

## A Additional Details on Adequacy Assessment

### A.1 Adequacy versus Relevance: Detailed Discussion

Our concept of adequacy is designed to complement, not replace, the established notion of relevance. A document can be highly relevant yet functionally inadequate for fulfilling a user’s underlying goal, especially in systems like Retrieval-Augmented Generation (RAG).

#### A.1.1 Illustrative Examples

##### Example 1: Analytical Tasks (From Data to Insight)

*Query:* "Explain the key financial risks associated with investing in emerging market equities for a portfolio manager."

*Document A (High Relevance, Moderate Adequacy):* A quantitative assessment for a portfolio manager shows key financial risks in emerging market equities are currency fluctuation ( $\sigma_{FX} = 12\%$ ), political instability ( $\beta_{pol} = 1.8$ ), and liquidity ( $L_{ratio} = 0.3$ ), with a 5% monthly Value at Risk of 15%.

*Document B (High Relevance, High Adequacy):* A guide for a portfolio manager explaining key financial risks: 1) Currency Risk: How a volatile exchange rate can diminish returns, even if stock prices rise. Example: A 10% depreciation can offset stock gains. 2) Political Risk: How unstable governance can lead to sudden policy shifts, citing the 2018 Argentine peso crisis as an example.

Both documents are highly relevant to the query. However, Document A presents information as raw data, requiring significant interpretation to form an explanation. Document B is structured for explanation, providing the definitions and context that directly facilitate the generation of a high-quality answer.

##### Example 2: Operational Tasks (From Information to Action)

*Query:* "Provide a guide on how to set up a secure private Docker registry using TLS for a DevOps engineer."

*Document A (High Relevance, Moderate Adequacy):* For a secure private Docker registry, a DevOps engineer must edit the `config.yml` to set the `tls.certificate` and `tls.key` paths. This is a primary step for enabling TLS. After saving the changes, the registry daemon must be restarted for them to take effect.

*Document B (High Relevance, High Adequacy):* This guide provides a step-by-step setup for a secure private Docker registry: Step 1) Generate TLS Certificates: Run the `openssl` command. Step 2) Configure Registry: Point to your generated files in `config.yml`. Step 3) Restart: Safely restart the Docker container with `docker-compose restart`.

Again, both documents are relevant. But Document A is a descriptive reference that forces the user to infer the correct sequence of actions. Document B’s explicitly structured, step-by-step format provides an unambiguous and complete process, making it far more adequate for fulfilling the user’s request for a "guide."

#### A.1.2 Conceptual Distinctions

Table 7 summarizes the key distinctions between adequacy, traditional relevance, and static authority metrics.

### A.2 Rationale for Adequacy Assessment Framework Design

#### A.2.1 Four-Dimensional Framework

Our adequacy assessment framework is designed to move beyond traditional "relevance" and evaluate a document’s utility as context for a generative model. The four dimensions—**Verifiability**, **Need Coverage**, **Evidence Completeness**, and **Structure Suitability**—were systematically developed based on an analysis of failure modes in RAG systems and a synthesis of principles from established research:

Table 7: Adequacy versus Relevance and Static Authority.

Feature	Relevance	Static Authority	Adequacy (Our Work)
Primary Goal	Measure topical alignment between query and document	Measure query-independent, global importance of a document	Measure a document’s functional utility in fulfilling the user’s underlying task
Core Question	Is this document about what the user asked?	Is this document generally authoritative or trustworthy?	Does this document provide the necessary reliable, complete, and well-structured information to achieve the user’s goal?
Key Signals	Lexical & Semantic: Keyword importance (e.g., TF-IDF, BM25), semantic similarity	External Graph & User Behavior: Link graphs, click-through rates, domain reputation	Intrinsic Content & Structure: Verifiability, Need Coverage, Evidence Completeness, Structure Suitability
Judgment Basis	Topical Match: Assesses how well the document’s subject matter aligns with the query’s topic	Global Importance: Assesses the document’s authority within the broader corpus	Task Fulfillment: Assesses how well the document’s content and presentation enable the completion of the query’s implicit task
Applicability	Universal in information retrieval systems	Primarily in large, interconnected corpora like the web	Especially critical for advanced systems like RAG

- **Verifiability:** Inspired by fact-checking and information reliability literature [30]. An unverifiable document, even if topically relevant, poses a high risk of causing the LLM to hallucinate or generate misinformation.
- **Need Coverage & Evidence Completeness:** Adapted from concepts in question answering and argumentation mining [26, 33]. A document might be relevant but incomplete, addressing only a fraction of the user’s query or presenting claims without sufficient supporting arguments.
- **Structure Suitability:** Addresses a practical but critical aspect of data quality and its impact on LLM behavior. Poorly structured documents can actively introduce misinformation (e.g., erroneous OCR), and LLMs are susceptible to the style and quality of their context.

### A.2.2 Six-Bin Structure Rationale

The six-bin structure with non-uniform score ranges was designed to provide both interpretable categories and operational utility for RAG systems:

- **[0.90, 1.00] & [0.75, 0.90):** This high-end separation distinguishes "perfect" context from "excellent" context. Empirically, documents providing fully comprehensive and perfectly structured answers are rare, justifying a narrow top bin.
- **[0.50, 0.75) & [0.25, 0.50):** The core of the distribution lies here. The 0.5 threshold is a critical pivot point, separating documents that are fundamentally useful from those that are only peripherally relevant.
- **[0.10, 0.25) & [0.00, 0.10):** This low-end separation helps distinguish documents with a faint, almost useless signal from those that are completely off-topic. The 0.10 threshold acts as a "hard filter" to discard content that offers no value.

The fine-grained splits at the extremes serve a key purpose in training. By isolating the absolute best and worst cases, we provide the model with clearer, more definitive guidance signals at the poles of the adequacy spectrum. The Irrelevance bin ([0.00, 0.10)) groups completely unrelated samples into a narrow, low-value range, reducing the model’s sensitivity to minute differences between them and encouraging it to allocate capacity toward distinguishing documents with varying degrees of utility in the higher adequacy ranges.

### A.3 LLM-Based Annotation Details

For the multi-model annotation framework, we employed seven state-of-the-art large language models with diverse architectures and training regimes: GPT-4o-mini, DeepSeek-v3, GLM-4-Flash, Gemini-1.5-Flash, Qwen2.5-72B, Llama-3.3-70B, and Claude-3-Haiku. Each model was instructed

to evaluate query-document pairs using our four-dimensional adequacy framework and assign scores according to the semantic bin definitions.

The prompt engineering for LLM-based scoring included the following key components:

- **Detailed role definition:** Models were instructed to assume the role of a professional information adequacy assessor with expertise in evaluating document utility for knowledge-intensive applications.
- **Comprehensive assessment criteria:** Each dimension (verifiability, need coverage, evidence completeness, structure suitability) was defined with specific guidelines for different quality levels.
- **Boundary case examples:** To enhance consistency, particularly at bin boundaries, we provided examples demonstrating threshold cases between adjacent bins.
- **Reasoning requirement:** Models were instructed to provide explicit reasoning for their assessments, encouraging transparency and enabling quality control.

#### A.4 Technical Details of the Combinatorial Validation Approach

The combinatorial validation approach in Algorithm 1 offers several advantages over traditional agreement-based methods:

- **Robustness to outliers:** By examining all possible combinations of four scores, the algorithm can identify consistent subsets even when individual models produce outlier assessments.
- **Efficiency through progressive validation:** The algorithm begins with an initial set of four models and only incorporates additional evaluations when necessary, optimizing computational resources while maintaining assessment quality.
- **Flexibility in consistency definition:** Rather than requiring global agreement, the algorithm accepts local consistency among subsets of models, recognizing that different models may have varying areas of expertise or evaluation tendencies.

The function  $\text{Combinations}(\mathcal{S}, 4)$  generates all possible subsets of size 4 from the current set of scores  $\mathcal{S}$ . For a set of 8 models, this results in  $\binom{8}{4} = 70$  different combinations to evaluate, providing a comprehensive search space for finding consistent assessments.

The tolerance threshold of 0.2 was empirically chosen to balance dataset quality and scale. Our semantic adequacy bins have widths ranging from 0.15 to 0.25 (see Table 1). A tolerance of 0.2 ensures that scores from the four LLMs in an accepted group are highly likely to fall within the same semantic bin, preventing major disagreements from polluting the label. We found that a stricter tolerance (e.g.,  $< 0.1$ ) significantly reduced the dataset size by rejecting many valid but nuanced cases, while a looser tolerance (e.g.,  $> 0.3$ ) introduced noticeable noise from inter-bin disagreements that harmed model performance.

#### A.5 Details on Bin-Aware Weighted Loss Function

The loss weight  $w_i$  for each sample in the bin-aware weighted loss function is computed according to:

$$w_i = \sigma(\alpha \cdot \text{overflow}_i) \quad (8)$$

The  $\text{overflow}_i$  term quantifies the magnitude by which the predicted score  $s_{pred,i}$  violates the boundaries of the bin containing the true score  $\text{Bin}(s_{true,i})$ :

$$\text{overflow}_i = \max(l_i - s_{pred,i}, 0) + \max(s_{pred,i} - h_i, 0) \quad (9)$$

where  $l_i$  and  $h_i$  represent the lower and upper bounds of  $\text{Bin}(s_{true,i})$ . The sigmoid function  $\sigma$  with scaling factor  $\alpha$  transforms the overflow magnitude into a weight value, approaching 1.0 for significant boundary violations while maintaining a base weight of 0.5 for predictions that remain within the appropriate bin.

The score distribution reveals that most samples concentrate within moderate adequacy intervals  $[0.25, 0.75]$ , accounting for nearly 46% of the corpus. This distribution reflects the typical adequacy

profile of documents retrieved via vector-based retrieval in RAG systems, rather than generic document relevance. Highly adequate documents (scores  $> 0.90$ ) are rare (0.34%), while completely irrelevant documents (below 0.10) remain infrequent (11.23%). The notable presence of borderline cases in the  $[0.10, 0.25)$  band (19.17%) underscores the need for models that finely distinguish varying marginal relevance levels.

## B Experimental

### B.1 Hardware and Software Configuration

All experiments were conducted using an NVIDIA RTX 4090 GPU with 24GB VRAM, 12 vCPU Xeon 8352V processor, and 90GB system memory running Ubuntu 22.04 LTS, with PyTorch 2.3.0 and Transformers 4.36.0. For model training, we employed the AdamW optimizer with hyperparameters  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , and weight decay=0.01.

### B.2 Adequacy Assessment Data Source

We curated a comprehensive adequacy assessment dataset by augmenting the bge-m3-data [15] comprising diverse query-document pairs with additional samples sourced from multiple heterogeneous collections: *ArguAna* [26], *CQADupstack* [27], *NFCorpus* [28], *TriviaQA* [29], *FEVER* [30], *FiQA-2018* [31], *Climate-FEVER* [32], *SciFact* [33], *TREC-COVID* [34], and *Quiz Works* [35]. The datasets used in this work are available under various licenses including CC BY-SA 4.0, CC BY-NC 2.0, CC BY-SA 3.0, Apache 2.0, CC BY 4.0, MIT, as well as specific research or public release licenses. This aggregated dataset spans a wide spectrum of domains and query intents, encompassing fact verification, specialized knowledge retrieval, and open-domain question answering. The finalized corpus includes approximately one million query-document pairs, with annotated adequacy scores binned and summarized in Table 11.

To ensure annotation quality, we adopted a multi-model scoring framework employing seven state-of-the-art large language models (LLMs) with diverse architectures and training regimes: GPT-4o-mini [36], DeepSeek-v3 [37], GLM-4-Flash [38], Gemini-1.5-Flash [39], Qwen2.5-72B [17], Llama-3.3-70B [40], and Claude-3-Haiku [41]. Each query-document pair was scored multiple times, with cross-validation and iterative selection to reduce model-specific bias and improve consistency, particularly for ambiguous boundary samples.

### B.3 Metrics Formulation

**ACC25** is a critical classification accuracy metric that measures the proportion of samples where the absolute error between predicted and true scores is no more than 0.25. This metric directly reflects the model’s precision in judging document value levels, particularly its ability to distinguish between different semantic bins. For example, it evaluates the model’s capability to differentiate between "high adequacy" and "middle adequacy," or between "marginal adequacy" and "weak relevance." ACC25 is calculated as

$$\text{ACC25} = \frac{1}{N} \sum_{i=1}^N \text{bin}(|\text{pred}_i - \text{true}_i| \leq 0.25) \quad (10)$$

where bin is an indicator function that equals 1 when the condition is satisfied and 0 otherwise.

**LACC@25** and **LACC@10** are binary classification accuracy metrics that use thresholds of 0.25 and 0.10, respectively, to divide documents into relevant and irrelevant categories, then calculate the model’s classification accuracy. These metrics are particularly crucial in Retrieval-Augmented Generation (RAG) systems, where filtering out irrelevant documents is essential to prevent noise contamination in the generation phase. Irrelevant context passages not only waste computational resources but can also lead to hallucinations or factual inconsistencies in the generated content. LACC is calculated as

$$\text{LACC@k} = \frac{\text{TP@k} + \text{TN@k}}{\text{TP@k} + \text{FP@k} + \text{TN@k} + \text{FN@k}} \quad (11)$$

where TP, TN, FP, and FN are determined based on whether predicted and true values exceed threshold  $k$ .

#### B.4 Zero-Shot Evaluation on Out-of-Domain Datasets

To validate EAreranker’s generalization capability, we conducted zero-shot evaluations on AIR-Bench [44], a challenging benchmark for real-world retrieval systems. We evaluated on `wiki_zh` and `healthcare_en` tasks following the official BM25 initial retrieval protocol. Table 8 presents the results. These results demonstrate EAreranker’s strong generalization capability across domains

Table 8: Zero-shot Evaluation on AIR-Bench (NDCG@10, %).

Model	wiki_zh	healthcare_en
bge-reranker-v2-m3 (Plaintext)	63.51	53.76
gte-reranker-base (Plaintext)	64.16	47.16
Cosine (bge-m3)	63.52	49.05
Cosine (gte-base)	61.86	47.48
EAreranker (bge-m3)	64.76	52.81
EAreranker (gte-base)	63.03	47.62

and languages. Our embedding-only approach remains competitive with plaintext rerankers while preserving the efficiency and privacy advantages.

#### B.5 Long-Document Retrieval Performance

To evaluate performance on long documents, we conducted experiments on NarrativeQA [45], a benchmark specifically designed for evaluating long-document retrieval. Table 9 presents the results. These results confirm that EAreranker effectively handles long documents while maintaining its

Table 9: Ranking Performance on NarrativeQA (NDCG@10, %).

Model	NDCG@10
Cosine (bge-m3)	48.72
EAreranker (bge-m3)	57.41

key efficiency advantage. In RAG systems, document embeddings are typically pre-computed and stored in vector databases. EAreranker leverages these existing embeddings, eliminating the need to re-process long texts for each query. Unlike plaintext rerankers that incur computational costs proportional to document length, EAreranker maintains constant inference time and memory usage regardless of the original document size.

#### B.6 Statistical Robustness Analysis

To demonstrate the statistical robustness of our results, we trained our model (EAreranker with bge-m3 embeddings) five times with different random seeds. Table 10 presents the mean and standard deviation for our primary adequacy metrics.

Table 10: Statistical Robustness Across Multiple Random Seeds (%).

Metric	Mean	Std. Dev.
ACC25	84.28	0.31
LACC@25	86.12	0.28
LACC@10	92.85	0.24

As shown, the standard deviations are very low ( $< 0.31\%$  across all metrics), confirming that our training process is robust and the reported results are reliable and representative.

## C Extended Analysis

### C.1 Privacy Threat Model and Architecture

We consider a scenario where the inquirer and the data provider are two non-trusting parties. Our method decouples the document embedding process from the reranking service. A user can convert sensitive documents to embeddings locally using a public pre-trained encoder. Only the fixed-size, anonymized embedding vectors are sent to the remote reranking service. An embedding is a lossy, one-way representation. Reconstructing long, complex, or private text from its embedding is computationally infeasible, even if an attacker possesses the same encoder model. This provides inherent privacy protection compared to plaintext-based reranking methods that require full text exposure.

### C.2 Relationship to Model Adapters and Efficiency Methods

Our work shares the goal of parameter efficiency with adapter methods [46, 47] and query-time efficiency with methods like preTTR [48]. However, our approach differs fundamentally:

**Comparison with general model adapters:** Traditional adapters are small neural modules inserted between the layers of a large pre-trained model (e.g., BERT), trained while the base model’s weights remain frozen. EAReranker, in contrast, is a standalone, post-hoc module that operates on the output of a pre-trained embedding model (i.e., the embedding vectors) and does not require access to the internal architecture of the embedding model itself. This makes our approach more flexible and modular, as it can be paired with any black-box embedding model without modification.

**Comparison with preTTR:** preTTR operates on text, pre-computing contextualized representations for each token in a document. At query time, it loads these numerous token vectors and executes the remaining transformer layers on the combined query-document text. Its computational cost is still dependent on the length of the document. EAReranker operates on embeddings, working with a single, fixed-size vector for the entire document. This fundamental design choice provides two key advantages: (1) Constant Computational Profile: As shown in Table 6 of the main text, EAReranker’s inference time and memory usage ( $\sim 550\text{MB}$ ) are constant and independent of the original document’s length. (2) Inherent Privacy: By relying solely on embedding vectors, our method never requires access to the original plaintext.

### C.3 Dataset Score Distribution

The finalized corpus includes approximately one million query-document pairs, with annotated adequacy scores binned and summarized in Table 11.

Table 11: Score Distribution of Annotated Adequacy Dataset.

Score Range	Number	Percentage
[0.90, 1.00]	3,416	0.34%
[0.75, 0.90)	234,111	23.41%
[0.50, 0.75)	252,655	25.27%
[0.25, 0.50)	205,817	20.58%
[0.10, 0.25)	191,684	19.17%
[0.00, 0.10)	112,317	11.23%

### C.4 Document Value Assessment Analysis

To validate EAReranker’s capability in distinguishing between documents with varying levels of informational value, we conducted experiments on CQADupstack and FEVER, comparing positive samples with randomly selected negative samples. Table 12 presents the results.

The document value assessment analysis reveals EAReranker’s nuanced understanding of informational adequacy in different contexts. For the CQADupstack dataset, where queries and documents represent similar or duplicate questions, the model assigns positive samples to the Weak Relevance range (0.1-0.25), with an average score of 0.1834. This scoring effectively distinguishes these samples

Table 12: Statistical Adequacy Scores on EAreranker (bge-m3).

Dataset	Positive Samples	Negative Samples
CQADupstack	0.1834	0.0876
FEVER	0.7634	0.0923

from negative samples (0.0876) while accurately reflecting their limited value for RAG systems, as duplicate questions provide minimal additional information. In contrast, the FEVER dataset, which contains documents with strong evidentiary support for factual claims, receives significantly higher adequacy scores (0.7634) for positive samples, placing them in the High Adequacy range. This substantial difference in scoring between datasets demonstrates the model’s ability to assess adequacy based on the actual informational value of documents rather than mere similarity metrics.