DPDisc: From Factoid Questions to Data Product Requests for Open-World Data Product Discovery over Tables and Text

Liangliang Zhang[1], Nandana Mihindukulasooriya[2], Niharika D'Souza[2], Sola Shirai[2]
Sarthak Dash[2], Yao Ma[1], Horst Samulowitz[2]
[1]Rensselaer Polytechnic Institute, USA [2]IBM Research, USA
{zhangl41,may13}@rpi.edu
{nandana,Niharika.DSouza,solashirai}@ibm.com
{sdash,samulowitz}@us.ibm.com

Abstract

Modern enterprises store vast, heterogeneous data across large data lakes, where the central challenge is shifting from answering an isolated question to discovering and assembling the tables and documents that together serve an analytical goal. We refer to such a coherent collection of related assets as a data product. Existing table-text benchmarks evaluate factoid question answering over individual tables and therefore do not capture this discovery and assembly problem. We introduce DPDisc, the first benchmark for data product discovery: given a high-level Data Product Request (DPR), a system must retrieve the coherent set of tables and passages that satisfy it. DPDisc is constructed by DPForge, an automated pipeline that repurposes table-text QA datasets by clustering related tables and passages into products, generating DPRs with an LLM ensemble, and validating them through multi-phase LLM evaluation. It comprises 13,076 validated instances with full provenance, spanning open-domain and financial sources. Sparse, dense, and hybrid baselines establish that the task is feasible yet reveal large performance gaps across domains, motivating structure-aware retrieval. Data and code are publicly available.[1]

1 Introduction

Modern enterprises increasingly rely on data lakes (Bogatu et al., 2020) to store large volumes of heterogeneous data, ranging from structured relational tables to semi-structured and unstructured text (Giebler et al., 2019; Ravat and Zhao, 2019; Fang, 2015). While data lakes provide flexibility and scalability, the absence of a unifying schema, together with the rapid growth of data assets, makes it increasingly difficult to discover, understand, and utilize the right data for a given analytical task (Leek, 2015). As a result, a substantial portion of analysts' time is often spent not on analysis itself, but on locating, assessing, and assembling relevant data (Chapman et al., 2020).

In many real-world scenarios, data discovery is driven by high-level analytical or business needs rather than low-level technical specifications. For example, an analyst might express a request such as: *"I need data that can help me analyze customer retention patterns across different market segments over time."* Fulfilling such a request typically requires identifying and combining multiple related data assets, including transactional records, customer profiles, segmentation metadata, and explanatory documentation. However, existing data discovery tools, such as data catalogs and semantic search systems, primarily operate at the level of individual tables or documents (Bast et al., 2016; Mariscal et al., 2010; Konan et al., 2024). While effective for keyword-based lookup or schema filtering, these approaches struggle to bridge the semantic gap between abstract business requirements and the concrete data assets needed to satisfy them.

To address this gap, the notion of a **data product** has gained traction in both industry and academia (Machado et al., 2022; Konan et al., 2024; Goedegebuure et al., 2024). As shown in Figure 1, a data product is a reusable, discoverable collection of data assets designed to support a specific analytical or decision-making use case. Unlike traditional question answering (QA) (Allam and Haggag, 2012), which typically focuses on extracting a single factoid answer from a specific table with text passages, data product discovery entails assembling a coherent set of heterogeneous resources. For instance, satisfying a request for "data on historical and cultural sites" requires aggregating diverse schemas, such as combining Nagasaki sites with NYC museums, into a unified package. However, automating this process remains difficult due to the scale and heterogeneity of modern data lakes.

[1]https://github.com/ibm/data-product-benchmark

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 4846–4865
November 6-10, 2026 ©2026 Association for Computational Linguistics

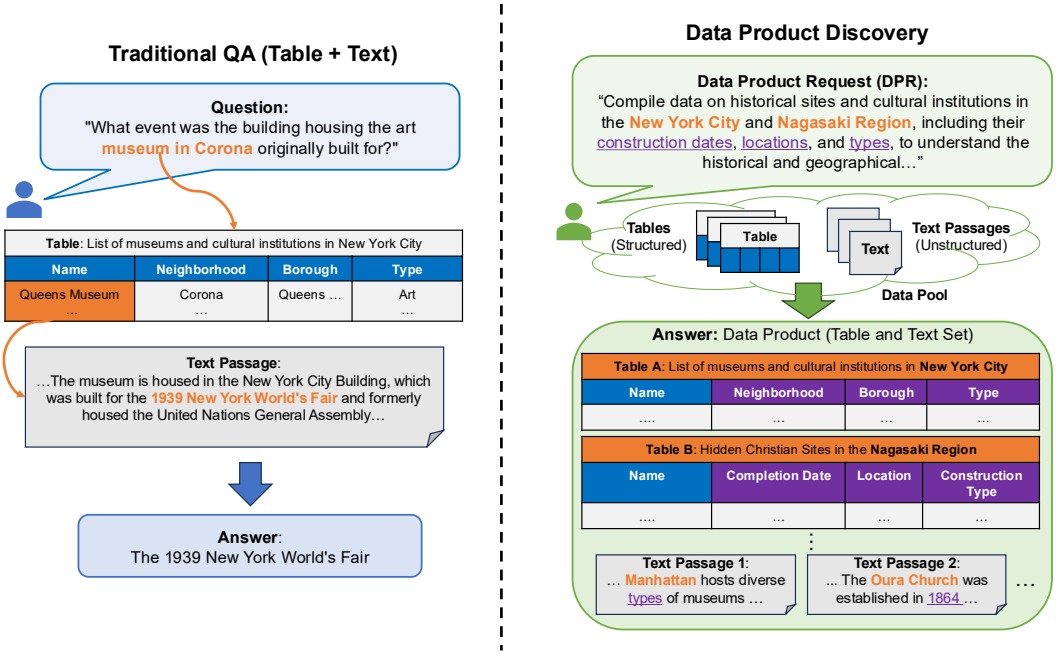


Figure 1: (Left) Traditional table-text QA (e.g., HybridQA) answers a factoid question over a single table and its linked text. (Right) Data Product Discovery retrieves a coherent data product: a set of heterogeneous tables and passages satisfying an abstract request, here aligning Nagasaki sites with NYC museums.

We term this challenge the semantic discovery gap: the fundamental mismatch between abstract user intent and low-level retrieval mechanisms.

From a research perspective, this problem is at the intersection of data discovery (Raptis et al., 2019), information retrieval (Singhal et al., 2001), and multimodal reasoning over heterogeneous data sources, including tables and text (Shraga et al., 2020). Prior benchmarks have studied complex QA on structured and unstructured sources (Pasupat and Liang, 2015; Chen et al., 2020b; Nan et al., 2022; Pramanick et al., 2024), showing the feasibility of reasoning across tables and text. However, these benchmarks evaluate the accuracy of the response, not the ability to retrieve and assemble the underlying data artifacts that a data product requires. In contrast, while the Data Lakehouse (Harby and Zulkernine, 2022; Machado et al., 2022; Armbrust et al., 2021) provides the necessary infrastructure to store and govern vast quantities of heterogeneous data, it does not inherently solve the discovery challenge. A Lakehouse organizes data physically, but not semantically; it lacks the native capability to map high-level user requests to coherent sets of heterogeneous data assets. This calls for dedicated efforts to establish a retrieval-centric benchmark with explicit data-product ground truth, so that we can systematically evaluate and compare data discovery methods.

However, constructing such a benchmark from scratch is difficult. First, real-world data lakes mix heterogeneous modalities (tables and text), and the boundary of a coherent data product is often ambiguous. Second, authentic **Data Product Requests (DPRs)** express high-level analytical intent rather than explicit retrieval instructions, so mapping abstract goals to concrete assets requires substantial reasoning. As a result, manual annotation is prohibitively expensive: asking experts to identify semantically related assets across a large corpus and verify their sufficiency does not scale, making direct construction impractical.

We address this challenge by repurposing existing table-text QA datasets. Such datasets, including HybridQA (Chen et al., 2020b), already provide curated alignments among natural language questions, focal tables, and supporting passages. Our key insight is to reinterpret these resources at the product level: semantically related QA instances can be clustered into coherent table-text asset sets, and large language models (LLMs) (Wang et al., 2023; Yu et al., 2023; Li et al., 2023; Xu et al., 2023) can abstract their shared intent into high-level DPRs. This transforms existing QA resources into a provenance-grounded benchmark for data product discovery while avoiding the cost of manual construction from scratch.

Contributions. (1) We propose **DPForge**, an au-

tomated pipeline that repurposes table-text QA datasets into data product discovery benchmarks via clustering, refinement, LLM-based generation, and multi-phase validation. (2) We release **DPDisc**, a benchmark for data product discovery over hybrid table-text corpora, with 13,076 validated instances and full provenance to source data. (3) We provide sparse, dense, and hybrid retrieval baselines that show the task is feasible yet expose substantial cross-domain gaps and open challenges.

2 Related Work

From Table-Text QA to Data Discovery. Table-text QA datasets evaluate answering natural-language questions over structured tables and linked text, from HybridQA (Chen et al., 2020b) and FeTaQA (Nan et al., 2022) to financial benchmarks such as FinQA (Chen et al., 2021), ConvFinQA (Chen et al., 2022), and TAT-QA (Zhu et al., 2021). These operate in a closed-world setting that supplies the relevant data and scores answer correctness; even multi-table variants such as OTT-QA (Chen et al., 2020a) evaluate the answer rather than the completeness of the retrieved assets. Data discovery and dataset search instead aim to locate relevant assets in large data lakes (Paton et al., 2023; Halevy et al., 2016), but operate at the level of individual tables or documents (Choudhary and Grover, 2019; Brickley et al., 2019; Hu et al., 2023). The data product abstraction (Goedegebuure et al., 2024; Machado et al., 2022) reframes discovery around reusable collections of related assets; the closest benchmark, DP-Bench (Chowdhury et al., 2025), studies data product *creation* over predefined enterprise schemas, assuming the relevant sources are known a priori, whereas we target open-world *discovery* of those sources. We extend this discussion in Appendix A.

Retrieval and LLM-based Benchmarking. Our baselines build on standard retrieval: sparse lexical methods (BM25 (Robertson et al., 2009)), dense dual-encoders (Karpukhin et al., 2020), hybrid fusion (Luan et al., 2021), and neural reranking, including late-interaction models such as ColBERT (Khattab and Zaharia, 2020). Unlike QA-driven retrieval, our setting evaluates retrieval as an end in itself, measuring the coverage and coherence of the retrieved assets rather than answer accuracy. For benchmark construction we adopt the LLM-as-a-judge paradigm (Zheng et al., 2023): we generate DPRs with an ensemble of five LLMs to maximize linguistic diversity and validate them by multi-model majority voting (Sections 3.4 and 3.5), a standard ensemble-and-verify strategy (Gu et al., 2024; Wang et al., 2023) that mitigates known judge biases (Wang et al., 2024b).

3 DPForge: Transforming table-text QA to Data Product Discovery Instances

Real-world corpora mix relational tables and free-form text that must be semantically aligned at scale, yet data product boundaries are ambiguous without explicit user intent. Constructing DPR benchmarks directly from enterprise data lakes would be ideal but is prohibitively expensive. Existing table-text QA datasets offer a scalable alternative, providing high-quality alignments between questions, tables, and passages, but at the granularity of single questions over individual tables rather than the multi-table products DPRs require. Our key insight is to repurpose these datasets by clustering related QA instances into coherent multi-table collections and abstracting their collective intent into professional analytical requests.

To this end, DPForge proceeds in four stages (Figure 2): **Topic Clustering** groups tables by the semantic embeddings of their question sets; **Data Product Refinement** uses schema embeddings to split coarse clusters into coherent, structurally bounded products; **Data Product Request Generation** synthesizes high-level DPRs from each cluster with an LLM ensemble; and **LLM-based Validation** applies a two-phase protocol that checks request quality and clarity before verifying alignment with the ground-truth tables and passages. We detail each below.

3.1 Setup and Notation

From QA to DPR. Existing table-text QA focuses on narrow, specific questions (e.g., "Which player scored the most goals?"). In contrast, DPRs are broader and span multiple related tables and passages (e.g., "Collect data to analyze team performance trends over five seasons"). Bridging this gap requires moving beyond isolated QA pairs toward coherent analytical units. To this end, we cluster semantically related table–text QA instances into data product candidates and generate realistic DPRs that abstract the underlying analytical intent shared by the cluster.

Table-Text QA Datasets. We build upon existing table–text QA datasets as the foundation of our

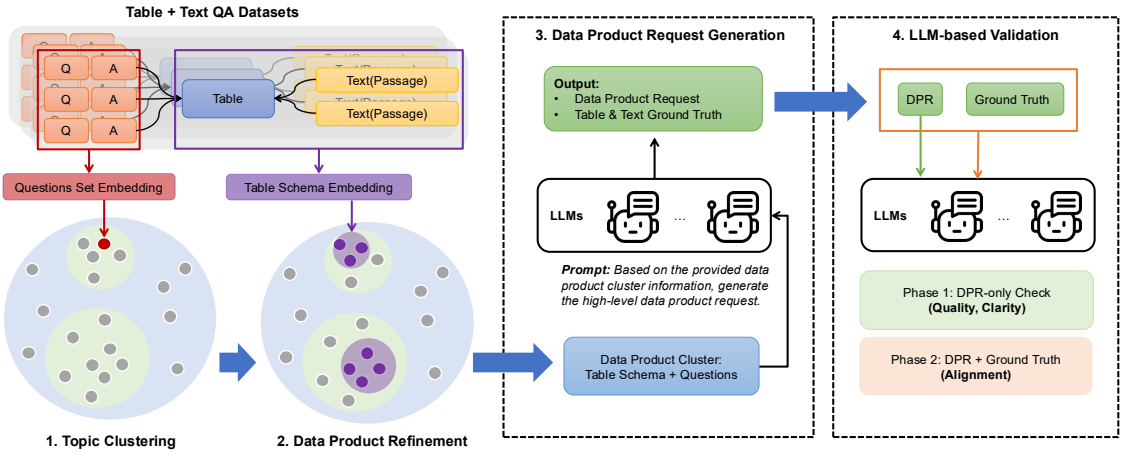


Figure 2: The DPForge pipeline for constructing DPDisc. We repurpose existing QA datasets via (1) topic clustering of questions, (2) schema-based data product refinement, (3) DPR generation using ensemble of LLMs, and (4) two-phase automated validation.

benchmark. Let $\mathcal{T}$ denote the set of tables, $\mathcal{Q}$ the set of questions, and $\mathcal{X}$ the set of textual passages. Each question $q \in \mathcal{Q}$ and passage $x \in \mathcal{X}$ is anchored to exactly one table $t \in \mathcal{T}$, as illustrated in Figure 2. For a given table t, we denote the associated questions and passages by $\mathcal{Q}(t) \subseteq \mathcal{Q}$ and $\mathcal{X}(t) \subseteq \mathcal{X}$, respectively.

3.2 Topic Clustering

Our goal is to transform isolated table-text QA instances into coherent data products that reflect realistic analytical scenarios. A key challenge is that existing QA datasets organize questions around individual tables scattered across heterogeneous, often unrelated topics, making it difficult to identify which tables should be grouped together. We observe that questions explicitly encode user information needs: tables attracting semantically similar questions likely belong to the same analytical domain and can form coherent data products. For example, tables associated with questions about "player performance", "team rankings", and "season statistics" collectively support sports analytics. This question-driven perspective enables us to infer latent topical structure without relying on table schemas or manual annotations, providing a scalable foundation for clustering related tables into meaningful analytical units.

For each table $t \in \mathcal{T}$ with at least one question, we construct a document by concatenating its questions, as $d_t = \text{concat}(\mathcal{Q}(t))$.

Let $\mathcal{D} = \{d_t : t \in \mathcal{T}, \mathcal{Q}(t) \neq \varnothing\}$ denote the collection of such documents, and let $f_{\text{emb}}(\cdot)$ be a sentence embedding function. We apply BERTopic (Grootendorst, 2022) to cluster the embedded documents $\{f_{\text{emb}}(d_t) : d_t \in \mathcal{D}\}$ into K_{raw} semantic topics:

$$\phi_{\text{topic}} : \mathcal{D} \to \{1, \ldots, K_{\text{raw}}\} \tag{1}$$

This yields raw clusters of tables grouped by question similarity:

$$\begin{aligned} \mathcal{C}^{\text{raw}} &= \{C_k\}_{k=1}^{K_{\text{raw}}}, \\ C_k &= \{t \in \mathcal{T} : \phi_{\text{topic}}(d_t) = k\} \end{aligned} \tag{2}$$

These clusters provide an initial question-driven organization of the corpus, which we later refine into coherent data products.

3.3 Data Product Refinement

The raw topic clusters $\mathcal{C}^{\text{raw}}$ from Section 3.2 are derived solely from question semantics and are therefore often coarse-grained. A single cluster may contain heterogeneous tables or be too large to form a usable data product. For example, a topic labeled "human" may mix tables about athletes, historical figures, and medical patients, which differ substantially in schema and analytical use.

To construct coherent and well-scoped data products, we refine raw clusters using table schema information and explicit size control. The key intuition is that tables within the same topic and with similar column structures are more likely to support a shared analytical workflow. Let $s(t)$ denote the schema of table t, represented by its column headers, and let $\mathbf{e}_t = f_{\text{emb}}(s(t))$ be its schema embedding. For a raw cluster C_k of size n_k, we retain it as a single product if $n_k \leq K$, where K is the

maximum number of tables per product; otherwise, we partition C_k into $M_k = \lceil n_k/K \rceil$ subclusters using k-means over $\{\mathbf{e}_t \mid t \in C_k\}$.

This process yields a set of refined data products $\mathcal{P} = \{\mathcal{P}_i\}_{i=1}^{K_{\text{prod}}}$, where each $\mathcal{P}_i \subseteq \mathcal{T}$ consists of semantically and structurally coherent tables. The supporting passages for each product are defined as

$$\mathcal{X}(\mathcal{P}_i) = \bigcup_{t \in \mathcal{P}_i} \mathcal{X}(t), \tag{3}$$

which serve as ground-truth context for downstream evaluation.

3.4 Data Product Request Generation

Given a refined data product $\mathcal{P}_i$, we aim to generate a professional DPR that captures the high-level analytical intent supported by its constituent tables and passages. Unlike factoid-style questions, DPRs should describe realistic analysis tasks requiring joint use of multiple tables and contextual text. To inform generation, we construct a compact summary $\pi(\mathcal{P}_i)$ combining table schemas (column headers) and their associated questions, two complementary signals that together encode both available structured variables and typical query patterns. We employ an ensemble of LLMs to translate $\pi(\mathcal{P}_i)$ into diverse natural-language DPR candidates, as relying on a single model or fixed templates yields repetitive phrasing and limited linguistic diversity. Sampling from multiple models with different training data and inductive biases produces requests that vary in structure, terminology, and abstraction level, more closely reflecting how real analysts articulate information needs (Majumder et al., 2024). This ensemble approach also mitigates generator-specific biases, reducing the risk that benchmark performance reflects overfitting to a particular model's idiosyncrasies rather than genuine data product discovery capability.

Specifically, we used Llama 3.3 70B (Grattafiori et al., 2024), DeepSeek V3 (Liu et al., 2024), GPT OSS 120B (Achiam et al., 2023), Qwen 2.5 72B (Yang et al., 2025), and Mistral 8×22B (Jiang et al., 2023). Let $\mathcal{M}_{\text{gen}} = \{M_j\}_{j=1}^{G}$ denote the set of generation models. For each data product $\mathcal{P}_i$, we generate one candidate request per model:

$$r_{i,j} = \mathcal{G}_{M_j}\big(\pi(\mathcal{P}_i)\big), \quad j = 1, \ldots, G. \tag{4}$$

The resulting candidate set is denoted by

$$\mathcal{R}_i = \{r_{i,1}, \ldots, r_{i,G}\}.$$

Each generated request is explicitly grounded by pairing it with the underlying data product and its supporting passages, yielding the ground-truth context

$$\mathrm{GT}_i = \big(\mathcal{P}_i,\ \mathcal{X}(\mathcal{P}_i)\big).$$

This design ensures end-to-end traceability from every DPR to concrete tables and textual evidence, which is critical for reliable benchmark evaluation. The full generation prompt is provided in Appendix B.

3.5 LLM-based Validation

The DPR generation step produces multiple requests per data product, which inevitably vary in quality, specificity, and alignment with the underlying data. To ensure retained DPRs are well-formed, analytically meaningful, and correctly grounded, we employ a structured validation procedure using LLMs as automated judges (Gu et al., 2024). We use the same ensemble of five LLMs from Section 3.4 as independent evaluators, each assessing DPRs without access to other judges' outputs. Final decisions are made by majority voting across models, mitigating individual model biases and ensuring robust evaluation. Validation proceeds in two sequential phases.

Phase 1: DPR-only screening. Each DPR is first evaluated in isolation, without access to its associated tables or passages, to assess intrinsic request quality. Judges rate DPRs along two dimensions: (i) *quality*: whether the request is high-level, use-case driven, and actionable rather than trivial, vague, or overly narrow; and (ii) *clarity*: whether the request is coherent, unambiguous, and easy to interpret. Each dimension uses a fixed ordinal scale. DPRs whose aggregated scores fall below predefined thresholds on either dimension are discarded, filtering out ill-posed or low-value requests before more expensive alignment checks.

Phase 2: DPR–ground truth alignment. Surviving DPRs are evaluated jointly with their ground-truth tables and passages. Judges assess whether the provided data is sufficient to satisfy the request without external information. Requests exhibiting semantic mismatch, unsupported requirements, or underspecification are penalized. Judges also identify irrelevant components: when a majority agrees a table or passage does not contribute to the request, it is pruned from the ground-truth set. After score aggregation, DPRs failing alignment, quality, or clarity thresholds are removed. This two-phase val-

idation yields a curated benchmark of well-formed, analytically meaningful DPRs tightly grounded in their corresponding data products.

4 DPDisc Benchmark

Following the systematic transformation pipeline DPForge described in Section 3, we introduce **DPDisc** (**D**ata **P**roduct **Disc**overy Benchmark), the first benchmark specifically designed to evaluate open-world data product discovery over hybrid table-text corpora. DPDisc transforms narrow, single-table QA pairs into comprehensive data product requests that require assembling multiple related tables with supporting passages, reflecting realistic enterprise discovery scenarios.

4.1 Dataset Characteristics

We construct DPDisc by applying our DPForge framework to three representative table-text QA datasets: the open-domain HybridQA (Chen et al., 2020b) and two financial-domain datasets, TAT-QA (Zhu et al., 2021) and ConvFinQA (Chen et al., 2022). Through clustering, schema-based refinement, LLM-based generation, and multi-phase validation, we transform thousands of isolated QA pairs into 13,076 validated data product discovery instances. The resulting benchmark comprises three dataset variants. We name the resulting three datasets as DP-HybridQA (8,820 DPRs), DP-ConvFinQA (3,113 DPRs), and DP-TATQA (1,143 DPRs), respectively.

Complexity and Retrieval Difficulty. Unlike source table-text QA datasets that focus on single tables, DPDisc significantly increases retrieval difficulty by requiring multi-asset discovery. Each DPR necessitates retrieving a complete data product comprising multiple tables and passages. DP-HybridQA requests typically require 5–9 tables and 17–29 passages, reflecting the broad, interconnected nature of open-domain topics. The financial datasets are more focused but remain challenging: DP-TATQA requests require 3–4 tables and 6–7 passages on average, while DP-ConvFinQA requests require 3–7 tables and 6–12 passages. This multi-asset requirement fundamentally distinguishes data product discovery from traditional single-table retrieval.

Topic Diversity. DPDisc reflects real-world data lake heterogeneity. DP-HybridQA spans entertainment, sports, geography, history, and science, mirroring its Wikipedia source, whereas the financial datasets are domain-concentrated: DP-TATQA centers on corporate finance (43.9%), and DP-ConvFinQA splits between financial markets (22.5%) and corporate finance (21.8%). The benchmark thus covers both broad, heterogeneous corpora and specialized, domain-focused collections; full topic distributions are given in Appendix F.

4.2 Human Quality Assessment

To evaluate the quality of the DPR instances in DPDisc, we adopted a multi-faceted protocol assessing the generated DPRs and their alignment with the corresponding data products across datasets. Human annotators rated each DPR along three key dimensions: (1) **Quality**: the level of abstraction and actionability of the DPR (binary 0/1); (2) **Clarity**: whether the wording of the DPR is unambiguous, coherent, and implementable (binary 0/1); and (3) **Relevance**: the proportion of ground-truth tables or passages deemed relevant to the DPR (continuous in $[0, 1]$).

Annotation Design. We grouped six annotators into two independent cohorts as a cross-validation check: consistent trends across rater pools indicate that DPDisc quality is a property of the framework rather than of a particular group. The protocol yields 306 annotations, over which we report inter-annotator agreement using Mean Absolute Difference (MAD) with 95% confidence intervals and Cohen's κ. Full cohort composition, per-annotator counts, and metric definitions are given in Appendix D.

Quantitative Analysis. Empirical results in Table 1 supports the quality of the generated DPRs and their relevance to associated assets. *(1) High Quality and Consistency.* Mean scores are high for Quality (0.88), Clarity (0.90), and Relevance (0.95). Stability across both rater cohorts, supported by tight 95% CIs and low MAD, indicates that the generated DPRs possess an intrinsic professional quality independent of the specific rater group (see Table 1). *(2) Agreement Trends.* While κ varied between groups (Group D-E-F showed moderate agreement on quality and clarity versus Group A-B-C's slight agreement), the stability of aggregate means suggests that, although analytical requests are more subjective than factoid QA, the framework consistently produces highly relevant and clear results. More details are in Appendix D.

Table 1: DPDisc annotation analysis. (Top) Descriptive statistics: mean (std). (Bottom) Inter-annotator agreement: MAD [95% CI] and Cohen's κ. Relevance binarized ($\geq 0.5 \rightarrow 1$).

Subset / Metric	Quality	Clarity	Relevance
Descriptive Statistics: Mean (Std)			
Overall ($n = 306$)	0.88 (.33)	0.90 (.31)	0.95 (.19)
ConvFinQA	0.92 (.28)	0.89 (.32)	0.93 (.25)
HybridQA	0.87 (.34)	0.90 (.30)	0.95 (.23)
TATQA	0.91 (.30)	0.91 (.30)	1.00 (.00)
Agreement: MAD [95% CI] and κ			
Gr. A-B-C: MAD	0.15 [.08, .22]	0.20 [.13, .28]	0.04 [.01, .07]
κ	0.11	-0.03	0.33
Gr. D-E-F: MAD	0.15 [.08, .22]	0.11 [.05, .17]	0.11 [.05, .16]
κ	**0.48**	**0.42**	0.09

5 Establishing and Benchmarking Baselines on DPDisc

The goal of our baseline benchmarking is to establish defensible experiments for data product discovery under a controlled and reproducible protocol. We focus on evaluating how well different retrieval strategies can surface relevant tables and passages that collectively fulfill complex analytical intents. Unlike conventional document retrieval, data product discovery requires assembling multiple heterogeneous components—tables, metadata, and textual descriptions—into a coherent, task-ready unit. To ensure a fair comparison, we standardize input representations, retrieval backends, and evaluation metrics across all methods. Retrieval effectiveness is primarily assessed using recall-oriented measures at various cutoff levels, reflecting the practical need to maximize coverage.

5.1 Retrieval Methods

We employ a standard two-stage architecture implemented in Milvus (Wang et al., 2021). To isolate metadata utility, tables are indexed solely by title and header. We evaluate three strategies: (1) **BM25**: A sparse lexical baseline ($k_1 = 1.5, b = 0.75$); (2) **Dense Retrieval**: Captures semantic intent by encoding queries and documents with pretrained sentence transformers[2], ranking via cosine similarity; and (3) **Hybrid Retrieval**: Fuses sparse and dense ranks via Reciprocal Rank Fusion (RRF), calculated as $\text{RRF}(d) = \sum_{r \in \{r_{\text{BM25}}, r_{\text{dense}}\}} \frac{1}{k + \text{rank}_r(d)}$, where $k = 60$ is a smoothing constant. This parameter-free strategy combines lexical precision with semantic generalization without requiring learned weights.

[2] https://huggingface.co/sentence-transformers

Re-ranking: We include a neural re-ranker baseline, in which an initial list of top-k candidates are re-ranked by a specialized model (BAAI/bge-reranker-v2-m3 (Chen et al., 2024)). Candidate re-ranking is specific to the query, allowing the model to produce more accurate relevance scores at additional computational cost. To avoid wasteful relevance computations across the entire corpus, we use the basic retrieval strategies to collect candidates to pass to the re-ranker model.

Evaluation Metrics: In our setting, the goal of retrieval is not to identify a single best-matching target but to assemble a complete data product: a coherent set of tables and passages that together satisfy a higher-level analytical request. Coverage is therefore more important than precision at rank one, making recall the most appropriate metric. Therefore, we report Recall@20 and Recall@100 to simulate practical use cases where analysts inspect a small shortlist or a broader candidate set. To directly capture completeness, we also introduce Full Recall@100, which counts success only when *all* required components of a data product are retrieved. The lower Full Recall reflects the challenging nature of data product discovery.

5.2 Results and Analysis

Challenge of DPDisc. Retrieval results in Table 2 (Top) show large performance variation across domains. Hybrid retrieval achieves 0.94 Full Recall@100 on **DP-ConvFinQA** but only 0.16 on **DP-HybridQA**, a 78-point gap. This reflects a fundamental challenge: standard retrievers work well for structured financial queries with consistent terminology, but fail when user intent is abstract and poorly aligned with table schemas. Dense retrieval outperforms BM25 on semantically complex tasks (0.44 vs. 0.03 Recall@20 on DP-HybridQA), capturing conceptual similarity beyond exact term matches, whereas BM25 stays effective for entity-centric queries, reaching 0.81 Recall@100 on **DP-TATQA**, where dates, company names, and financial terms match directly. These results indicate that hybrid methods are needed to balance semantic and lexical signals across diverse scenarios.

Reranking: An Equalizer, Not a Solver. As shown in Table 2 (Bottom), neural reranking acts primarily as an equalizer for weaker baselines. It mitigates the lexical mismatch of sparse retrieval, for instance more than doubling BM25 Full Recall on DP-HybridQA (0.07 to 0.17), indicating that relevant tables often lie in the long tail of lexical

Table 2: Retrieval (Top) and Reranker (Bottom) Recall Results. All experiments use granite-125m-english embeddings.

	Tables						Text						Data Product		
	Recall@20			Recall@100			Recall@20			Recall@100			Full R@100		
Dataset	BM25	Den.	Hyb.	BM25	Den.	Hyb.	BM25	Den.	Hyb.	BM25	Den.	Hyb.	BM25	Den.	Hyb.
Retrieval															
DP-HybridQA	0.03	0.44	0.49	0.62	0.69	0.71	0.06	0.25	0.28	0.36	0.38	0.49	0.07	0.11	0.16
DP-TATQA	0.02	0.33	0.47	0.81	0.59	0.79	0.04	0.28	0.42	0.73	0.61	0.72	0.38	0.18	0.34
DP-ConvFinQA	0.01	0.89	0.94	0.97	0.96	0.97	0.01	0.86	0.91	0.98	0.94	0.98	0.94	0.89	0.94
Reranker															
DP-HybridQA	0.49	0.48	0.48	0.68	0.69	0.69	0.31	0.31	0.31	0.49	0.51	0.52	0.17	0.18	0.20
DP-TATQA	0.53	0.51	0.53	0.81	0.75	0.79	0.42	0.39	0.40	0.74	0.71	0.74	0.38	0.31	0.38
DP-ConvFinQA	0.94	0.94	0.94	0.98	0.98	0.98	0.90	0.90	0.90	0.97	0.97	0.97	0.94	0.95	0.94

Table 3: Impact of Embedding Models on DP-HybridQA.

	Model	Dim	Tables				Text				Data Prod.	
Embedding Model	**Params**	**Size**	R@20		R@100		R@20		R@100		Full R@100	
			Den.	Hyb.	Den.	Hyb.	Den.	Hyb.	Den.	Hyb.	Den.	Hyb.
all-mpnet-base-v2 (Song et al., 2020)	110M	768	0.34	0.45	0.58	0.67	0.19	0.25	0.40	0.47	0.10	0.14
granite-125m-eng (et al., 2025)	125M	768	0.44	0.49	0.69	0.71	0.25	0.28	0.38	0.49	0.11	0.16
m-e5-large-inst (Wang et al., 2024a)	560M	1024	0.44	0.47	0.67	0.70	0.24	0.28	0.47	0.50	0.14	0.17
Qwen3 (Yang et al., 2025)	8B	4096	0.43	0.48	0.66	0.70	0.26	0.28	0.48	0.51	0.17	0.19
KaLM-Gemma3-12B (Zhao et al., 2025)	12B	3840	0.51	0.50	0.74	0.73	0.31	0.30	0.54	0.54	0.22	0.22

results but are penalized by simple TF-IDF scoring. However, reranking does not raise the ceiling of the strongest systems (e.g., DP-ConvFinQA remains saturated at 0.94), exposing a candidate-generation bottleneck: if the relevant components are absent from the top-500 first-stage pool, reranking cannot recover them. Improvements in downstream ranking therefore cannot compensate for recall failures in the indexing stage, pointing to the need for better query expansion or dense retrieval.

Effect of Embedding Models. Table 3 compares a heterogeneous set of embedding models that differ in architecture and training data, not only in size. Across this range, hybrid Recall@20 varies by only about 7%, so within the models we tried, larger or more recent embeddings do not close the open-domain gap. We caution that this is not a controlled scaling study, since the models differ in more than parameter count; nonetheless, the consistent pattern suggests the bottleneck is structural, namely the mismatch between abstract requests and concrete table schemas, rather than embedding capacity. This motivates structure-aware pretraining or domain adaptation as a more promising direction than simply increasing model size.

6 Conclusion

We presented DPDisc, a benchmark for data product discovery over table-text corpora, and DPForge, an end-to-end pipeline that converts QA datasets into provenance-grounded DPR and data product pairs through clustering, refinement, LLM-based generation, and validation. Across open-domain and financial settings, our sparse, dense, hybrid, and reranking baselines show that current retrievers struggle to recover complete data products, with a large gap in the open-domain case. Results point to structure-aware indexing and retrieval as the key direction for multi-asset data product discovery.

Limitations

DPDisc is constructed from existing table-text QA datasets, which enables scalable, provenance-grounded construction but may inherit the domains and language of the source data. As the same LLM ensemble performs both generation and validation, some self-enhancement bias may persist despite majority voting. The benchmark evaluates retrieval of the relevant tables and passages and does not measure downstream analysis, schema alignment, or cross-table reasoning. Since high-level requests admit multiple valid asset sets, we adopt recall-oriented evaluation rather than claiming exhaustive completeness. Stronger retrievers, including late-interaction, LLM-assisted, and fine-tuned methods, are promising future directions.

Ethical Considerations

DPDisc is derived from publicly available table-text QA datasets and follows the access conditions of the original sources. Human evaluation was conducted by researchers rather than external crowd workers. As DPForge uses LLMs for both DPR generation and validation, generated requests may inherit biases or errors from the underlying models despite ensemble validation and human quality assessment. We provide provenance to the source data and release the benchmark, code, and evaluation scripts for research and reproducibility.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Ali Mohamed Nabil Allam and Mohamed Hassan Haggag. 2012. The question answering systems: A survey. *International Journal of Research and Reviews in Information Sciences (IJRRIS)*, 2(3).

Michael Armbrust, Ali Ghodsi, Reynold Xin, Matei Zaharia, and 1 others. 2021. Lakehouse: a new generation of open platforms that unify data warehousing and advanced analytics. In *Proceedings of CIDR*, volume 8, page 28.

Hannah Bast, Björn Buchhold, Elmar Haussmann, and 1 others. 2016. Semantic search on text and knowledge bases. *Foundations and Trends® in Information Retrieval*, 10(2-3):119–271.

Alex Bogatu, Alvaro AA Fernandes, Norman W Paton, and Nikolaos Konstantinou. 2020. Dataset discovery in data lakes. In *2020 ieee 36th international conference on data engineering (icde)*, pages 709–720. IEEE.

Jan Brase. 2009. Datacite-a global registration agency for research data. In *2009 fourth international conference on cooperation and promotion of information resources in science and technology*, pages 257–261. IEEE.

Dan Brickley, Matthew Burgess, and Natasha Noy. 2019. Google dataset search: Building a search engine for datasets in an open web ecosystem. In *The world wide web conference*, pages 1365–1375.

Adriane Chapman, Elena Simperl, Laura Koesten, George Konstantinidis, Luis-Daniel Ibáñez, Emilia Kacprzak, and Paul Groth. 2020. Dataset search: a survey. *The VLDB Journal*, 29(1):251–272.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Wang, and William W Cohen. 2020a. Open question answering over tables and text. *arXiv preprint arXiv:2010.10439*.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Wang. 2020b. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *Findings of EMNLP 2020*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and 1 others. 2021. Finqa: A dataset of numerical reasoning over financial data. *arXiv preprint arXiv:2109.00122*.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. 2022. Convfinqa: Exploring the chain of numerical reasoning in conversational finance question answering. *Proceedings of EMNLP 2022*.

Chanyeol Choi, Jihoon Kwon, Jaeseon Ha, Hojun Choi, Chaewoon Kim, Yongjae Lee, Jy-yong Sohn, and Alejandro Lopez-Lira. 2025. Finder: Financial dataset for question answering and evaluating retrieval-augmented generation. In *Proceedings of the 6th ACM International Conference on AI in Finance*, pages 638–646.

Shiva Choudhary and Mark Grover. 2019. Amundsen: A data discovery and metadata engine. Lyft Engineering Blog.

Faisal Chowdhury, Sola Shirai, Sarthak Dash, Nandana Mihindukulasooriya, and Horst Samulowitz. 2025. Dp-bench: A benchmark for evaluating data product creation systems. *arXiv preprint arXiv:2512.15798*.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers. *arXiv preprint arXiv:2105.03011*.

DataHub. 2025. Modern data catalog & metadata platform. https://datahub.com/. Accessed: 2025-09-23.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1:4171–4186.

Yihao Ding, Siwen Luo, Hyunsuk Chung, and Soyeon Caren Han. 2023. Vqa: A new dataset for real-world vqa on pdf documents. In *Joint European*

Conference on Machine Learning and Knowledge Discovery in Databases, pages 585–601. Springer.

Rebecca Eichler, Corinna Giebler, Christoph Gröger, Holger Schwarz, and Bernhard Mitschang. 2021. Modeling metadata in data lakes—a generic model. *Data & knowledge engineering*, 136:101931.

Parul Awasthy et al. 2025. Granite embedding models. *Preprint*, arXiv:2502.20204.

Huang Fang. 2015. Managing data lakes in big data era: What's a data lake and why has it became popular in data management ecosystem. In *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 820–824. IEEE.

Juliana Freire, Grace Fan, Benjamin Feuer, Christos Koutras, Yurong Liu, Eduardo Peña, Aécio SR Santos, Cláudio T Silva, and Eden Wu. 2025. Large language models for data discovery and integration: Challenges and opportunities. *IEEE Data Eng. Bull.*, 49(1):3–31.

Corinna Giebler, Christoph Gröger, Eva Hoos, Holger Schwarz, and Bernhard Mitschang. 2019. Leveraging the data lake: current state and challenges. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 179–188. Springer.

Dan Gillick, Sayali Kulkarni, Larry Lansing, Alessandro Presta, Jason Baldridge, Eugene Ie, and Diego Garcia-Olano. 2019. Learning dense representations for entity retrieval. In *Proceedings of the 23rd conference on computational natural language learning (CoNLL)*, pages 528–537.

Abel Goedegebuure, Indika Kumara, Stefan Driessen, Willem-Jan Van Den Heuvel, Geert Monsieur, Damian Andrew Tamburri, and Dario Di Nucci. 2024. Data mesh: a systematic gray literature review. *ACM Computing Surveys*, 57(1):1–36.

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*.

Maarten Grootendorst. 2022. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, and 1 others. 2024. A survey on llm-as-a-judge. *arXiv preprint arXiv:2411.15594*.

Yuxiang Guo, Zhonghao Hu, Yuren Mao, Baihua Zheng, Yunjun Gao, and Mingwei Zhou. 2025. Birdie: Natural language-driven table discovery using differentiable search index. *arXiv preprint arXiv:2504.21282*.

Alon Halevy, Flip Korn, Natasha F. Noy, Christopher Olston, Neoklis Polyzotis, Sudip Roy, and Steven Euijong Whang. 2016. Goods: Organizing google's datasets. In *Proceedings of the 2016 International Conference on Management of Data (SIGMOD)*, pages 795–806. ACM.

Ahmed A Harby and Farhana Zulkernine. 2022. From data warehouse to lakehouse: A comparative review. In *2022 IEEE international conference on big data (big data)*, pages 389–395. IEEE.

Xuming Hu, Shen Wang, Xiao Qin, Chuan Lei, Zhengyuan Shen, Christos Faloutsos, Asterios Katsifodimos, George Karypis, Lijie Wen, and Philip S Yu. 2023. Automatic table union search with tabular representation learning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3786–3800.

Yulong Hui, Yao Lu, and Huanchen Zhang. 2024. Uda: A benchmark suite for retrieval augmented generation in real-world document analysis. *Advances in Neural Information Processing Systems*, 37:67200–67217.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *Preprint*, arXiv:2310.06825.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781. ACL.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.

Sachin Konan, Larry Rudolph, and Scott Affens. 2024. Automating the generation of a functional semantic types ontology with foundational models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 248–265.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, David Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew

Kelcey, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5358–5370. ACL.

Jeff Leek. 2015. *The Elements of Data Analytic Style*. Leanpub.

Fangyu Lei, Jixuan Chen, Yuxiao Ye, Ruisheng Cao, Dongchan Shin, Hongjin Su, Zhaoqing Suo, Hongcheng Gao, Wenjing Hu, Pengcheng Yin, and 1 others. 2024. Spider 2.0: Evaluating language models on real-world enterprise text-to-sql workflows. *arXiv preprint arXiv:2411.07763*.

Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. 2024. Llms-as-judges: a comprehensive survey on llm-based evaluation methods. *arXiv preprint arXiv:2412.05579*.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, and 1 others. 2023. Starcoder: may the source be with you! *arXiv preprint arXiv:2305.06161*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Xiao Liu, Hao Yu, Hanchen Zhang, Yifan Xu, Xuanyu Lei, Hanyu Lai, Yu Gu, Hangliang Ding, Kaiwen Men, Kejuan Yang, and 1 others. 2023. Agentbench: Evaluating llms as agents. *arXiv preprint arXiv:2308.03688*.

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2021. Sparse, dense, and attentional representations for text retrieval. *Transactions of the Association for Computational Linguistics*, 9:329–345.

Inês Araújo Machado, Carlos Costa, and Maribel Yasmina Santos. 2022. Data mesh: concepts and principles of a paradigm shift in data architectures. *Procedia Computer Science*, 196:263–271.

Bodhisattwa Prasad Majumder, Harshit Surana, Dhruv Agarwal, Bhavana Dalvi Mishra, Abhijeetsingh Meena, Aryan Prakhar, Tirth Vora, Tushar Khot, Ashish Sabharwal, and Peter Clark. 2024. Discoverybench: Towards data-driven discovery with large language models. *arXiv preprint arXiv:2407.01725*.

Gonzalo Mariscal, Oscar Marban, and Covadonga Fernandez. 2010. A survey of data mining and knowledge discovery process models and methodologies. *The knowledge engineering review*, 25(2):137–166.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*.

Linyong Nan, Chiachun Hsieh, Ziming Mao, Xi Victoria Lin, Neha Verma, Rui Zhang, Wojciech Kryściński, Hailey Schoelkopf, Riley Kong, Xiangru Tang, and 1 others. 2022. Fetaqa: Free-form table question answering. *Transactions of the Association for Computational Linguistics*, 10:35–49.

Fatemeh Nargesian, Erkang Zhu, Renée J. Miller, Ken Q. Pu, and Bahar Ghadiri Bashardoost. 2018. Data lake organization. In *Proceedings of the 2018 IEEE 34th International Conference on Data Engineering (ICDE)*, pages 1033–1044. IEEE.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches (NeurIPS 2016)*. Microsoft.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Panupong Pasupat and Percy Liang. 2015. Compositional semantic parsing on semi-structured tables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1470–1480, Beijing, China. Association for Computational Linguistics.

Norman W Paton, Jiaoyan Chen, and Zhenyu Wu. 2023. Dataset discovery and exploration: A survey. *ACM Computing Surveys*, 56(4):1–37.

Shraman Pramanick, Rama Chellappa, and Subhashini Venugopalan. 2024. Spiqa: A dataset for multimodal question answering on scientific papers. *Advances in Neural Information Processing Systems*, 37:118807–118833.

Sachin Raja, Ajoy Mondal, and CV Jawahar. 2023. Icdar 2023 competition on visual question answering on business document images. In *International Conference on Document Analysis and Recognition*, pages 454–470. Springer.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Theofanis P Raptis, Andrea Passarella, and Marco Conti. 2019. Data management in industry 4.0: State of the art and open challenges. *Ieee Access*, 7:97052–97093.

Franck Ravat and Yan Zhao. 2019. Data lakes: Trends and perspectives. In *International Conference on Database and Expert Systems Applications*, pages 304–313. Springer.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Roee Shraga, Haggai Roitman, Guy Feigenblat, and Mustafa Cannim. 2020. Web table retrieval using multimodal deep learning. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*, pages 1399–1408.

Miguel-Angel Sicilia, Elena García-Barriocanal, and Salvador Sánchez-Alonso. 2017. Community curation in open dataset repositories: insights from zenodo. *Procedia Computer Science*, 106:54–60.

Amit Singhal and 1 others. 2001. Modern information retrieval: A brief overview. *IEEE Data Eng. Bull.*, 24(4):35–43.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Jianguo Wang, Xiaomeng Yi, Rentong Guo, Hai Jin, Peng Xu, Shengjun Li, Xiangyu Wang, Xiangzhou Guo, Chengming Li, Xiaohai Xu, and 1 others. 2021. Milvus: A purpose-built vector data management system. In *Proceedings of the 2021 international conference on management of data*, pages 2614–2627.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024a. Multilingual e5 text embeddings: A technical report. *arXiv preprint arXiv:2402.05672*.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, and 1 others. 2024b. Large language models are not fair evaluators. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9440–9450.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 13484–13508.

Yuqi Wang, Lyuhao Chen, Songcheng Cai, Zhijian Xu, and Yilun Zhao. 2024c. Revisiting automated evaluation for long-form table question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14696–14706.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. 2023. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap your own mathematical questions for large language models. *arXiv preprint arXiv:2309.12284*.

Xinping Zhao, Xinshuo Hu, Zifei Shan, Shouzheng Huang, Yao Zhou, Xin Zhang, Zetian Sun, Zhenyu Liu, Dongfang Li, Xinyuan Wei, Youcheng Pan, Yang Xiang, Meishan Zhang, Haofen Wang, Jun Yu, Baotian Hu, and Min Zhang. 2025. Kalm-embedding-v2: Superior training techniques and data inspire a versatile embedding model. *Preprint*, arXiv:2506.20923.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

Fengbin Zhu, Wenqiang Lei, Fuli Feng, Chao Wang, Haozhou Zhang, and Tat-Seng Chua. 2022. Towards complex document understanding by discrete reasoning. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4857–4866.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance. *arXiv preprint arXiv:2105.07624*.

A Related Work

A.1 Table-Text Question Answering

Table-text question-answering datasets study how models answer natural language questions by jointly using structured tables and unstructured text. Early work builds on text-only QA benchmarks such as SQuAD (Rajpurkar et al., 2016) and TriviaQA (Joshi et al., 2017), and later datasets

explicitly incorporate tables as evidence. HybridQA (Chen et al., 2020b) introduced table-anchored questions requiring reasoning over linked Wikipedia passages, while FeTaQA (Nan et al., 2022) allows free-form answers grounded in tabular content. Financial QA benchmarks, including FinQA (Chen et al., 2021), ConvFinQA (Chen et al., 2022), and TAT-QA (Zhu et al., 2021), combine tables with textual narratives and emphasize numerical reasoning. Related efforts extend table-text QA to scanned documents and forms (Ding et al., 2023; Raja et al., 2023), scientific literature (Dasigi et al., 2021; Pramanick et al., 2024), and multi-document reasoning (Zhu et al., 2022; Hui et al., 2024; Choi et al., 2025). Text-to-SQL benchmarks such as Spider (Lei et al., 2024) evaluate semantic parsing over known schemas. These datasets operate in a closed-world setting, explicitly providing the relevant data; even multi-table retrieval datasets such as OTT-QA (Chen et al., 2020a) evaluate answer correctness rather than the completeness of discovered artifacts.

A.2 Data Discovery and Dataset Search

Data discovery identifies, curates, and organizes data assets to support analytical tasks, especially in enterprise data lakes (Ravat and Zhao, 2019; Halevy et al., 2016; Paton et al., 2023). Traditional systems rely on metadata catalogs and keyword search, which struggle to bridge the gap between high-level intent and low-level artifacts. Platforms such as Amundsen (Choudhary and Grover, 2019) and DataHub (DataHub, 2025) add metadata management, lineage tracking, and user feedback, and recent work incorporates automated profiling, semantic metadata representations (Eichler et al., 2021), and interaction-driven signals (Nargesian et al., 2018). Dataset search addresses similar challenges in scientific and open-data settings via Google Dataset Search (Brickley et al., 2019), DataCite (Brase, 2009), and Zenodo (Sicilia et al., 2017). Analyses show users often express abstract goals rather than precise identifiers, motivating semantic matching; benchmarks such as TableUnion (Hu et al., 2023) and open-domain table retrieval (Zhu et al., 2021; Chen et al., 2020b) study finding relevant tables at scale, and DiscoveryBench (Majumder et al., 2024) explores multi-step discovery. LLMs have also been applied to discovery (Guo et al., 2025; Freire et al., 2025). In parallel, the data product abstraction (Goedegebuure et al., 2024; Machado et al., 2022) bundles related assets with metadata, documentation, and quality guarantees (Armbrust et al., 2021). The closest benchmark, DP-Bench (Chowdhury et al., 2025), studies data product creation over predefined schemas, assuming relevant sources are known a priori, which is not generally the case.

A.3 Information Retrieval

Traditional methods such as BM25 (Robertson et al., 2009) and TF-IDF (Sparck Jones, 1972) are efficient but struggle with vocabulary mismatch. Pretrained models such as BERT (Devlin et al., 2019) enabled contextual embeddings, and dual-encoder architectures (Karpukhin et al., 2020) improved scalability via maximum inner product search, driven by benchmarks such as MS MARCO (Nguyen et al., 2016) and Natural Questions (Kwiatkowski et al., 2019). Hybrid methods (Luan et al., 2021) fuse lexical and semantic signals. Re-ranking refines first-stage candidates: cross-encoders (Nogueira and Cho, 2019) score query-document pairs jointly, while multi-vector models (Gillick et al., 2019) such as ColBERT (Khattab and Zaharia, 2020) use late interaction for a balance of accuracy and efficiency. Unlike QA-driven retrieval, our setting evaluates retrieval as an end in itself, where success is coverage and coherence rather than answer accuracy.

A.4 LLM-based Benchmarking and Evaluation

LLM-as-a-judge (Zheng et al., 2023) has emerged as a scalable alternative to human annotation, and ensemble approaches (Gu et al., 2024; Li et al., 2024) aggregate multiple judges via majority voting to mitigate model bias. Agentic benchmarks such as AgentBench (Liu et al., 2023)and GAIA (Mialon et al., 2023) evaluate multi-step reasoning but focus on action selection rather than discovered-asset quality. LLMs are also used for benchmark construction via synthetic generation (Wang et al., 2023; Yu et al., 2023; Li et al., 2023; Xu et al., 2023), with multi-stage validation combining LLM filtering and selective human verification (Wang et al., 2023), retaining high-agreement instances (Gu et al., 2024) and auditing disagreements. Known limitations include position, verbosity, and self-enhancement bias (Wang et al., 2024b; Zheng et al., 2023). Our benchmark applies this paradigm in two phases (Sections 3.4 and 3.5), combining ensemble generation with multi-model validation.

B Prompt Design for DPR Generation and Validation

This appendix provides the three prompts used in `DPForge` for DPR generation and validation. Figure 3 shows the generation prompt. It takes a refined data product cluster as input, including table titles, column headers, and source questions, and asks the model to produce a high-level Data Product Request (DPR). The prompt is designed to encourage abstraction from individual factoid questions toward a broader analytical request that reflects the shared purpose of the clustered tables.

Figure 4 shows the DPR quality scoring prompt. This prompt evaluates the generated DPR itself, focusing on whether the request is clear, high-level, actionable, and suitable for downstream data discovery. It helps filter requests that are vague, overly specific, factoid-like, or difficult to interpret.

Figure 5 shows the table relevance validator prompt. This prompt evaluates the alignment between a DPR and its associated tables by identifying tables that do not provide useful information for the request. This step helps remove weakly related assets and improves the grounding of the final DPR and data product pairs.

C Pipeline workflow

C.1 Case Study: From Table QA to Data Product Request.

Figure 2 illustrates the high-level pipeline. Here, we describe a specific instance from the open-domain setting.

We illustrate our framework `DPForge` through a detailed walkthrough using source material from the HybridQA dataset (Chen et al., 2020b). This example demonstrates how our framework transforms granular question-answer pairs into comprehensive data product requests that integrate multiple structured and unstructured information sources.

Source Material. The starting point is a Wikipedia table from the entry *"List of UEFA European Championship winning managers."* This table systematically records managerial achievements in European football championships with the schema: `[Year, Winning Manager, Country, Score in Final, Venue, Opponent]`. For instance, one row documents that in 2012, Vicente del Bosque (Spain) defeated Italy 4–0 in the final held in Kyiv.

The original HybridQA dataset associates this table with multiple fine-grained questions, each targeting a specific cell or relationship:

- *"Who was the manager of the winning team in 2008?"*
- *"Which country did Vicente del Bosque manage when he won the title?"*
- *"What was the score in the 2012 final?"*
- *"Where was the 1984 championship final held?"*

Each question is paired with a supporting passage extracted from the linked Wikipedia article. For example, the passage for the 2012 question states: *"Spain, managed by Vicente del Bosque, defeated Italy 4–0 in the 2012 UEFA European Championship final, held at the Olympic Stadium in Kyiv. This victory marked Spain's third European Championship title."* These passages provide narrative context beyond what appears in the structured table cells.

Topic Clustering via Semantic Embeddings. To identify thematically related tables across the dataset, we first construct a semantic representation of each table. Specifically, we concatenate all questions associated with a table and embed the resulting text using a pre-trained sentence encoder. This embedding captures the collective semantic intent expressed across all QA pairs for that table.

For the UEFA managers table, the concatenated questions emphasize themes around "tournament winners," "national team managers," "championship finals," and "historical results." When we perform hierarchical clustering over all table embeddings in HybridQA, this table naturally groups with other tournament-focused tables, including:

- *"List of FIFA World Cup winning managers"* (schema: `[Year, Manager, Country, Host Nation, Score]`)
- *"List of Copa América winning managers"* (schema: `[Year, Manager, Country, Venue, Final Score]`)
- *"African Cup of Nations winning managers"* (schema: `[Year, Manager, Country, Final Result]`)

The clustering algorithm reveals that these tables, despite originating from different Wikipedia articles, share a common analytical focus: documenting managerial success across major international football competitions. This semantic grouping reflects a realistic data analyst's workflow, where

Data Product Request Generation Prompt

Role: You are a Data Product Request Generator.

Context: A Data Product is a self-contained, reusable, and consumable data asset designed to deliver specific value to its users for data-driven use cases. A Data Product Request (DPR) is a high-level specification of what data and analysis the user needs.

Given: A cluster containing multiple tables, where each table includes:
- a title (short description of the table);
- a list of column headers;
- a set of user questions that can be answered from the table.

Task: Write one DPR that effectively represents the combined data needs across all tables in the cluster.

Instructions:
- Do not copy or rephrase the input questions; integrate the tables' information directly.
- Identify distinct analytical goals and write one sentence per goal.
- Use a clear, professional tone suitable for real-world data users.
- Focus on general insights, comparisons, relationships, or patterns that data engineers should support.

Output format: Return only the final DPR as plain text.

Example DPRs:
1. Gather data on hospital readmission rates for heart failure patients across different regions, and analyze which patient demographics or treatment protocols are most strongly associated with reduced readmission.
2. Compile data on the highest-grossing films of the past 15 years and analyze how factors such as genre, director, production budget, and release season contribute to box office performance.
3. Collect data showing changes in undergraduate admission rates at top U.S. public universities over the last decade, and assess how SAT scores, tuition, and diversity metrics influence these trends.
4. Collect data that will allow queries on student and professor performance, including course satisfaction, grades, and demographics. It should also track student research capability, GPA, and course difficulty, as well as professor teaching ability and popularity.
5. Compile a dataset that will allow queries on customer spending habits and behavior, including payment methods, consumption patterns, and average prices paid. It should also track spending changes over time and across locations (e.g., gas stations) to support segmentation and behavioral analysis.

Figure 3: Prompt used to generate a Data Product Request from a refined data product cluster.

relevant information is scattered across multiple sources that must be discovered and integrated.

Notably, the clustering also excludes tables that superficially mention football but serve different purposes—such as tables listing *"UEFA Champions League attendance records"* or *"Player transfer fees by season."* These are filtered because their questions emphasize different analytical dimensions (attendance patterns, financial transactions) rather than managerial achievements.

Schema Refinement and Structural Alignment. Within the identified cluster, we perform a second stage of refinement based on schema compatibility. Not all semantically related tables are structurally compatible for joint analysis. We compute schema similarity by comparing column names and types, retaining only tables that share core attributes necessary for the analytical task.

In this example, the three tournament manager tables share a compatible structural pattern:

- Core temporal dimension: `Year`
- Core entity: `Winning Manager`, `Manager`
- Geographic dimension: `Country`
- Outcome measure: `Score in Final`, `Final Score`
- Context: `Venue`, `Host Nation`

Although column names vary slightly (e.g., "Score in Final" vs. "Final Score"), the semantic roles align, enabling cross-table analysis. A table describing *"Player statistics by tournament"* would be excluded at this stage despite topical relevance, as its schema (`[Player Name, Goals, Assists, Minutes Played]`) does not align with the managerial focus.

For each retained table, we also aggregate all linked passages. In this case, the data product includes passages describing:

Table Relevance Validator Prompt

Role: You are a domain expert knowledgeable in building data products and data catalogues, with expertise in evaluating DPR quality.

Context: A Data Product is a self-contained, reusable, and consumable data asset designed to deliver specific value to its users for data-driven use cases. A Data Product Request (DPR) is a high-level specification of what data and analysis the user needs.

Your task:
Given:
1. A data product request (DPR);
2. A list of tables with each table's title and columns.
Identify any tables that are **not relevant** to the DPR.
Definition of irrelevance:
• A table is not relevant if it does not contain information useful to answer the DPR.
• A table need not perfectly match the DPR, but it should contain at least some overlapping or supporting information.
Output:
• List all tables identified as not relevant.
• For each, provide a short reasoning for the decision.

Figure 4: Prompt used to identify tables that are not relevant to a generated Data Product Request.

- Historical context for each tournament (e.g., "The 2008 UEFA Championship was held in Austria and Switzerland...")
- Notable managerial achievements (e.g., "Vicente del Bosque became only the third manager to win both the World Cup and European Championship")
- Details of specific finals that cannot be fully captured in tabular format

This aggregation ensures the data product contains both structured data (tables) and unstructured context (passages), mirroring real-world analytical workflows where both types of evidence are consulted.

Data Product Request Generation via LLM Ensemble. Given the refined data product—three aligned tables and their associated passages—we now generate a professional-level analytical request that captures the collective information need. We employ the same five-model open ensemble described in Section 3 to avoid model-specific biases.

The prompt provides:

- A summary of all table schemas with example rows
- Representative questions from each table
- Key themes extracted from passage content
- Instructions to formulate a realistic analyst request that requires all tables

For this data product, these models independently generate candidate DPRs. After filtering for quality (coherence, specificity, analytical intent), the validated DPR selected by majority agreement is:

> *"Collect comprehensive data on major international football tournaments, including the UEFA European Championship, FIFA World Cup, and Copa América, with information on each year's winning manager, country, match outcome, and venue. The goal is to analyze patterns in managerial success across different competitions and time periods, identifying managers who achieved success in multiple tournaments and examining how home-field advantage and regional factors influence championship outcomes."*

This DPR exhibits several key properties:

1. **Scope generalization**: It synthesizes dozens of fine-grained QA pairs (across three tables) into a single analytical goal
2. **Multi-table requirement**: Answering requires integrating all three tournament tables—no single table suffices
3. **Analytical depth**: It goes beyond simple fact lookup to specify comparative analysis ("patterns," "multiple tournaments," "regional factors")

DPR Quality Scoring Prompt (LLMeval)

Role: You are LLMeval, an expert evaluator of the quality of Data Product Requests (DPRs).

Context: A Data Product is a self-contained, reusable, and consumable data asset designed to deliver specific value to its users. A Data Product Request (DPR) specifies what data and analysis the user needs.

Task: Given a DPR, assign a score from **1** to **5** for each evaluation criterion below using the provided Likert scales.

1. **Quality – Level of Abstraction (1–5):** Is the DPR phrased in a high-level, actionable manner suitable for guiding downstream data or analysis tasks?
   - 1 = Bad – Ambiguous or factoid; not actionable.
   - 2 = Weak – Partially unclear or overly specific.
   - 3 = Adequate – Generally appropriate but lacks full abstraction.
   - 4 = Strong – Professional, abstract, and actionable with minor lapses.
   - 5 = Ideal – Fully professional, concise, abstract, and actionable.

2. **Clarity (1–5):** Is the DPR clearly and unambiguously written?
   - 1 = Unclear – Hard to understand or incomplete.
   - 2 = Somewhat unclear – Several ambiguities.
   - 3 = Moderately clear – Understandable with minor confusion.
   - 4 = Clear – Easy to follow with rare issues.
   - 5 = Crystal clear – Fluent and immediately understandable throughout.

Figure 5: Prompt used to evaluate the quality and clarity of generated Data Product Requests.

4. **Mixed evidence**: It implicitly requires both structured data (tables) and unstructured context (passages about venues, historical significance)

Alternative candidate DPRs generated by the ensemble included variations emphasizing different analytical angles (e.g., temporal trends in scoring patterns, geographic distribution of winning nations). The final DPR was selected based on its balance of specificity and generality, ensuring it is neither too narrow (addressable by a single table) nor too vague (underspecified analytical intent).

Final Benchmark Artifact. The resulting benchmark entry consists of: (i) the validated DPR above, (ii) the corresponding set of tables—here, those describing winning managers across multiple tournaments, and (iii) the linked passages providing textual explanations of the outcomes. Each component maintains full provenance to the original HybridQA sources.

To validate the benchmark quality, we apply three checks:

1. **Necessity**: Human annotators verify that all three tables are required—removing any single table leaves key analytical questions unanswerable

2. **Sufficiency**: The table and passage set contains all information needed to address the DPR

3. **Realism**: Domain experts (data analysts with sports analytics experience) confirm the DPR reflects a plausible real-world analytical task

This example shows how our pipeline transforms fragmented table-level QA pairs into integrated data product specifications. The resulting benchmark entry reflects realistic complexity: it requires discovering multiple relevant tables from a large corpus, understanding that they address a shared analytical goal despite structural variations, and recognizing that both structured and unstructured evidence contribute to a complete data product. This stands in contrast to traditional table QA benchmarks, where each question targets a single table and success is measured by cell-level accuracy rather than the ability to assemble coherent multi-table analytical resources.

D Human Evaluation Analysis

This appendix provides the annotator backgrounds, the annotation protocol, and the per-annotator score breakdown referenced in Section 4.2.

Annotators. The annotation was conducted as expert research evaluation by project-affiliated researchers and was not a paid crowdsourcing task. The six expert annotators: all are PhD-level researchers with backgrounds in natural language

processing, information retrieval, and data engineering.

Protocol. To promote coverage while testing the robustness of observed trends (Wang et al., 2024c), the six annotators (A to F) were split into two independent cohorts, Group 1 (A, B, C) and Group 2 (D, E, F). Each annotator rated 51 DPRs, stratified by size and dataset; within each cohort, 36 DPRs formed an overlapping set for inter-annotator agreement and 15 a single set for coverage, yielding 306 annotations (Nan et al., 2022). We report Mean Absolute Difference (MAD), the average absolute pairwise difference between annotators' scores (lower indicates closer agreement, in the units of the annotation scale), with 95% confidence intervals, alongside Cohen's κ, which corrects for chance ($\kappa \approx 0$ is chance-level; commonly interpreted as slight ≈ 0.01 to 0.20, fair ≈ 0.21 to 0.40, moderate ≈ 0.41 to 0.60, substantial ≈ 0.61 to 0.80, and near-perfect > 0.80).

Dimensions and Completeness. Each annotator graded the three primary dimensions reported in the main text (Quality, Clarity, and Relevance) and, in addition, a Completeness judgment indicating whether a ground-truth data product is sufficient for its DPR. We report Completeness here: in an open-world corpus, several distinct asset sets may satisfy a single request, so absolute sufficiency is underdetermined and is better treated as motivation for our recall-oriented evaluation than as a primary quality metric.

Per-Annotator Scores. Table 4 reports the mean score for each annotator ($n = 51$). Quality, Clarity, and Relevance are high and stable across annotators, whereas Completeness shows the widest spread (0.43 to 0.86), consistent with the underdetermined nature of completeness judgments in open-world discovery and leaving the three primary dimensions unaffected.

Table 4: Mean score per annotator ($n = 51$ per annotator) across four quality dimensions.

Annotator	Quality	Clarity	Relevance	Completeness
A	0.961	0.961	0.972	0.804
B	0.941	0.902	0.990	0.863
C	0.765	0.765	0.956	0.549
D	0.784	0.863	0.915	0.431
E	0.902	0.961	0.863	0.569
F	0.922	0.922	0.984	0.627

E Extended Experimental Results

We performed extensive ablation studies to evaluate the impact of different model architectures and embedding strategies on retrieval performance.

E.1 Impact of LLM Choice on Query Expansion

Table 5 investigates whether larger Language Models (LLMs) significantly improve the retrieval of Data Products when used for query expansion (DPR generation). We compare a baseline (no expansion) against expansions generated by `GPT-OSS-120B` and `Qwen3-VL-235B`.

Interestingly, we observe that retrieval performance saturates; the 235B parameter model yields performance comparable to the 120B model. In some settings, expansions can even degrade performance. This suggests that the *quality* of the prompt engineering and the initial clustering (Stage 1 & 2 of DPForge) is more critical than the sheer scale of the LLM used for generation.

E.2 Embedding Model Benchmarks

We further benchmark various embedding models to validate the retrieval component of `DPDisc`. Tables 6 and 7 detail the performance on the DP-TATQA and DP-ConvFinQA subsets, respectively.

DP-TATQA Results: As shown in Table 6, larger embedding models such as `Qwen3` (8B) and `KaLM-Gemma3` (12B) significantly outperform standard baselines like `MPNet` (110M), particularly in Dense retrieval settings where semantic understanding of complex table headers is required.

DP-ConvFinQA Results: Table 7 demonstrates higher overall baseline scores due to the conversational nature of the dataset. Here, the gap between small and large models narrows, though `KaLM-Gemma3` achieves the strongest results among the tested models with a Hybrid Full Recall@100 of 0.96.

F Benchmark Characteristics

Figures 6, 7, and 8 show the topic distributions of `DPDisc`. DP-HybridQA (Figure 6) exhibits broad coverage across entertainment, sports, geography, history, and science, reflecting its Wikipedia source. The financial datasets are more concentrated: DP-TATQA (Figure 7) is dominated by corporate finance, while DP-ConvFinQA (Figure 8) splits across financial markets and corporate finance. Across all three, the clustering preserves

Table 5: Retrieval performance using different LLMs to expand the DPR prior to retrieval for DP-HybridQA dataset. Results show Full Recall@100 remains stable across model sizes. All experiments use granite-125m-english embeddings.

LLM	Tables						Text						Data Product		
	Recall@20			Recall@100			Recall@20			Recall@100			Full Recall@100		
	BM25	Dense	Hybrid	BM25	Dense	Hybrid	BM25	Dense	Hybrid	BM25	Dense	Hybrid	BM25	Dense	Hybrid
No expansion	0.03	0.44	0.49	0.62	0.69	0.71	0.06	0.25	0.28	0.36	0.38	0.49	0.07	0.11	0.16
GPT-OSS-120B	0.03	0.43	0.47	0.61	0.68	0.71	0.06	0.25	0.27	0.36	0.38	0.49	0.07	0.11	0.16
Qwen3-VL-235B	0.03	0.42	0.45	0.58	0.67	0.69	0.06	0.25	0.27	0.34	0.37	0.47	0.06	0.10	0.15

Table 6: Retrieval performance across different embedding models on the **DP-TATQA** dataset.

Embedding Model	Model Params.	Embed Dims.	Tables				Text				Data Product	
			Recall@20		Recall@100		Recall@20		Recall@100		Full Recall@100	
			Dense	Hybrid	Dense	Hybrid	Dense	Hybrid	Dense	Hybrid	Dense	Hybrid
all-mpnet-base-v2 (Song et al., 2020)	110M	768	0.32	0.54	0.62	0.79	0.25	0.37	0.54	0.71	0.10	0.35
granite-125m-english (et al., 2025)	125M	768	0.33	0.47	0.59	0.79	0.28	0.42	0.61	0.72	0.18	0.34
multilingual-e5-large-instruct (Wang et al., 2024a)	560M	1024	0.24	0.43	0.55	0.78	0.29	0.40	0.55	0.69	0.09	0.34
Qwen3 (Yang et al., 2025)	8B	4096	0.53	0.59	0.84	0.88	0.42	0.47	0.75	0.79	0.38	0.45
KaLM-Gemma3-12B-2511 (Zhao et al., 2025)	12B	3840	0.44	0.56	0.77	0.83	0.41	0.46	0.75	0.78	0.34	0.47

Table 7: Retrieval performance across different embedding models on the **DP-ConvFinQA** dataset.

Embedding Model	Model Params.	Embed Dims.	Tables				Text				Data Product	
			Recall@20		Recall@100		Recall@20		Recall@100		Full Recall@100	
			Dense	Hybrid	Dense	Hybrid	Dense	Hybrid	Dense	Hybrid	Dense	Hybrid
all-mpnet-base-v2 (Song et al., 2020)	110M	768	0.81	0.92	0.94	0.97	0.82	0.91	0.92	0.98	0.85	0.94
granite-125m-english (et al., 2025)	125M	768	-	-	-	-	-	-	-	-	-	-
multilingual-e5-large-instruct (Wang et al., 2024a)	560M	1024	0.88	0.92	0.95	0.97	0.86	0.92	0.95	0.98	0.89	0.93
Qwen3 (Yang et al., 2025)	8B	4096	0.93	0.94	0.98	0.98	0.91	0.94	0.97	0.98	0.93	0.93
KaLM-Gemma3-12B-2511 (Zhao et al., 2025)	12B	3840	0.93	0.95	0.99	0.99	0.92	0.94	0.98	0.99	0.95	0.96

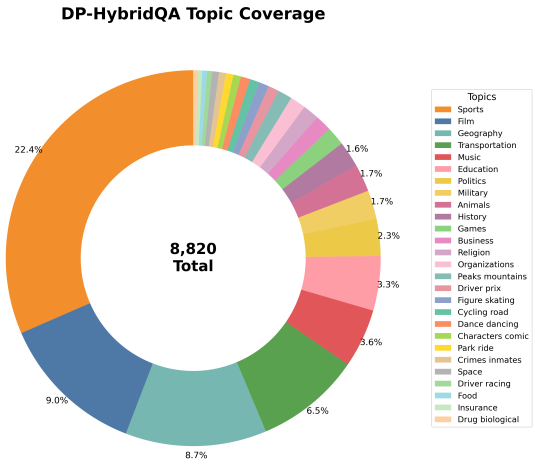


Figure 6: Topic distribution of DP-HybridQA.

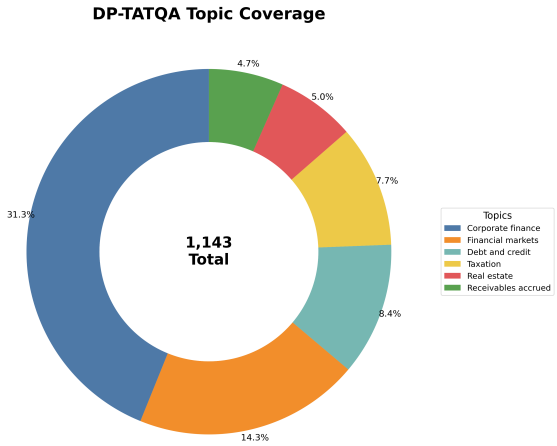


Figure 7: Topic distribution of DP-TATQA.

the source datasets' domain structure while reorganizing it into product-centric clusters.

G Artifact Use and Licensing.

DPDisc is constructed from publicly available table-text QA datasets and follows the access conditions of the original source datasets. The released benchmark, code, and evaluation scripts are intended for research and reproducibility.

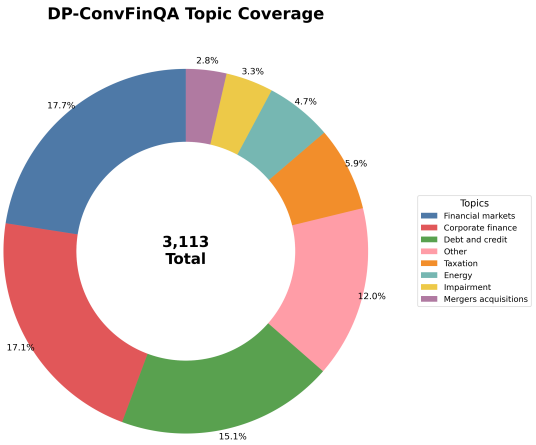


Figure 8: Topic distribution of DP-ConvFinQA.