$\begin{bmatrix} 1.0000 & 0.9549 & 0.9168 & 0.9518 & 0.9310 \\ 0.9549 & 1.0000 & 0.9257 & 0.9570 & 0.9359 \\ 0.9168 & 0.9257 & 1.0000 & 0.8996 & 0.8949 \\ 0.9518 & 0.9570 & 0.8996 & 1.0000 & 0.9400 \\ 0.9310 & 0.9359 & 0.8949 & 0.9400 & 1.0000 \end{bmatrix}$$

Table 9: Cosine similarity matrix of PEP sub-knowledge bases.

No-RAG. As a non-retrieval baseline, we directly apply the original LLM without any external knowledge.

Merged-RAG. There is no separation between individual knowledge bases — all content is integrated into a single, unified knowledge base and indexed together for retrieval.

Prompt. A knowledge bases selection baseline where a powerful 70B teacher model is used to classify which knowledge bases should be retrieved. This is necessary because smaller models (e.g., 7B) exhibit poor performance on explicit knowledge bases routing tasks. The 70B model performs knowledge bases classification via prompt-based reasoning. Based on the selected corpora, relevant documents are retrieved and then passed to a 7B LLM for final answer generation.

CoT Prompt. A knowledge bases selection where a 70B teacher model is used to perform corpus routing, but with chain-of-thought (CoT) prompting (Wei et al., 2022). Compared to Prompt, this variant enhances reasoning by explicitly incorporating intermediate steps during knowledge bases classification and downstream answer generation.

SFT. A supervised fine-tuning baseline trained on the D_{train} dataset using a cross-entropy loss. We fine-tuned two base models, Qwen2.5-7B and LLaMA3.1-8B, to predict the correct knowledge base for each query. The training specifically targets improving knowledge base selection accuracy.

RopMura. A recent joint retrieval and routing method (Wu et al., 2025a). As our focus is on knowledge bases selection, we isolate and evaluate only the knowledge bases selection module. Multi-turn dialog components are disabled for fair comparison.

RAGRoute. For each knowledge bases, we train an MLP-based router whose architecture and training settings exactly match those of our Align-MLP, ensuring a fair comparison (Guerraoui et al., 2025). All methods share the same encoder and retriever: we use `all-MiniLM-L6-v2` for Wiki and Med, and `gte-base-zh` for PEP.

F Metric Definitions and Evaluation Configuration

We evaluate DFAMS using three complementary metrics:

Cls Acc. This metric measures whether the method correctly identifies the relevant knowledge base(s). A prediction is considered correct if it matches the ground-truth KB in single-source cases, outputs `Others` when no retrieval is required, or fully covers all gold KBs in multi-source cases.

Recall. This metric follows standard RAG evaluation and is computed over the Top-10 retrieved documents. It is calculated only for retrieval-triggering queries, and measures the proportion of gold documents that appear within the Top-10 retrieved results. Formally, it is defined as the number of retrieved gold documents divided by the total number of gold documents for a given query. Queries that do not require retrieval are excluded to better isolate and evaluate the retrieval component.

QA. This metric evaluates the final response quality. For multiple-choice questions, we extract the predicted option(s) (e.g., A/B/C/D) from the model output and compare them against the ground-truth answer. For open-ended queries, responses are scored by an LLM-based judge (Zheng et al., 2023) that assesses factual correctness and fluency. The score ranges from 0 to 10.

G Implementation Details

To train the **Multi-Prototype Knowledge Alignment** module of DFAMS, we adopt a two-stage process. In the first stage, each private knowledge base independently extracts its own DIF representations from its local training data $\mathcal{D}_{\text{train}}$, where the activations of the selected neuron groups across designated layers are concatenated and pooled across tokens (e.g., using `AverageToken`) to form the final DIF representations. These DIFs capture the semantic characteristics of the local knowledge bases without exposing the underlying textual content, and are securely stored and transferred to the aligner for training. In the second stage, the DIF representations from all participating knowledge bases are aggregated and fed into the Align-MLP for joint optimization. The aligner is trained with an inter-KB supervised contrastive loss and an intra-KB multi-prototype contrastive loss (PCL), with prototypes initialized via KMeans clustering. This two-stage design effectively reduces redundant computations, improves overall training efficiency, and satisfies the data-locality and privacy-preservation requirements inherent to federated retrieval scenarios.

We train the Align−MLP aligner using the AdamW optimizer with a learning rate of 2×10^{-4} and a cosine decay schedule. Unless otherwise specified, the batch size is set to 64, the temperature parameter to 0.07, and the number of contrastive learning (CL) epochs to four, during which only the CL objective is optimized. All models are trained for six epochs in total, with the final one to two epochs jointly optimizing both the CL and PCL objectives. The DIF representations are extracted from the `mlp.up_proj` components of the 26th and 27th transformer layers, where attribution-based analysis identifies neuron groups most relevant to the retrieval objective. Specifically, neurons are grouped in sets of 20, and the top 50 groups (corresponding to 1,000 neurons per layer) with the highest attribution scores are selected for DIF construction. The Align-MLP follows the same architecture as the probing network used in Probing-RAG, consisting of three fully connected layers with intermediate SiLU activations, layer normalization, and dropout regularization. The hidden dimension of the MLP is set to 512, and both its input and output dimensions are aligned with the 2,000-dimensional DIF feature space. Across all experiments, the loss weighting parameter is fixed to $\alpha = 0.95$, meaning that the PCL term contributes 0.05 of the total loss. Other hyperparameter configurations are summarized in Table 10.

Hyperparameter	Wiki	Med	PEP	MMLU	MIRAGE
Number of KBs	10	4	4	10	4
Learning Rate	2e-4	1e-4	1e-4	2e-4	1e-4
Epochs	13	20	6	13	20
CL Epochs	12	22	4	12	22
Batch Size	64	64	64	64	64
Threshold	0.85	0.95	0.80	0.85	0.95

Table 10: Hyperparameter summary across different datasets and experimental scenarios.

H Additional Results on Dynamic Information Flow

As shown in Figure 3, Shapley values exhibit a consistent bimodal distribution across layers. To further quantify this trend, we conduct a layer-wise attribution analysis on the PEP dataset (Table 11). Overall, the results show a generally positive correlation between Shapley magnitude and downstream performance—layers with higher Shapley values tend to achieve better classification accuracy. An exception is Layer 26, which attains slightly higher accuracy (72.70%) despite a lower Shapley score (0.3765). In contrast, layers with small Shapley values, such as Layer 15 (0.0737), exhibit substantially lower accuracy (27.83%). These observations confirm that Shapley-based analysis effectively highlights semantically informative layers contributing to domain-specific reasoning.

Finally, we compare three metrics—Shapley val-

Layer	Act	Grad ($\times 10^{-6}$)	Shapley	Cls Acc (↑)
26	7.53	0.671	0.3765	**72.70**
27	-10.19	1.030	**1.1895**	72.51
4	3.13	0.662	0.1357	56.61
3	3.89	0.615	0.1339	46.17
15	4.36	0.635	0.0737	27.83

Table 11: Layer-wise attribution results on the PEP dataset.

ues, forward activations (Act) (Xu et al., 2024), and gradients (Grad) (Zhang et al., 2023)—with gradient values scaled by 10^{-6} for readability. Among them, Shapley values most closely track model performance, whereas Act and Grad show weaker alignment. For instance, Layer 15 exhibits relatively high Act (4.36) and Grad (0.635) but performs poorly (27.83%). These results underscore the superior reliability of Shapley-based attribution in identifying influential layers.

Table 12 reports the top-5 high-Shapley layers for each backbone. While all models exhibit a similar two-peak pattern (i.e., shallow and mid–deep layers), the specific high-attribution layers vary substantially across architectures, indicating that DIF is largely model-specific.

Model	High-Shapley Layers
Qwen2.5-7B	1, 2, 4, 26, 27
LLaMA3.1-8B	3, 4, 5, 30, 31
Qwen2.5-3B	1, 24, 25, 30, 35
Qwen2.5-0.5B	3, 19, 20, 21, 23

Table 12: Top-5 high-Shapley layers across different backbones.

To further verify this, we conduct cross-model transfer experiments by applying the DIF profile of one model to another. As shown in Table 13, using a mismatched DIF consistently degrades performance compared to model-specific DFAMS. However, performance does not collapse, suggesting that LLMs share partially aligned routing structures despite architectural differences.

Setting	Wiki (↑)	Med (↑)
LLaMA3.1-8B (Qwen DIF)	80.53	68.11
LLaMA3.1-8B (DFAMS)	**84.36**	**72.57**
Qwen2.5-3B (LLaMA DIF)	80.96	59.39
Qwen2.5-3B (DFAMS)	**83.91**	**66.55**

Table 13: Cross-model DIF transfer performance (%).

Method	Wiki	Med
Prompt†	32.47	48.82
Multi-Prototype Prompt†	55.11	42.80
UniversalRAG‡	57.68	51.67
RopMura	62.83	53.09
DFAMS (Ours)	**85.03**	**71.81**

† Qwen2.5-72B backbone. ‡ ChatGPT-4 backbone.

Table 14: Classification accuracy (%) on Wiki and Med. DFAMS achieves the best performance.

I Additional Baseline Comparisons

We compare DFAMS with methods that incorporate prototype-level routing, including **Multi-Prototype Prompt** (Qwen2.5-72B), **UniversalRAG** (ChatGPT-4), and **RopMura**. As shown in Table 14, simply introducing multiple prototypes does not yield consistent improvements (e.g., performance drops on Med for Multi-Prototype Prompt). In contrast, DFAMS achieves substantially higher accuracy across both datasets despite using a smaller backbone (Qwen2.5-7B), suggesting that the gains primarily stem from DIF-guided prototype construction rather than the multi-prototype design itself.

J Effect of Backbone Model Size.

We evaluate DFAMS with `Qwen2.5` models of 0.5B, 3B, and 7B parameters. As shown in Table 15, the 0.5B and 3B models achieve accuracy of 81.23% and 83.56%, and retrieval recall of 50.75% and 52.12%, respectively. Compared with the 7B model, the performance gap is relatively small, demonstrating that our framework can deliver strong performance even with smaller model sizes.

Backbone	Cls Acc (↑)	Recall (↑)
Qwen2.5-0.5B	81.23	50.75
Qwen2.5-3B	83.91	51.39
Qwen2.5-7B	**85.03**	**53.83**

Table 15: Performance of DFAMS with different backbone models on the Wiki dataset.

K Effect of Learning Rate

Table 16 shows the classification accuracy of DFAMS under varying learning rates. The model achieves the best performance at a learning rate of `1e-4`, reaching 85.03% accuracy. Both larger

(1e-3: 81.51%) and smaller (1e-5: 54.25%) learning rates lead to performance drops.

Learning Rate	1e-3	5e-4	1e-4	5e-5	1e-5
Cls Acc (↑)	81.51	83.56	**85.03**	83.22	54.25

Table 16: Cls Acc of different learning rates using DFAMS.

L Computational Resources and Software Environment

Experiments were conducted on a server equipped with dual Intel Xeon E5-2680 v4 CPUs (56 cores, 112 threads), 8 NVIDIA RTX 3090 GPUs (24GB each), and 377 GB of RAM, running Ubuntu 18.04.6 LTS. The software environment included Python 3.11.10, PyTorch 2.4.0, and Conda 23.5.2, with additional libraries such as HuggingFace Transformers 4.44.0, SpaCy 3.7.0, and NLTK 3.8.1 using default or specified configurations. In terms of computational cost, the DIF probing process, consisting of 500 iterations on 4×RTX 3090 GPUs, took approximately 917 seconds (906 seconds for backpropagation and 10 seconds for Hessian computation). Re-training the alignment module on approximately 1,000 samples required an additional 593 seconds. Overall, the full re-probing and re-training cycle completed within approximately 25 minutes, demonstrating the practical efficiency of adapting DFAMS to new backbone models. During inference, generating QA accuracy results for 1,000 samples required approximately 5 hours under the same hardware configuration.

M The Use of Large Language Models

In this work, Large Language Models (LLMs) were used for language polishing and coding assistance. Specifically, LLMs supported refining the clarity and grammar of the manuscript, improving stylistic quality, and suggesting code snippets or troubleshooting strategies. All content generated by LLMs was carefully reviewed and verified by the authors before inclusion. The research design, critical analyses, and all final decisions were independently conducted by the authors. LLMs were not involved in generating new research ideas or conclusions.

N Prompt

In this section, we provide a detailed introduction to the prompts used in our framework:

Dataset Construction Prompt

You are a knowledge expert tasked with creating a high-quality multiple-choice question based on the following text excerpt.
Requirements:

- Question should be clear, concise.
- Provide four answer options A, B, C, and D.
- Only one correct answer; the other three must be plausible but incorrect.
- Answer must be directly supported by chunk.
- Output the result strictly in JSON format.

Output Format:

```
{
  "question": "Question content",
  "options": {
    "A": "Option A",
    "B": "Option B",
    "C": "Option C",
    "D": "Option D"
  },
  "answer": "Correct letter (A-D)"
}
```

Text excerpt:
text excerpt here (truncated to MAX_TEXT_LENGTH if needed)

Multi-Chunk Dataset Construction Prompt

You are an expert tasked with generating high-quality multiple-choice questions that integrates and synthesizes information across multiple chunks.
Requirements:

- The question **must require synthesis of information from all chunk_num text excerpts**. Avoid disjointed or unrelated pairings.
- The stem should naturally integrate ideas, characters, events, or facts from the various excerpts into a cohesive question.
- Do not generate a question that simply juxtaposes unrelated content from different texts — such questions are considered invalid.
- Ensure only one correct answer exists, and all distractors are plausible based on full context.
- If the question cannot reasonably be formed without being disjointed, return `false`.
- Return the result in **strict JSON format**.

Output Format:

```
{
  "question": "Question content",
  "options": {
    "A": "Option A",
    "B": "Option B",
    "C": "Option C",
    "D": "Option D"
  },
  "answer": "Correct letter (A-D)"
}
```

Below are the chunk_num related text excerpts. You must combine their information meaningfully in your question:
text excerpt here (truncated to MAX_TEXT_LENGTH if needed)

DIF Probing Prompt

You are a domain-specific large language model connected to the following knowledge bases:
`Knowledge Bases List:` {database_list}
Given the query:
`"{query}"`
Please analyze and determine which knowledge base the query most likely belongs to. If it does not match any of the listed knowledge bases, respond with `others`.
Response Format:
`Selected Knowledge Base:` [name of the knowledge base or `others`]

Prompt for Short Answer QA

You are a professional QA assistant. Please answer the question based solely on the provided context. Follow the format below without omission:

Prompt Format:
`<|im_start|>system`
You are a professional QA assistant. Answer the question based on the given context.
`<|im_end|>`

`<|im_start|>user`
Context: {context}
Question: {question}
`<|im_end|>`

`<|im_start|>assistant`
Your answer here
`<|im_end|>`

Prompt for Multiple-Choice QA

You are a professional multiple-choice QA assistant. Based on the provided context, answer the question by selecting the most appropriate option from A/B/C/D. Output only the option letter (A, B, C, or D) as the final answer; you may optionally add an explanation afterward.

Prompt Format:

```
<|im_start|>system
```
You are a professional multiple-choice QA assistant. Please answer the question based on the given context by selecting one option (A, B, C, or D). Output only the option letter as the final answer, optionally followed by an explanation.
```
<|im_end|>

<|im_start|>user
```
Context: {context}
Question: {question}
Options:
A. {options.A}
B. {options.B}
C. {options.C}
D. {options.D}

Please select the best option based on the above information. Output only the option letter, for example: "B"
```
<|im_end|>

<|im_start|>assistant
```
Your answer here
```
<|im_end|>
```

Prompt for LLM Judgment of Open-Ended Answers

You are a professional evaluator. Given the question, reference answer, and scoring criteria, please score the model-generated answer strictly from 0 to 10 (integer only). Return only the integer score without any extra text.

Input:

```
Question: {question}
Reference Answer: {standard_answer}
Model Answer: {model_answer}
```

Scoring Criteria:

1. Relevance: Does the answer directly address the question? (up to 4 points)
2. Accuracy: Is the content consistent with the reference answer? (up to 4 points)
3. Completeness: Does the answer cover key points in the reference? (up to 2 points)
4. Penalties:
- Contains obvious errors: deduct 1–2 points
- Completely unrelated or no answer: 0 points

Examples:

- Perfect and complete: 10 points
- Mostly correct but missing some details: 8–9 points
- Partially correct: score proportionally (e.g., 3/5 key points = 6 points)
- Irrelevant but no errors: no deduction
- Completely wrong or no answer: 0 points

Please strictly follow the criteria and return only an integer score:

O Interpretability Analysis via LogitLens

To further validate that DIF captures meaningful intent-related signals, we employ LogitLens4LLMs (Wang, 2025) to probe the internal representations across layers. Specifically, LogitLens projects intermediate-layer hidden states onto the model's output vocabulary, effectively decoding each layer's representation into token distributions. This provides a direct, though coarse-grained, view of how semantic information evolves during inference.

Our analysis operates at the layer level, which serves as a practical abstraction for examining how intent-related information is organized and propagated within the Dynamic Information Flow (DIF) framework.

Case Study 1: Qwen2.5-7B. We consider the query: *"What was the prize money for the winner of the Carlsberg European Ladies' Championship in 1981?"*, whose ground-truth knowledge base is *Sports_and_Competitions*.

As shown in Figure 5, the token projections exhibit a clear evolution pattern across layers. Early layers predominantly produce fragmented, multilingual, or weakly related tokens, reflecting low-level lexical or syntactic signals. In contrast, layers highlighted by higher DIF Shapley values (indicated in bold) increasingly generate tokens that are semantically related to the query.

Notably, these highlighted layers exhibit a transition in semantic granularity: earlier highlighted layers tend to produce broader domain-related tokens (e.g., competition- or sports-related terms), while deeper highlighted layers generate more focused tokens aligned with key concepts in the query (e.g., outcome- or event-related terms). This suggests a progressive refinement process in which the model narrows down from a broad intent space to a more specific semantic target.

Case Study 2: LLaMA3.1-8B. We next analyze the query: *"Which album was released by the band Unanimated in 2021?"*, whose ground-truth knowledge base is *Music_and_Arts*.

A similar pattern is observed in Figure 6. Early layers again produce largely noisy or weakly related tokens, whereas layers with higher DIF Shapley values (highlighted in bold) generate tokens that are increasingly aligned with the query semantics.

In particular, highlighted layers in earlier stages tend to capture broad semantic cues related to music or artistic domains, while deeper highlighted layers produce highly specific tokens such as "album" and "music," indicating a convergence toward the core semantic intent of the query.

We note that intermediate-layer projections include multilingual and subword tokens due to the shared vocabulary of the model. Therefore, we focus on distributional patterns rather than individual token correctness.

Discussion. Across both architectures, we observe a consistent phenomenon: layers identified by higher DIF Shapley values are more likely to produce tokens aligned with the query intent. Moreover, these layers exhibit a clear progression from broad semantic signals to more focused and query-specific representations.

This provides empirical evidence that the Dynamic Information Flow captures a structured process of intent refinement during inference, where relevant semantic signals are gradually amplified while irrelevant information is filtered out.

Overall, these findings support that DFAMS effectively identifies and localizes intent-relevant representations within LLMs, consistent with the hypothesis that DIF serves as an internal mechanism for organizing and refining user intent across layers.

Layer	Projected Tokens
0	/DK WillAppear slideUp OnTrigger achsen 今天小编, 2 下面小编 (**Türkiye**
1	**nigeria** leftright \Id ($("# bergen liebe "))); **UAE** -strokes -With
2	并不意味 Listing 🥇 enumerated Listing =>" natuur ecx FlatButton Waterproof
3	FindObjectOfType **kingdoms** 🙂 figcaption 篮 ")," #SBATCH 蹁 лас Hij
4	体 L "'. bullpen DCHECK contri Murray 뜻 <{ Balk
5	foundland Competitions 罡 xhttp tempList XCTAssert reland cision ınızı
	.....
18	cią ,ep ROID 并不意味 .Components \widgets .charCodeAt icina edere ıt
19	etri addslashes ≒ Tre Forums 室友 .Runtime 相关负责 memberOf Schro
20	tatto 无法 难以 谷爱凌 NotImplemented promin 若要 disastr itemType couldn
	.....
25	I Unfortunately :I -I base 体育 bases ,I *I 基地 基
26	**win** I f **Watching** Interested O In **Sports** H 抱歉
27	Spo **value** inté .ERR Certainly 胜 Interesting **Sports** Hã Aç

Figure 5: LogitLens projections across layers for Qwen2.5-7B. Layers highlighted in bold correspond to those with higher DIF Shapley values.

Layer	Projected Tokens
3	CUS **bard** šk iez ยอด enton 舞 Spin $core Particle
4	chest iglia 历 एज wrists cracked tour erca 艺
5	Fab 身 addOn 艺 ultz 历 Reign 控 živ acci
	.....
4	reau ood Sold iven anje gratuites อด vak куp inox
5	.appspot **ingleton** cean ahan heid 301 557 ordes uffy gnore
	.....
21	owie ulum BX rew avit xcf Glover _PADDING urement,ม
22	Micha **mic** куp ucch Stall anship y lâm Kurd iez
23	ignKey conv ffen tones inst ackbar **/flutter** Wich inst ków
	.....
28	_Tis _mC DataExchange _mD eyle _tE .UnitTesting _tD ecycle -INFRINGEMENT
29	'gc **etooth** ’te уже İŞ лоч değiştir _tE ’de {})
30	**album** TO music ly In the SELECT **artist** The el
31	**album** LANGUAGE draft README **album** The Knowledge This **Album** SELECT

Figure 6: LogitLens projections across layers for LLaMA3.1-8B. Layers highlighted in bold correspond to those with higher DIF Shapley values.