BULGARIAN ACADEMY OF SCIENCES
INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGIES

MIROSLAVA DONCHEVA DIMITROVA

EVALUATION FRAMEWORK OF RETRIEVAL-AUGMENTED GENERATION

SYNOPSIS OF PhD THESIS

IN "INFORMATICS" PROGRAM
PROFESSIONAL FIELD 4.6. "INFORMATICS AND COMPUTER SCIENCES"

Scientific Supervisor:

Acad. Ivan Popchev

Sofia, 2026

The dissertation thesis has been reviewed and approved for public defense at an extended meeting of the section "_____" at IICT–BAS, held on _____ 2026.

The dissertation consists of an Introduction, five chapters, a Conclusion, appendices, and a bibliography. The back matter includes Supporting Publications, a Citation Record, a Summary of Project Participation, Acknowledgements, and a Declaration of Originality of the Results.
The dissertation comprises a total of 215 pages, 24 figures, 45 tables, 34 equations, and 145 bibliographic references.

The public defense of the dissertation will take place on _____ 2026 at _____ hours in Hall _____, Block 2 of IICT–BAS, at an open session of the Scientific Jury composed of:

Scientific Jury:

1. _____
2. _____
3. _____
4. _____
5. _____

The reviews and opinions of the members of the Scientific Jury, as well as the synopsis, are published on the website of the Institute of Information and Communication Technologies – Bulgarian Academy of Sciences.

The dissertation materials are available to interested parties in Room _____ at IICT–BAS, Acad. G. Bonchev Street, building 2.

Author:
Miroslava Dimitrova

Title:
Evaluation Framework of Retrieval-Augmented Generation

INTRODUCTION

Transformer-based language models [1] scaled to billions of parameters, commonly termed Large Language Models (LLMs), have advanced, Natural Language Processing (NLP) across machine translation, summarization, question answering, and dialogue [2]. These models generate fluent text, yet their reliability is limited in settings that require factual correctness, current knowledge, or verifiable sourcing [3]. When answers must be grounded in external evidence rather than inferred from parametric knowledge, purely generative architectures frequently produce unsupported or outdated content [4]. Retrieval-Augmented Generation (RAG) addresses this limitation by separating evidence retrieval from text generation [5].

Instead of relying exclusively on parametric knowledge, RAG retrieves evidence from an external corpus and conditions generation on the retrieved context. This design enables knowledge updates through corpus refresh rather than model retraining and supports evidence-grounded responses when sources are available. However, the effectiveness of grounding depends on the retrieval policy: how relevance is determined, what context is selected, and how the similarity threshold and ranking strategy shape the evidence presented to the generator [6]. It also depends on the generator's capacity to use retrieved context consistently rather than overriding it with unsupported content. Evaluating such systems therefore requires assessing not only generation fluency but also how retrieval behavior and context selection shape factuality and completeness.

Relevance of the Topic

RAG systems are increasingly applied in domains where factual accuracy and traceability directly influence decision quality, including healthcare [7] and legal research [8]. In such settings, unsupported statements carry practical consequences: a misattributed clinical guideline or an unverifiable legal precedent can compromise downstream decisions. Retrieval failures, where relevant evidence exists but is not surfaced - can negate the intended benefits of grounding.

Although some LLMs now incorporate web browsing capabilities, such functionality does not inherently provide reproducible provenance under controlled evaluation conditions. Web sources vary in permanence, access conditions are not always documented, and the criteria by which content is selected or retained may be opaque—factors that complicate systematic comparison across experimental runs. Parametric LLMs also remain prone to producing fluent but incorrect outputs when evidence is missing, conflicting, or weakly linked to the query, and they frequently provide limited or opaque attribution for individual claims [3], [9].

Research Motivation

Although surveys describe rapid growth in RAG architectures and pipelines, they also highlight fragmentation in configurations and evaluation practices, which limits comparability and informed deployment decisions [10]. Three deficiencies motivate the research:

Deficiency 1: Threshold-aware evaluation. Similarity threshold determines whether retrieved documents are included in the generation context and directly influences retrieval precision and recall [6], [11]. In practical deployments, similarity threshold selection is not merely an implementation detail: it governs whether a system supplies insufficient context (leading to missing evidence and incomplete answers) or excessive context (introducing irrelevant passages that can distract generation and degrade factual alignment). Similarity threshold choices also affect computational cost by controlling context size and downstream processing. A threshold-aware evaluation procedure is therefore necessary to characterize retrieval selectivity systematically and to support evidence-based configuration choices. Retrieval selectivity is operationalized through similarity thresholds varied under controlled conditions, enabling systematic identification of threshold-sensitive performance patterns across datasets and models.

Note: "Similarity threshold" refers to the minimum cosine similarity score required for a retrieved passage to be included in the generation context. This parameter is also referred to as a "relevance threshold" or

"selectivity threshold" in some literature; in the following chapters "similarity threshold" or just "threshold" is used for consistency.

Deficiency 2: Reproducibility infrastructure. RAG pipelines introduce multiple interacting configuration layers: corpus preprocessing, chunking strategy, embedding model choice, index construction, retrieval settings, generation settings, and evaluation logic. Because these layers interact, independent verification becomes difficult when configurations are not captured precisely and reported in a complete and comparable form [12], [13]. When configurations are incompletely specified, results cannot be independently verified, and apparent improvements may be attributable to hidden differences rather than the intended experimental variable. This deficiency is especially acute in threshold-sensitive studies, where small changes in retrieval configuration can alter the evidence presented to the model. A reproducibility infrastructure is therefore required to record and preserve the complete run context and outputs in a form that supports independent verification and cross-study comparison.

Deficiency 3: Practical guidance for open-source deployments. Organizations that require local or on-premises deployment due to security, data governance, or cost constraints must rely on open-source LLMs and locally controlled pipelines. However, comparative evidence that links similarity threshold sensitivity in retrieval, generation quality, and deployment feasibility under consistent experimental conditions remains limited [10], [14]. Without such evidence, model selection and retriever configuration are frequently guided by informal benchmarks or mismatched assumptions about retrieval behavior across systems. Practical guidance is therefore needed to make threshold sensitivity visible and comparable across open-source deployment candidates.

Research Aim

The aim of the dissertation is to develop an evaluation framework for Retrieval-Augmented Generation that supports evidence-based retrieval configuration decisions for RAG systems with open-source LLMs, with particular focus on similarity threshold configuration.

Research Questions

Building on the identified deficiencies, three research questions are addressed:

RQ1: Does varying the similarity threshold produce measurable changes in generation quality?

RQ2: Do similarity threshold effects differ across language models?

RQ3: Do comparable similarity threshold ranges hold across knowledge domains?

Each question corresponds to a specific experimental phase: RQ1 is addressed through systematic threshold variation in Phases II–IV, RQ2 through cross-model comparison in Phases III–IV, and RQ3 through cross-domain analysis comparing agricultural and biodiversity corpora in Phase IV. Phase I provides system demonstration and runtime profiling, establishing baseline platform functionality prior to threshold experimentation.

Objectives

Four objectives are pursued:

Objective 1: Define and implement the core components of the evaluation framework by integrating three layers: **(a)** a threshold-aware evaluation procedure with composite scoring, **(b)** the Performance Assessment System for Similarity Evaluation and Retrieval (PaSSER) platform [15] providing reproducibility infrastructure with blockchain-based provenance logging, and **(c)** a controlled experimental design producing comparative threshold-aware evidence across models and domains.

Objective 2: Establish model selection criteria. Define selection criteria aligned with local deployment feasibility, licensing constraints, and computational requirements, including profiling of selected models with respect to context window size and decoding settings.

Objective 3: Define metric selection and computation procedures. Select metrics aligned with the evaluation constructs of lexical overlap, semantic similarity, fluency, accuracy, and language modeling, and

implement metric computation consistently across models and experimental conditions.

Objective 4: Conduct controlled testing and analysis. Prepare domain corpora and question-answer datasets with specified preprocessing and retrieval configurations; execute controlled evaluations under systematic parameter variation, including similarity threshold sweeps; and aggregate results to interpret outcomes with respect to retrieval selectivity, generation quality, and reproducibility, producing practical guidance for model and threshold selection.

Relationship between research questions and objectives. Research Questions RQ1-RQ3 are empirical and examine how similarity threshold configuration affects RAG performance across models and domains. Objective 1 defines the core components of the evaluation framework, including the evaluation procedure, the reproducibility infrastructure, and the controlled experimental design. Objectives 2-4 operationalize this framework through model selection, metric definition and computation, and controlled testing and analysis. Together, these objectives provide the basis for answering RQ1-RQ3 and for deriving practical guidance for model and threshold selection.

Table I.1 summarizes the mapping between the identified deficiencies, research questions, objectives, and resulting contributions.

Table I.1 *Mapping of Deficiencies, Research Questions, Objectives, and Contributions.*

Deficiency	Research Question(s)	Objective(s)	Chapter(s)	Contribution	Framework Layer
D1: Threshold-aware evaluation	RQ1, RQ2, RQ3	Obj 1 (a), Obj 3, Obj 4	Ch. 3–4	C1	Evaluation Procedure
D2: Reproducibility infrastructure	—	Obj 1 (b)	Ch. 2	C2	Infrastructure
D3: Practical guidance for open-source deployments	RQ1, RQ2, RQ3	Obj 1, Obj 2, Obj 3, Obj 4	Ch. 3–4	C3	Evidence

** Objective 1 contributes to all three deficiencies by defining the threshold-aware evaluation procedure (D1), implementing provenance logging and blockchain recording (D2), and establishing the controlled experimental design that produces comparative evidence (D3). Objectives 2-4 operationalize specific components of this framework.*

Three scientific-applied contributions together constitute the evaluation framework.

The evaluation procedure layer (C1) introduces a threshold-aware evaluation procedure incorporating Composite Performance Score, Threshold-aware Composite Performance Score, and Balance Score for characterizing retrieval selectivity across similarity threshold settings [15], [16].

The infrastructure layer (C2) implements reproducibility infrastructure through the PaSSER platform, combining blockchain-based provenance logging with complete configuration capture [15], [17].

The evidence layer (C3) produces practical guidance for open-source RAG deployments, grounded in comparative empirical evidence linking similarity threshold sensitivity, generation quality, and deployment feasibility across seven models in the 7-8 billion parameter range under controlled experimental conditions [16] [18].

Structure

The dissertation consists of an Introduction, five chapters, conclusion, appendices and bibliography.

Chapter 1 establishes the research foundations and reviews related work on RAG architectures, evaluation practices, and reproducibility challenges, positioning Deficiencies D1–D3 within the relevant literature.

Chapter 2 presents the infrastructure of the PaSSER platform. It describes the workflow, the

configuration of different settings, the automated testing process, and blockchain-based provenance logging.

Chapter 3 specifies the model selection rationale and defines the evaluation metrics and computation procedures applied in cross-model assessment.

Chapter 4 reports empirical results from controlled testing across the agriculture and biodiversity datasets, analyzing similarity threshold sensitivity, model performance and cross-domain comparison.

Chapter 5 discusses the research questions and scientific-applied contributions, addresses limitations, and outlines future research directions.

The **Conclusion** summarizes the main results.

The back matter includes Supporting Publications, a Citation Record, a Summary of Project Participation, Acknowledgements and a Declaration of originality of the results.

Note: *Tables and figures are retained only where required for understanding the summarized content. The numbering of all retained tables and figures is preserved as in the dissertation, in accordance with the synopsis requirements.*

CHAPTER 1: RETRIEVAL-AUGMENTED GENERATION

LLMs generate fluent text but remain constrained by static parametric knowledge; outputs may be ungrounded, temporally outdated, or difficult to trace. RAG mitigates these limits by incorporating external evidence at inference time. Chapter 1 positions the dissertation by tracing RAG's development from information retrieval (IR) [11] and natural language processing (NLP) [19], outlining representative RAG architectures and failure modes, reviewing evaluation approaches, and identifying research gaps aligned with Deficiencies D1–D3.

1.1 Foundational Developments

RAG reflects a long convergence of IR and NLP [20]. This section traces that convergence through four strands that shaped contemporary RAG systems: (1) basic indexing and data organization, (2) formal IR evaluation and early IR–NLP fusion, (3) advanced semantic retrieval and NLP methods, and (4) large-scale IR–NLP integration with modern embedding-based approaches [20]. Early indexing and evaluation practices established retrieval effectiveness concepts such as precision and recall [11], while semantic retrieval evolved toward embedding-based similarity matching. These developments provide the technical context for integrating retrieval with modern generative models.

1.2 The Emergence of RAG

The RAG framework was introduced in 2020 in "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks" [5]. RAG integrates retrieval into generation to mitigate limitations of purely parametric models, including reliance on static training data, limited source attribution, and hallucination [21], [22]. Architecturally, it separates a retriever and a generator: at inference time, relevant passages are retrieved from a document collection and used to condition generation, supporting improved factual accuracy, transparency, and domain adaptation without retraining. In the original formulation, retrieval uses dense representations (e.g., DPR [23] with BERT-based encoders [24]) and similarity search (e.g., MIPS [25]) with approximate nearest-neighbor methods [26], [27], while generation is implemented in a sequence-to-sequence setting (e.g., BART [28]) grounded in earlier machine translation paradigms [29]. RAG is trained end-to-end to align retrieval with generation needs. Reported evaluations covered open-domain question answering and related knowledge-intensive tasks, including Natural Questions [30], TriviaQA [31], FEVER [32], and MS MARCO NLG [33], demonstrating competitive performance with fewer parameters than purely parametric approaches [5].

1.3 RAG Innovations and Extensions

Subsequent research extended RAG to improve retrieval precision, generative accuracy, interpretability, and efficiency. A functional categorization of advancements is presented in Table 1.2, spanning seven focus areas [20]: architectural efficiency and scalability, data-centric optimization, iterative

retrieval and self-refinement, knowledge integration and multimodal extensions, domain adaptation and specialization, factual verification and grounding, and few-shot/low-resource enhancement. Many of these directions rely on selectivity controls that regulate what evidence is exposed to the generator.

Table 1.2 *Functional Categorization of Recent RAG Advancements. Reproduced from* [20].

Focus Area	Objective
Architectural Efficiency and Scalability	Reduce computational cost, improve inference speed, and support real-time applications
Data-Centric Optimization	Improve training quality through noise reduction, sampling, and data selection techniques
Iterative Retrieval and Self-Refinement	Introduce mechanisms for multi-step reasoning, response revision, or feedback-based retrieval
Knowledge Integration and Multimodal Extensions	Combine symbolic and neural retrieval to access diverse knowledge representations; extend retrieval to additional modalities
Domain Adaptation and Specialization	Enable RAG systems to perform well in specialized domains or narrow knowledge fields
Factual Verification and Grounding	Reduce hallucinations and improve transparency by anchoring outputs in verifiable sources
Few-Shot and Low-Resource Enhancement	Improve generalization with limited training data through retrieval-enhanced few-shot learning

1.3.1–1.3.7 of the dissertation detail each category; the subsection most directly relevant to the evaluation framework is summarized below.

1.3.8 Retrieval Configuration and Similarity Threshold Selection

Retrieval selectivity is commonly controlled through top-k retrieval and threshold-based filtering. Top-k returns a fixed number of the highest-scoring passages, which can include marginally relevant content when strong matches are scarce. Threshold-based filtering returns only passages whose similarity exceeds a minimum value, enabling variable context sizes and the possibility of returning no passages when evidence is weak. Deployments may combine both mechanisms, applying threshold filtering and then truncating to top-k (or applying a minimum similarity gate within top-k), creating a configuration space that directly influences retrieval precision and recall.

Across published systems, threshold-like controls are introduced through routing decisions, confidence triggers, or decision boundaries (e.g., Self-RAG [34], CRAG [35], Adaptive-RAG [36], FLARE [37]). Similarity thresholds are typically selected through local tuning or heuristics rather than systematic sensitivity characterization. A published industry case study in the banking domain showed that, under a baseline configuration with a fixed similarity threshold of 0.7, the false positive rate for some embedding models could reach 99% [38]. Published evaluations commonly report results for a single retrieval configuration and do not show how performance changes as thresholds vary. This limits evidence-driven threshold selection, confounds model comparisons across selectivity regimes, and reduces transferability across domains.

1.4 Evaluating RAG

RAG evaluation requires assessment beyond answer accuracy or lexical overlap [39], including whether retrieved evidence is relevant and whether generation remains grounded in that evidence [40], [14]. Approaches such as RAGAS [40], RGB [14], and the TREC 2024 RAG Track [41] provide metric suites and shared infrastructure, while instrumentation-oriented tooling such as TruLens [42] and tracing/observability platforms (e.g., LangSmith [43], [44] and Arize Phoenix [45]) support run-level diagnostics and regression-style comparisons. Testing harnesses such as DeepEval [46] integrate automated evaluation into development

workflows.

1.4.1–1.4.5 of the dissertation analyze each approach in detail; the resulting research gaps are summarized below.

1.4.6 Research Gaps in RAG Evaluation

Across the reviewed evaluation approaches, three gaps are observed.

First, evaluation is commonly reported under fixed retrieval configurations, without systematic characterization across similarity threshold ranges.

Second, reproducibility support often depends on incomplete configuration capture, and trace logs do not inherently provide tamper-evident provenance linking datasets, configurations, intermediate artifacts, and outputs.

Third, comparative evidence is frequently reported relative to proprietary baselines, and guidance for open-source deployments under practical constraints remains limited or fragmented.

These gaps motivate the dissertation's framework (Obj.1): Gaps related to fixed retrieval configurations correspond to **D1** and are addressed by the **Evaluation Procedure** layer. Gaps related to configuration capture and provenance correspond to **D2** and are addressed by the **Infrastructure** layer. Gaps related to comparative guidance under open-source constraints correspond to **D3** and are addressed by the **Evidence** layer.

1.5 Persistent Challenges and Emerging Solutions

Recurring RAG failure points include missing content, missed top-ranked documents, fragmented context, poor content extraction, inconsistent output structuring, incorrect specificity, and incomplete responses [47]. Emerging methods attempt to address these issues (e.g., Auto-RAG [48], GraphRAG [49], relevance sampling [6], FLARE [37], LightRAG [50], Self-RAG [34], Speculative RAG [51]), but their effectiveness depends on evaluation approaches capable of systematically detecting failure modes and characterizing selectivity effects.

1.6 Chapter Summary

Chapter 1 reviewed RAG background, representative innovations, and evaluation approaches, identifying retrieval selectivity, particularly similarity threshold selection, as an under-characterized factor influencing downstream generation quality. Three research gaps mapped to D1–D3 were articulated: limited threshold-aware evaluation evidence, incomplete reproducibility infrastructure beyond logging, and limited comparative guidance for open-source deployment decisions. These gaps motivate the infrastructure described in Chapter 2, the model and metric selection procedures in Chapter 3, and the controlled threshold-sweep experiments reported in Chapter 4.

CHAPTER 2. DESIGN AND ARCHITECTURE OF PaSSER

Performance Assessment System for Similarity Evaluation and Retrieval (PaSSER) is a modular, browser-based platform for configuring and evaluating RAG pipelines with open-source LLMs [15], [16], [17]. The platform integrates threshold-aware retrieval, multi-metric scoring, and blockchain-backed provenance logging into a unified workflow for controlled experimentation, addressing D2 (reproducibility infrastructure) and fulfilling the infrastructure component (b) of Obj.1. Chapter 2 summarizes PaSSER's design rationale, architecture, and the batch evaluation workflow used for the controlled experiments reported in Chapter 4.

2.1 Initial System Design

PaSSER was developed as a complementary module to the Smart Crop Production Data Exchange (SCPDx) platform [52], [53], [54]. SCPDx combines a blockchain layer, Antelope (formerly EOSIO) [55], with the InterPlanetary File System (IPFS) [56] to support secure and decentralized data management. Prior studies informed the blockchain choice, including platform suitability analysis [52] and oracle integration work [57]. Antelope was adopted to provide auditable, tamper-resistant records with an explicit permission model and low-cost transactions suitable for frequent submissions, while Anchor Wallet [58], [59] supports a client-side

signing workflow that keeps private keys outside the web application. Although SCPDx integrates IPFS for distributed content storage, PaSSER does not currently utilize IPFS; future integration is noted in Section 5.4.2. PaSSER was designed as a browser-based environment to improve platform independence and reduce local installation requirements, supporting reproducibility across heterogeneous client devices.

2.2 System Architecture

PaSSER follows a three-layer architecture comprising a web interface (single-page application, SPA), backend services, and a blockchain subsystem. The web interface supports configuration input and result visualization, while computation is delegated to backend services to keep retrieval, generation, and scoring behavior invariant across client environments. Authentication and transaction signing are handled client-side via Anchor Wallet, while the blockchain subsystem persists verifiable records after execution completes [17]. **2.2.1** of the dissertation describes the web interface in detail; the synopsis focuses on the backend pipeline and blockchain subsystem as the core enablers of reproducible evaluation.

2.2.2 Backend Services

Backend services implement PaSSER's execution pipeline, performing retrieval, language model inference, evaluation dispatch, and provenance logging. Semantic retrieval is implemented using ChromaDB [60]. Document corpora are embedded using the Ollama embedding endpoint and stored as vector collections. During ingestion, documents are segmented into overlapping text chunks; PaSSER exposes chunk size and overlap as configuration parameters. The default configuration is 1024 characters with 50 characters overlap and is used in the controlled experiments reported in Chapter 4. Chunking affects retrieval granularity and context continuity and may influence selectivity [61]; implications of chunking sensitivity are discussed in Section 5.3.2.

Retrieval modes. Two retrieval modes are implemented at runtime. Normal Mode invokes the LangChain VectorStoreRetriever [62] and returns the top-k passages ranked by cosine similarity, yielding a fixed number of passages regardless of absolute similarity values. Score Mode employs a ScoreThresholdRetriever to filter passages by a minimum cosine similarity threshold. Score Mode exposes three parameters: minSimilarityScore (minimum similarity for inclusion), maxK (upper bound on passages returned), and kIncrement (step size for iterative threshold sweeps). Normal Mode provides the fixed top-k baseline (Phase I, Section 4.1), while Score Mode supports controlled threshold sensitivity experiments (Phase II, Section 4.2).

Inference and evaluation. Text generation is executed via the Ollama API [63]. Generated outputs are evaluated against reference answers by a Python-based evaluation service using established libraries including Natural Language Toolkit (NLTK), torch, NumPy, rouge, transformers, and SciPy [17]. Formal metric definitions are provided in Chapter 3, while Chapter 4 specifies phase-specific selections and aggregation procedures (including CPS and T-CPS).

Provenance logging. Each execution records the active configuration (model identifier, retrieval parameters, decoding settings, dataset/vector store identifiers) together with timing and quality outputs. Compact summaries are submitted to the Antelope blockchain via a connector using Pyntelope [64]. Transaction signing is performed externally via the Anchor wallet, while submission is handled by the backend.

Figure 2.2 summarizes the execution pipeline (ChromaDB retrieval, Ollama inference, Python scoring, and blockchain logging).

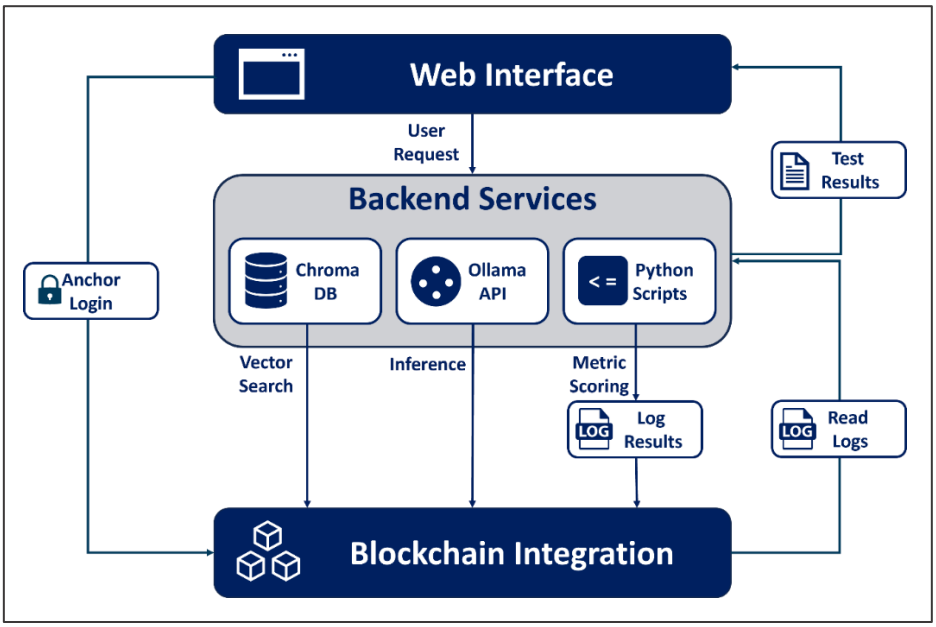

Figure 2.2 *Backend services in PaSSER. Reproduced from* [16].

2.2.3 Blockchain Integration

PaSSER employs the Antelope blockchain (formerly EOSIO) to provide immutable logging of evaluation results and associated metadata. The blockchain is not part of the retrieval or generation path; it persists verifiable records after execution completes. Authentication and transaction signing are handled via Anchor Wallet, while the backend prepares payloads and submits signed transactions without accessing private keys.

Evaluation results are stored via the *"llmtest"* smart contract through two append-only actions: *"addtest"* records accuracy-related metrics, and *"addtimetest"* records time-related indicators (e.g., model load and inference duration). Independent verification can be performed by retrieving the corresponding on-chain record and comparing persisted fields (submitting account, run identifier, timestamp, descriptive label, and numeric payload stored as float64[]) against exported run artifacts used in analysis.

Figure 2.3 shows *"addtest"* and *"addtimetest"* logging actions and the on-chain persistence boundary.

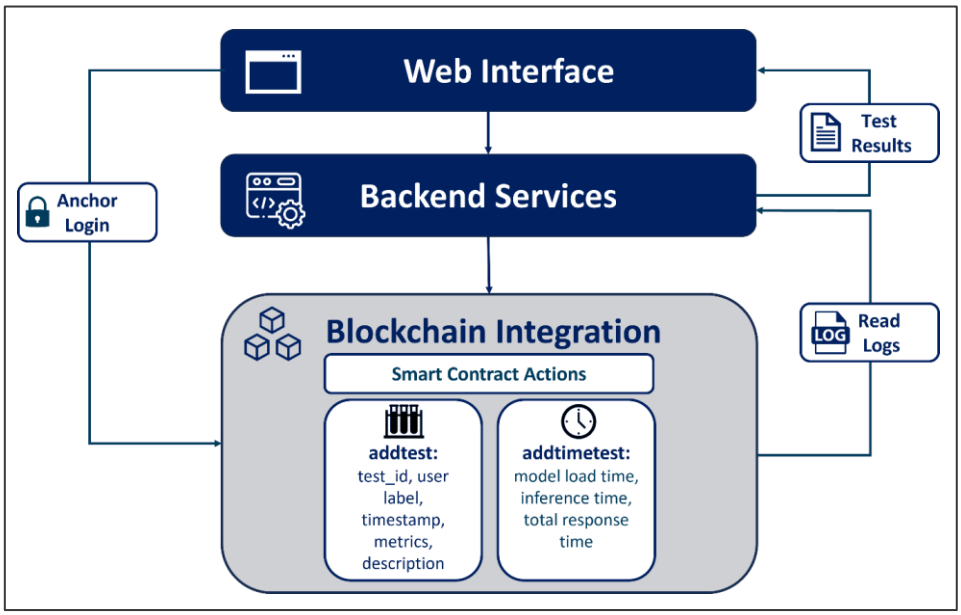

Figure 2.3 *Blockchain integration in PaSSER. Reproduced from [16].*

2.3 PaSSER App Functionalities

This section summarizes the user-facing workflow and artifacts required for controlled evaluation

runs. The platform supports both interactive exploration and batch evaluation; controlled experiments reported in Chapter 4 rely on batch evaluation.

2.3.1 System Configuration

Configuration proceeds by specifying endpoints for the inference server (Ollama API) and vector store (ChromaDB), selecting the active LLM and generation temperature, choosing a retrieval policy (Normal or Score mode), and linking data resources (vector store selection and import of a JSON dataset of question–answer pairs). Once confirmed, configuration is session-scoped and attached to outputs for traceability and reproducibility.

2.3.2 and 2.3.4 of the dissertation provide additional interface-level workflows the synopsis summarizes only the configuration, retrieval control, and batch evaluation steps used in Chapter 4.

2.3.3 Retrieval Configuration

Retrieval configuration determines how passages are selected and assembled as context. **Normal Mode** provides fixed-size *top-k* retrieval. **Score Mode** enables threshold-based filtering using a minimum cosine similarity cutoff and supports controlled similarity threshold sweeps via *minSimilarityScore*, *maxK*, and *kIncrement*. A sweep is operationalized by repeating the same evaluation workload while varying only the threshold value and holding all other parameters constant (dataset, vector store identifier, chunking configuration, and model configuration). Each run is labeled and stored as a separate evaluation record to enable direct comparison across threshold values under a fixed setup.

2.3.5 Evaluation and Testing

PaSSER supports dataset-driven batch evaluation under controlled retrieval configurations. For each dataset item, the system retrieves context, constructs an augmented prompt, generates a response, computes evaluation metrics, and records results to the blockchain via *addtest*. A complementary timing evaluation module logs latency-related indicators (e.g., model load time, inference duration, total response time) via *addtimetest*. Results can be retrieved for inspection and exported for offline analysis.

Figure 2.7 shows the evaluation logic.

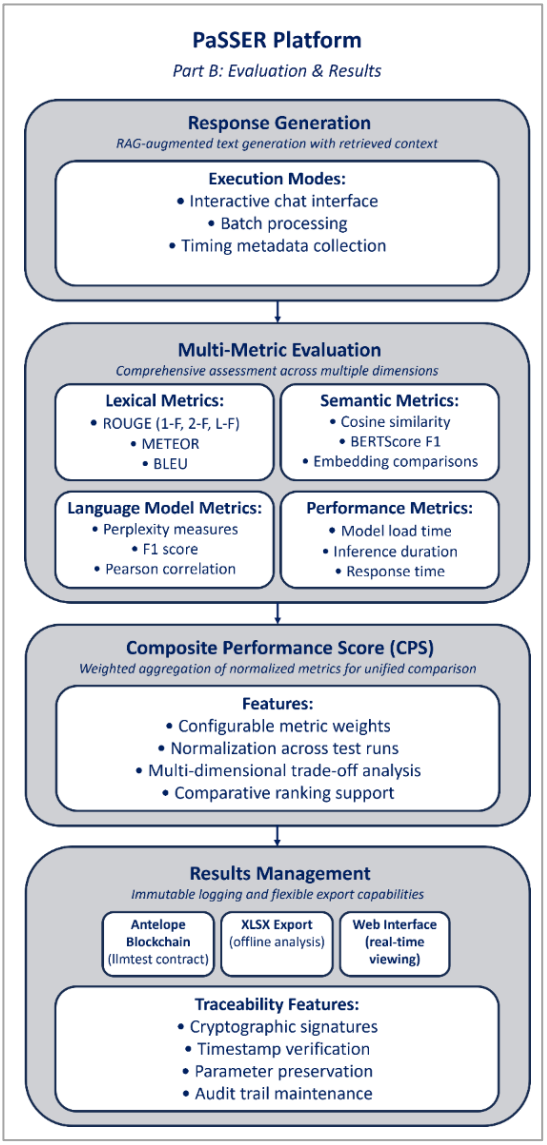

Figure 2.7 *PaSSER evaluation workflow overview. Reproduced from [16].*

2.4 Chapter Summary

Chapter 2 presented PaSSER as a reproducibility infrastructure for threshold-aware RAG evaluation [15], [16], [17]. The platform integrates backend execution services coordinating retrieval and inference with an Antelope blockchain subsystem providing tamper-evident provenance logging. PaSSER operationalizes both fixed *top-k* retrieval and threshold-gated retrieval, enabling controlled similarity threshold sweeps under constant experimental conditions. By attaching configuration metadata to each execution and persisting results as wallet-signed immutable records via *"addtest"* and *"addtimetest"*, PaSSER supports auditability and independent verification, addressing Deficiency 2 and enabling the experiments reported in Chapter 4.

CHAPTER 3. MODEL SELECTION AND EVALUATION METRICS

Chapter 3 defines the components required for the experimental analyses in Chapter 4: the open-source LLMs evaluated within PaSSER and the evaluation metrics used to assess generation quality. Sections 3.1–3.3 document the model sets, selection rationale, and integration constraints (Obj. 2). Section 3.4 specifies the evaluation metrics, including definitions, computation actions, and reporting formats. Section 3.5 defines the Composite Performance Score (CPS) and Threshold-aware CPS (T-CPS) formulations used for multi-metric aggregation (Obj. 3). Together, these components contribute to Deficiency 1 (threshold-aware scoring through CPS, T-CPS, and Balance Score) and Deficiency 3 (practical guidance for open-source deployments through model selection criteria and evaluation procedures).

3.1 Overview of Embedded LLMs and Model Selection Criteria

A representative set of open-source LLMs was integrated into PaSSER to support controlled comparisons under consistent experimental conditions. Selection prioritized open availability, feasibility of local inference on mid-range hardware, and architectural diversity within the 7B–8B parameter range. This range balances capability and accessibility and typically enables inference on 16–32 GB RAM without dedicated GPU clusters, while avoiding the requirements of larger 40B-class models. Models below 7B were excluded due to limited capacity in knowledge-intensive generation, while models above 8B were excluded due to higher hardware demands [65]. Proprietary models were also excluded because licensing, cost, and limited configuration transparency undermine strict reproducibility [66].

Two model sets were evaluated: an initial 7B set used in Phases I–II and an updated set (8B-class plus an updated Mistral) used in Phases III–IV. Architectural diversity is relevant because similarity threshold sensitivity may interact with model-specific factors such as context-window limits, attention efficiency, and instruction/reasoning tuning [5], [67], [68]. Table 3.1 summarizes key characteristics of the evaluated models.

Table 3.1 *Comparative summary of evaluated models.*

Model	Param.	Context (tokens)	Key design emphasis	Primary evaluation role	Ollama tag
Mistral 7B / v0.3	~7.3B	8,192 / 32,768	GQA; sliding-window attention	Efficiency-focused; cross-version continuity	Mistral 7B / Mistral:latest
Llama 2 7B	7B	4,096	Standard transformer; widely adopted	General-purpose baseline	Llama 2 7B
Orca 2 7B	7B	4,096	Reasoning-oriented fine-tuning	Reasoning-tuned comparator	Orca 2 7B
Granite 3.2 8B	8B	8,192	Enterprise-oriented curation	Enterprise reliability comparator	Granite3.2 8b
DeepSeek R1 8B	8B	8,192	Reasoning-focused RL training	Reasoning-focused 8B comparator	Deepseek r1:8b
Llama 3.1 8B	8B	8,192	Updated LLaMA; extended context	Current-generation baseline	llama3.1:8b

3.2 Initial Set

The initial set—Mistral 7B [69], Llama 2 7B [70], and Orca 2 7B [71]—was used to establish end-to-end functionality and profile runtime behavior under fixed top-k retrieval in Phase I [17]. In Phase II [15], the same models were evaluated under systematic threshold variation (0.50–0.80) to characterize threshold sensitivity.

3.2.1 Mistral 7B

Mistral 7B (2023) targets efficient inference in the 7B parameter range [69], [72]. It employs grouped-query attention (GQA) and sliding-window attention (SWA), which are relevant for RAG workflows where

retrieved passages increase prompt length, stress latency and memory. Reported benchmark results position Mistral 7B above Llama 2 7B on standard evaluations [73], including MMLU [74] and GSM8K [75]. Within the initial set, Mistral 7B serves as the efficiency-focused representative.

3.2.2 Llama 2 7B

Llama 2 7B (2023) is a widely used 7B-class model [70]. In the initial model set, Llama 2 7B is included as a baseline reference because it is well established and widely adopted, enabling retrieval-setting effects (top-k vs. threshold filtering) to be interpreted against a stable comparator under identical experimental conditions. Meta also released instruction-tuned variants [68], [70].

3.2.3 Orca 2 7B

Orca 2 7B (2023) is a fine-tuned derivative of Llama 2 designed to strengthen reasoning behavior through curated instruction tuning rather than architectural changes [71]. Reported improvements appear on reasoning benchmarks such as GSM8K and BIG-bench Hard [76]. Within the initial set, Orca 2 7B serves as the reasoning-focused representative.

3.3 Updated Set

The updated set—Granite 3.2 8B [77], DeepSeek R1 8B [78], Llama 3.1 8B [79], and Mistral 7B v0.3 [80]—was evaluated in Phase III (model-dependent threshold analysis) and Phase IV (cross-domain threshold analysis). Mistral was retained as an anchor model but updated to v0.3.

3.3.1 Granite 3.2 8B

Granite 3.2 8B (IBM, 2024) is an enterprise-oriented 8B instruct model designed for tasks such as question answering, summarization, extraction, coding and RAG [77]. In the updated model set, it is included as a representative oriented toward enterprise scenarios where more predictable behavior and deployment stability are typically expected.

3.3.2 DeepSeek R1 8B

DeepSeek R1 8B (2024) emphasizes reasoning-oriented training regimes and reports reinforcement learning components [81], and the released distilled variants transfer reasoning patterns from the larger DeepSeek R1 line into smaller open models [78]. Within the updated set, it serves as the reasoning-focused comparator at the 8B scale.

3.3.3 Llama 3.1 8B

Llama 3.1 8B (Meta, July 2024) is an instruction-tuned multilingual model from the Llama 3.1 family. Meta positions the 8B variant as a general-purpose assistant model and reports a 128k context length for the Llama 3.1 text-only models. [79]. Within the updated set, it functions as the general-purpose baseline at the 8B scale [82].

3.3.4 Mistral 7B (Latest Edition v0.3)

Mistral 7B v0.3 (2024) is distributed via Ollama under mistral:latest [63], reports an extended context window of 32,768 tokens and function calling support [80], and preserves the efficiency-oriented architecture described in Section 3.2.1. It is retained to maintain continuity across phases while capturing revisions within the Mistral family [69], [72].

3.4 Evaluation Metrics

Evaluating RAG requires multi-dimensional assessment because retrieval explicitly shapes generation. A multi-metric panel of twenty-four metrics was applied across lexical overlap, semantic alignment, fluency/predictability and answer quality, statistical association, and readability proxies. The selection follows the principle that diverse measures can yield more reliable aggregate assessment than any single metric [83]. Sixteen metrics were implemented in Phase I; eight additional metrics were added in Phase III. Table 3.2 enumerates the complete 24-metric panel (categories, output columns, and implementation phase).

Table 3.2 *Complete enumeration of the 24-evaluation metrics.*

Category	Metric	Output column	Description	Implementation
Lexical overlap	METEOR	METEOR	Token-level alignment with stemming and synonym matching; balances precision and recall	Phase I
	ROUGE-1	Rouge-1.r	Unigram overlap recall	Phase I
	ROUGE-1	Rouge-1.p	Unigram overlap precision	Phase I
	ROUGE-1	Rouge-1.f	Unigram overlap F1	Phase I
	ROUGE-2	Rouge-2.r	Bigram overlap recall	Phase I
	ROUGE-2	Rouge-2.p	Bigram overlap precision	Phase I
	ROUGE-2	Rouge-2.f	Bigram overlap F1	Phase I
	ROUGE-L	Rouge-l.r	Longest common subsequence recall	Phase I
	ROUGE-L	Rouge-l.p	Longest common subsequence precision	Phase I
	ROUGE-L	Rouge-l.f	Longest common subsequence F1	Phase I
	BLEU	BLEU	n-gram precision with brevity penalty	Phase I
	F1 Score	F1 Score	Token overlap F1; diagnostic for answer correctness	Phase I
Semantic similarity	Cosine similarity	Cosine similarity	Embedding-space similarity between generated and reference texts	Phase I
	BERTScore	Bert-Score.precision	Contextual token similarity precision	Phase III
	BERTScore	Bert-Score.recall	Contextual token similarity recall	Phase III
	BERTScore	Bert-Score.f1	Contextual token similarity F1	Phase III
Fluency/ Predictability	Laplace perplexity	Laplace Perplexity	Surface-level predictability under a Laplace-smoothed bigram n-gram language model	Phase I
	Lidstone perplexity	Lidstone Perplexity	Surface-level predictability under a Lidstone-smoothed trigram n-gram language model	Phase I
Statistical correlation	Pearson correlation	Pearson correlation	Linear association between generated and reference representations	Phase I
Readability proxy (B-RT)	B-RT Coherence	B-RT.coherence	Topic focus and local organization	Phase III
	B-RT Consistency	B-RT.consistency	Self-consistency across claims	Phase III
	B-RT Fluency	B-RT.fluency	Readability and grammatical flow	Phase III
	B-RT Relevance	B-RT.relevance	Alignment to query framing	Phase III
	B-RT Average	B-RT.average	Arithmetic mean of B-RT components	Phase III

Sections 3.4.1–3.4.5 provide full definitions, computation actions, and reporting formats for each metric. For brevity, **formulas (3.1) - (3.24)** are provided in the dissertation and are also presented in [17], [15] and [18]; the synopsis retains only the minimum descriptive context for Sections 3.4.1–3.4.4 and the defining formula block for B-RT in Section 3.4.5.

3.4.1 Lexical Overlap Metrics

Lexical overlap metrics compare system outputs to reference answers at the level of tokens and n-grams. METEOR [84], ROUGE [85], and BLEU [86] provide complementary indicators of surface-form alignment.

3.4.2 Semantic Similarity Metrics

Semantic similarity metrics assess meaning alignment beyond surface overlap. Cosine similarity [87] and BERTScore [88] are used to compare generated answers and references through embedding-based

representations.

3.4.3 Fluency, Predictive, and Answer Quality Metrics

Fluency and predictability indicators are computed using classical n-gram perplexity measures (NLTK) as model-independent proxies; implications are discussed in Section 5.3.3. Token-level F1 is used as an answer-quality indicator derived from overlap between generated and reference content.

3.4.4 Statistical Correlation Metrics

Statistical association is assessed using Pearson correlation to characterize linear relationships between selected metric outputs.

3.4.5 Human-Readability Inspired Metrics (B-RT)

The B-RT suite implements a Nubia inspired regression-style proxy intended to approximate human readability judgments in RAG [89], [90]. Scores are treated as automated comparative signals rather than validated human judgments. The base similarity is defined as:

$$s = \cos\big(E_{cls}(Reference),\ E_{cls}(G)\big), \quad \textbf{(3.25)}$$

and the aggregate index is:

$$B - RT.Average = \frac{Coherence+Consistency+Fluency+Relevance}{4} \quad \textbf{(3.26)}$$

3.5 Composite Performance Scores

Because individual metrics capture distinct aspects of quality, multi-metric aggregation is required for systematic comparison. Although PaSSER computes all 24 metrics, CPS aggregation uses a nine-metric subset to avoid double-weighting correlated variants. Metric families from Section 3.4 are regrouped into four evaluation constructs. Table 3.3 summarizes the mapping and aggregation logic.

Table 3.3 *Mapping of Section 3.4 Metric Families to CPS Evaluation Constructs*

Section 3.4. metric family (5 groups)	Primary purpose of the family	Mapped CPS evaluation construct (4 constructs)	Mapping rule / notes
3.4.1 Lexical overlap (METEOR, ROUGE variants, BLEU)	Measures surface-form overlap with the reference (token/phrase overlap)	Lexical overlap	Direct mapping. All lexical overlap measures belong here.
3.4.2 Semantic similarity (Cosine Similarity, BERTScore variants)	Measures meaning preservation beyond exact word overlap	Semantic similarity and alignment	Direct mapping. Embedding-based similarity measures form the core of this construct.
3.4.3 Fluency, predictive, answer quality (Laplace Perplexity, Lidstone Perplexity, F1 Score)	Captures linguistic predictability and answer correctness	Split mapping: Language modeling (perplexities) and Fluency and answer correctness (F1)	This family spans two constructs: perplexity-based metrics map to Language modeling; F1 maps to Fluency and answer correctness.
3.4.4 Statistical correlation (Pearson Correlation)	Measures linear association between paired evaluation signals	Semantic similarity and alignment	Pearson Correlation is used as an alignment indicator and grouped under Semantic similarity and alignment rather than treated as a separate aggregation construct.
3.4.5 Human-readability inspired metrics (B-RT suite) (B-RT.fluency, B-RT.relevance, B-RT.coherence, B-RT.consistency, B-RT.average)	Multi-aspect readability and quality signals not confined to one dimension	Split mapping across constructs (by component)	B-RT is a metric suite. Each component is mapped to the construct it operationalizes: relevance/average → Semantic similarity and alignment; fluency → Fluency and answer correctness; coherence/consistency → fluency/coherence signals within the same construct.

3.5.1 Composite Performance Score (CPS) Formulation

CPS aggregates normalized metric values using a weighted sum, addressing three complications in

multi-metric comparison: metrics operate on different scales, have opposing polarities, and differ in diagnostic importance [15].

For a given query *q* evaluated under model *m* at similarity threshold *t*:

$$CPSq = \sum_{i=1}^{n} w_i \times \left[d_i \frac{(m_{i,q} - min_i)}{(max_i - min_i)} + \frac{(1 - d_i)}{2} \right] \quad \textbf{(3.27)}$$

where $m\_(i,q)$ is the raw metric value for metric i on query q, min_i and max_i are observed extremes across the evaluation set $d\_i \in \{-1, +1\}$ is the polarity indicator ($d\_i = +1$ if higher is better, $d\_i = -1$ if lower is better), and w_i is the metric weight($\Sigma w\_i = 1$).

For positive-polarity metrics $d\_i = +1$:

$$Normalized\ value = \frac{m_{i,q} - min_i}{max_i - min_i} \quad \textbf{(3.28)}$$

For negative-polarity metrics $d\_i = -1$, normalization is inverted:

$$Normalized\ value = \frac{max_i - m_{i,q}}{max_i - min_i} \quad \textbf{(3.29)}$$

This ensures all normalized values fall within [0, 1] and higher values consistently indicate better performance regardless of original polarity.

Mean CPS for model m at similarity threshold t across Q queries:

$$\mu_{m,t} = \frac{1}{Q} \sum_{q=1}^{Q} CPS_q^{(m,t)} \quad \textbf{(3.30)}$$

3.5.2 Threshold-Aware Composite Performance Score (T-CPS)

CPS captures mean performance but not consistency across queries. A configuration with high mean CPS but large variance may be less reliable than one with slightly lower mean but stable outputs. T-CPS incorporates a reward-penalty structure based on the coefficient of variation (CV) [91].

$$CV_{m,t} = \frac{\sigma_{m,t}}{\mu_{m,t}} \quad \textbf{(3.31)}$$

Accordingly, the formulation rewards stable performance while penalizing higher variability:

$$T - CPS = \mu \times (1 + \alpha \times (1 - CV)) - \beta \times CV^2 \quad \textbf{(3.32)}$$

The reward term $(1 + \alpha \times (1 - CV))$ increases scores for low-variability configurations. The penalty term $\beta \times CV^2$ applies quadratic reduction as CV increases. Parameters $\alpha = 0.1$ and $\beta = 0.05$ enforce a 2:1 asymmetry: the consistency reward contributes up to approximately +10% when variability is low, while the penalty reduces scores by up to approximately −5% under the maximum observed CV. These values are not claimed as optimal; they represent starting points informed by CPS variation observed in preliminary runs.

Sensitivity analysis across 25 parameter combinations ($\alpha \in \{0.05, 0.10, 0.15, 0.20, 0.25\} \times \beta \in \{0.025, 0.05, 0.075, 0.10, 0.15\}$) confirmed ranking stability: 29 of 31 configurations (93.5%) showed no rank change across all combinations. α explained 99.87% of T-CPS variance, confirming that the consistency reward is the primary driver while the variability penalty plays a secondary moderating role [18].

3.5.3 Statistical Significance Testing

To determine whether observed differences between threshold configurations and baseline are statistically meaningful, paired two-tailed t-tests compare per-question CPS distributions between baseline and each threshold configuration ($\alpha = 0.05$). Pairing is defined at the question level: the same question is evaluated under both conditions, controlling for query-specific variation. Significance is reported using conventional notation: * for $p < 0.05$, ** for $p < 0.01$, *** for $p < 0.001$.

Effect size is assessed using Cohen's d for paired samples:

$$d = \frac{M_{diff}}{SD_{diff}} \quad \textbf{(3.33)}$$

where M_diff is the mean of per-question CPS differences (threshold minus baseline) and SD_diff is the standard deviation of those differences. Effect sizes follow standard conventions [92]: negligible ($d < 0.2$), small ($0.2 \leq d < 0.5$), medium ($0.5 \leq d < 0.8$), or large ($d \geq 0.8$).

Reported p-values are uncorrected. Each phase includes 40 paired comparisons (4 models × 10 thresholds); at $\alpha = 0.05$, approximately 2 significant results per phase would be expected by chance under

the null hypothesis. Significance is therefore interpreted at the pattern level — consistency across thresholds within models, coherence across phases, and alignment with effect sizes — rather than as definitive evidence from any single comparison (see Section 5.3.3).

3.5.4 Balance Score (Stability–Performance Ratio)

While T-CPS incorporates stability directly into the composite score, deployment decisions often require an explicit criterion expressing the trade-off between improvement magnitude and output variability. Balance Score quantifies how much stability-adjusted improvement is obtained per unit of variability [18]:

$$Balance\ Score_{m,t} = \frac{(T\text{-}CPS improvement\ \%_{m,t}/100)}{CV_{m,t}} \quad \textbf{(3.34)}$$

Higher Balance Scores indicate configurations providing larger stability-adjusted gains per unit of variability. Configurations with large improvements but high variability produce lower Balance Scores than configurations with moderate improvement and low variability, reflecting a preference for predictable behavior when selecting retrieval parameters for deployment.

3.6 Chapter Summary

Chapter 3 defined the model set and evaluation metrics used in Chapter 4. Seven open-weight models spanning the 7B–8B range were selected under constraints of open availability and feasibility of local inference on mid-range hardware [65], [66]. Evaluation employed a 24-metric panel across lexical overlap, semantic similarity, fluency/predictability and answer quality, statistical correlation, and readability proxies. Multi-metric aggregation was defined through CPS (3.27)–(3.30), extended with stability-aware T-CPS (3.31)–(3.32) and Balance Score (3.34), with statistical evaluation procedures and effect size reporting (3.33) supporting comparative interpretation. These components support Deficiency 1 through threshold-aware composite scoring and Deficiency 3 through reproducible model selection criteria and evaluation procedures applied in Chapter 4.

CHAPTER 4: EXPERIMENTAL EVALUATION AND RESULTS

4.1 Phase I: System Testing and Runtime Profiling

Phase I validates end-to-end platform functionality and profiles baseline behavior by executing the complete RAG pipeline for three 7B-parameter models (Mistral 7B, Llama 2 7B, Orca 2 7B) across two hardware environments [17]. Retrieval operated in Normal Mode (fixed top-k, K = 100) with generation temperature 0.2. This phase supports verification of the infrastructure component (b) of Obj.1; threshold-aware evaluation begins in Phase II.

4.1.1 Experimental Design

The agricultural corpus was constructed from Regulation (EU) 2018/848 on organic production [93] and the FAO Climate-Smart Agriculture Sourcebook [94]. Documents were preprocessed, segmented using the parameters in Section 2.2.2 (1024 characters, overlap 50), embedded using Mistral 7B, and stored in ChromaDB. The evaluation dataset comprises 446 question–answer pairs; questions were generated by prompting Mistral 7B, introducing a fairness consideration acknowledged in Section 5.3.2 and mitigated in Phase IV through use of Claude Opus. Two hardware environments were used: Apple Mac Mini M1 (macOS, 16 GB RAM, GPU-accelerated) and Intel Xeon Ubuntu Server (128 GB RAM, CPU-only). Two test procedures were executed: a RAG Q&A Score Test (16 metrics) and a Timing Performance Test. Outputs were logged on-chain via smart contract actions (Section 2.2.3).

4.1.2–4.1.3 Timing and Quality Results.

The Mac M1 environment achieved approximately 2.2 times higher throughput than the CPU-only Ubuntu configuration (Table 4.1 of the dissertation), while quality metric values remained comparable across the two environments. The small difference in load duration indicates that the main performance advantage is related to inference speed rather than model initialization. Under a fixed retrieval configuration, Mistral 7B showed the strongest overall results on most lexical and semantic alignment metrics, Orca 2 7B achieved the highest ROUGE precision values, and Llama 2 7B reported the lowest perplexity. These results indicate that the relative ranking of the models depends on the metric family considered. The Phase I results are descriptive and establish the baseline for the later threshold-sensitive analyses.

4.1.4–4.1.5 Analysis and System Verification

End-to-end system verification confirmed the correct operation of the full workflow - from data ingestion and retrieval, through the generation of 1,338 answers and the computation of 16 metrics, to spreadsheet export and blockchain logging. The comparison across the two environments supports the interpretation that quality outcomes depend primarily on the model and retrieval configuration rather than on the computing environment.

4.1.6 Phase I Summary

Phase I establishes PaSSER as a functional and reproducible evaluation environment. With system correctness established, Phase II introduces similarity threshold filtering.

4.2 Phase II: Similarity Threshold and CPS

Phase II served as a pilot study to examine how similarity threshold influences generation quality under a constrained Score Mode configuration and to evaluate CPS aggregation before the broader analyses in Phases III-IV [15]. The experiments used three 7B-parameter LLMs - Mistral 7B, Llama 2 7B, and Orca 2 7B - with K = 100, K-Inc = 2, and temperature fixed at 0.2. The similarity threshold was varied from 0.50 to 0.80 in steps of 0.05, producing 2,121 evaluations (3 models × 7 thresholds × 101 questions) on an Apple Mac mini M1. This range was selected to capture the transition from more permissive to more selective retrieval while avoiding the sparsity conditions examined later in Phases III-IV. Phase II addresses RQ1 within a pilot setting, applying components (a) and (c) of Obj. 1 and contributing to Obj.4.

4.2.1 Experimental Design

Phase II used a subset of 101 question-answer pairs from the Phase I dataset, while preserving the same chunking, embedding, and vector store settings. The main design change was the introduction of Score Mode retrieval, which enabled threshold-based filtering of retrieved passages. Per-question exports were retained only for thresholds 0.50-0.80, and all Phase II analyses were therefore restricted to this interval.

4.2.2 CPS Weighting Scheme

CPS was applied as a weighted aggregation of nine evaluation metrics covering four construct families: lexical overlap (METEOR, BLEU, ROUGE-1 F, ROUGE-L F), semantic similarity and alignment (Cosine Similarity, Pearson Correlation), fluency and correctness (F1 Score), and language modeling (Laplace Perplexity, Lidstone Perplexity). Perplexity metrics are negative-polarity and inverted after normalization (Section 3.5.1). Table 4.3 specifies the metric panel and weights.

Table 4.3 *CPS Metric Panel and Weighting Scheme (Phase II). Reproduced from [15].*

Metric	Weight	Rationale
METEOR	0.15	Overall assessment of text quality
ROUGE-1 F-score	0.075	Different levels of text similarity overlap
ROUGE-L F-score	0.075	Different levels of text similarity overlap
BLEU	0.15	Overall assessment of text quality
Laplace Perplexity	0.075	Predicts performance and accuracy (lower is better)
Lidstone Perplexity	0.075	Predicts performance and accuracy (lower is better)
Cosine Similarity	0.10	Measures relevance and retrieval correlation
Pearson Correlation	0.10	Measures relevance and retrieval correlation
F1 Score	0.20	Most comprehensive and impactful metric
TOTAL	1.00	

4.2.3 Results

Phase II results are reported descriptively, without formal statistical significance testing because this phase serves as a pilot study. Within the retained threshold range, the highest CPS was observed at 0.55 for Mistral 7B and Llama 2 7B, and at 0.65 for Orca 2 7B. Across the sweep, Orca 2 7B showed the most stable CPS profile, whereas Mistral 7B and Llama 2 7B displayed greater threshold sensitivity, including model-specific declines at intermediate values. Figure 4.1 illustrates these CPS trends across thresholds for all three models.

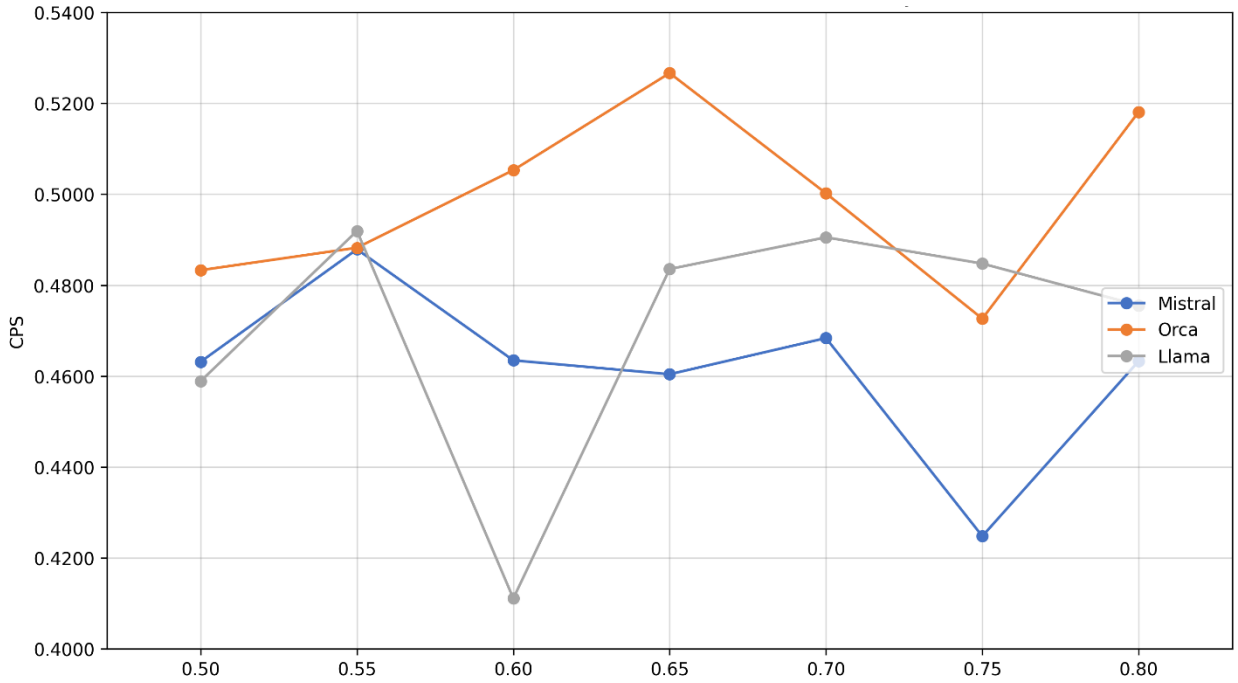

Figure 4.1 *CPS values for the three models across similarity threshold values* ***0.50–0.80*** *(Phase II pilot, Score Mode). Adapted from [15].*

4.2.4–4.2.5 Analysis and Summary

The Phase II pilot indicates that the threshold associated with the highest CPS is model-dependent within the tested configuration. A lower or moderate threshold may remain effective for models that tolerate weakly related context more successfully, whereas models more sensitive to retrieval noise may benefit from stricter filtering. These interpretations are explanatory rather than causal, since Phase II does not isolate the mechanisms linking model characteristics to threshold behavior. The findings are also bounded by the pilot conditions: a single domain, a 101-question subset, fixed preprocessing settings, and retained artifacts only for thresholds 0.50-0.80. Even with these limits, Phase II provides initial evidence for RQ1 and establishes the basis for the broader threshold analyses in Phases III-IV.

4.3 Phase III: Model-Dependent Similarity Thresholds

Phase III investigates how open-source LLM performance changes as similarity threshold varies in a RAG pipeline, with emphasis on model-dependent sensitivity under progressively more selective retrieval. The model set was revised from Phase II: Mistral 7B was retained for continuity in version 0.3, while Llama 2 7B and Orca 2 7B were replaced by three 8B-class models - DeepSeek R1 8B, Llama 3.1 8B, and Granite 3.2 8B. Similarity thresholds from 0.50 to 0.95 were evaluated in steps of 0.05. Generation quality was aggregated using CPS and T-CPS, with the full 24-metric set computed for each run. Baseline refers to Normal Mode (top-k retrieval without thresholding), while thresholded runs used Score Mode. Phase III addresses RQ1 and RQ2, applying components (a) and (c) of Obj.1 and contributing to Obj. 4.

4.3.1 Experimental Design

A subset of 369 question-answer pairs from the Phase I dataset was used, while chunking, embedding, and vector store settings were kept unchanged. Experiments were executed across three hardware environments - M1 Mac Mini, M2 Mac Mini, and a CPU-only server. Context buffer sizes differed by model (2,048-10,000 tokens) because of memory constraints; the implications are discussed in Section 5.3.2. To limit hardware-related confounding, statistical comparisons were conducted within each model against its own baseline. In total, 16,236 evaluations were executed (4 models × 11 configurations × 369 question-answer pairs).

4.3.2 CPS Weighting Scheme

Phase III applies CPS and T-CPS using an updated metric panel with broader semantic and fluency coverage: 30% lexical overlap (METEOR, ROUGE), 25% semantic similarity (BERTScore.f1, B-RT.average), 25%

fluency and accuracy (F1 Score, B-RT.fluency), and 20% language modeling (Laplace/Lidstone Perplexity). Table 4.6 summarizes the transition from the Phase II panel to the expanded formulation used in Phases III-IV.

Table 4.6 *Evolution of CPS Metric Panel Across Experimental Phases.*

Category	Metric	Phase II	Phase III-IV	Change	Rationale
Lexical Overlap	METEOR	0.150	0.150	—	Core text quality metric retained
	BLEU	0.150	—	Removed	Redundant with METEOR; brevity penalty less relevant for QA
	ROUGE-1.f	0.075	—	Replaced	Unigram overlap less informative than bigram
	ROUGE-2.f	—	0.075	Added	Bigram overlap captures phrase-level similarity
	ROUGE-L.f	0.075	0.075	—	Longest common subsequence retained
	Subtotal	**0.450**	**0.300**	**−0.150**	
Semantic Similarity	Cosine Similarity	0.100	—	Removed	Replaced by contextual embeddings
	Pearson Correlation	0.100	—	Removed	Statistical measure less interpretable
	BERTScore.f1	—	0.125	Added	Contextual token similarity
	B-RT.average	—	0.125	Added	Multi-dimensional readability proxy
	Subtotal	**0.200**	**0.250**	**+0.050**	
Fluency & Accuracy	F1 Score	0.200	0.150	−0.050	Weight redistributed to semantic metrics
	B-RT.fluency	—	0.100	Added	Explicit fluency assessment
	Subtotal	**0.200**	**0.250**	**+0.050**	
Language Modeling	Laplace Perplexity*	0.075	0.100	+0.025	Increased emphasis on text predictability
	Lidstone Perplexity*	0.075	0.100	+0.025	Increased emphasis on text predictability
	Subtotal	**0.150**	**0.200**	**+0.050**	
TOTAL		**1.000**	**1.000**		

4.3.3 CPS Performance Overview

Table 4.7 summarizes the highest CPS improvement achieved by each model. The best-performing threshold was model-dependent: Mistral 7B v0.3 showed the largest gain (+4.58% at 0.95), followed by Llama 3.1 8B (+1.58% at 0.90), Granite 3.2 8B (+1.25% at 0.95), and DeepSeek R1 8B (+1.01% at 0.90).

Table 4.7 Top CPS Improvement Configurations by Model (Top 3 Agriculture).

Model	Rank	Threshold	Mean CPS	Improvement %
Mistral 7B v0.3	1	0.95	0.5454	**4.58**
	2	0.9	0.5338	2.37
	3	0.7	0.5325	2.11
Granite 3.2 8B	1	0.95	0.5182	**1.25**
	2	0.7	0.5179	1.2
	3	0.8	0.5178	1.17
Llama 3.1 8B	1	0.9	0.508	**1.58**
	2	0.7	0.5065	1.33
	3	0.55	0.5052	1.01
DeepSeek R1 8B	1	0.9	0.4559	**1.01**
	2	0.95	0.455	0.8
	3	0.65	0.4548	0.77

4.3.4 T-CPS Performance and Stability

Table 4.8 presents the highest T-CPS configurations per model. For Mistral 7B v0.3, Granite 3.2 8B, and Llama 3.1 8B, the best T-CPS thresholds remain close to the CPS optima, indicating broad agreement between mean performance and stability-aware ranking. DeepSeek R1 8B differs from this pattern: its highest T-CPS occurs at 0.65 (+0.79%), rather than at its CPS-optimal threshold of 0.90, indicating a trade-off between performance gain and variability. This shift is consistent with its lower coefficient of variation (CV = 0.085-0.108) compared with the other models (CV = 0.122-0.148). Figure 4.2 compares threshold-wise CPS and T-CPS improvements for all models.

Table 4.8 *Top T-CPS Improvement Configurations by Model (Top 3 Agriculture).*

Model	Rank	Threshold	T-CPS	T-CPS Impr. %	CV	Interpretation
Mistral 7B v0.3	1	0.95	0.5916	4.54	0.134	Large improvement
	2	0.9	0.5793	2.36	0.131	Moderate improvement
	3	0.7	0.5783	2.17	0.128	Moderate improvement
Granite 3.2 8B	1	0.95	0.5628	1.25	0.124	Small improvement
	2	0.8	0.5625	1.2	0.122	Small improvement
	3	0.7	0.5622	1.15	0.129	Small improvement
Llama 3.1 8B	1	0.9	0.5501	1.48	0.148	Small improvement
	2	0.7	0.549	1.26	0.142	Small improvement
	3	0.55	0.5484	1.16	0.129	Small improvement
DeepSeek R1 8B	1	0.65	0.4961	0.79	0.085	Small improvement
	2	0.9	0.496	0.78	0.108	Small improvement
	3	0.95	0.4958	0.74	0.093	Small improvement

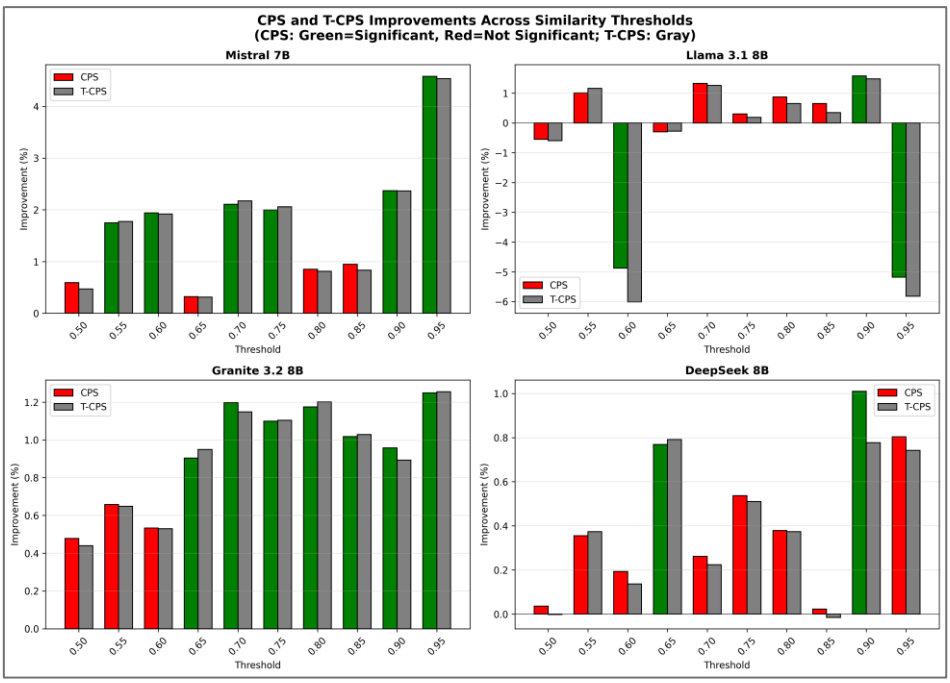

Figure 4.2 *Phase III (Agriculture, N = 369): Threshold-wise CPS and T-CPS improvements per model. CPS bars indicate percent improvement relative to baseline and are colored by significance at p < 0.05 (uncorrected). T-CPS bars (gray) indicate stability-aware improvement.*

4.3.5 Correlation Analysis

Correlation analysis was used to examine metric redundancy and to determine whether T-CPS contributes information beyond mean CPS. Figure 4.3 presents the Spearman correlation matrix for the Phase III component metrics, pooled across models and thresholds 0.50-0.95 plus baseline using per-question results.

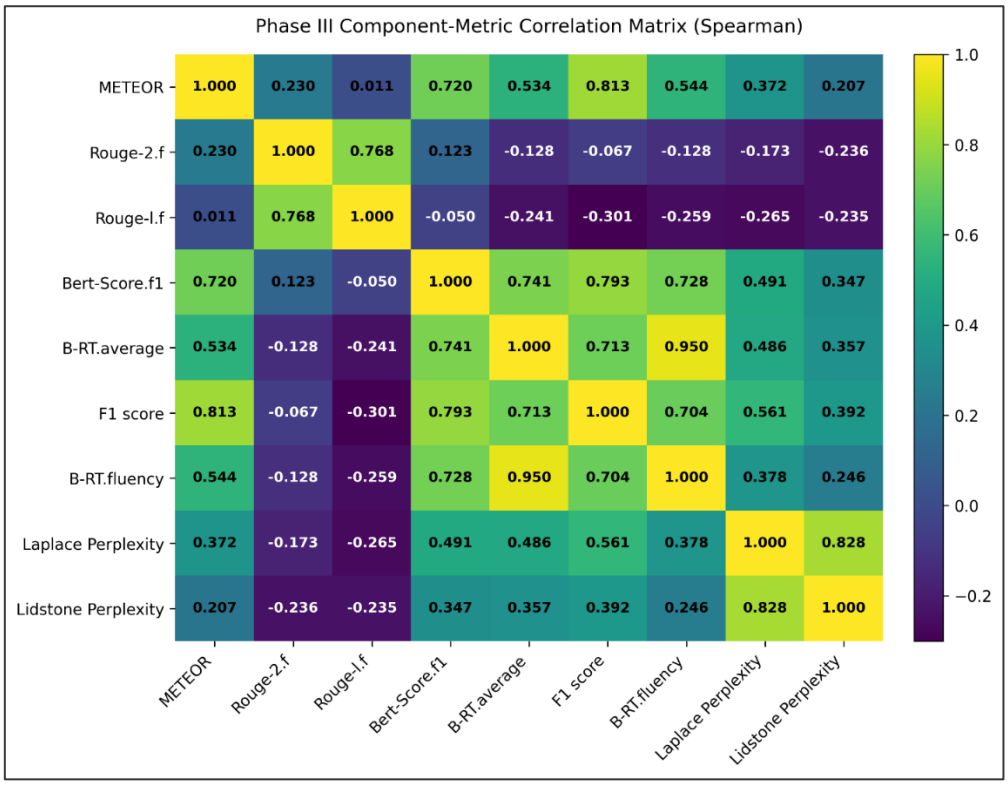

Figure 4.3 *Phase III component metric correlation matrix (Spearman), pooled across models and thresholds 0.50–0.95 plus baseline using per-question results.*

Table 4.9 shows that T-CPS is almost perfectly associated with CPS (ρ ≈ 0.999), while also maintaining a weaker positive relationship with CV (ρ ≈ 0.392). This indicates that T-CPS remains primarily driven by mean performance, while still reflecting stability as a secondary preference.

Table 4.9 *Phase III associations between stability-aware scoring T-CPS, mean CPS, and variability CV (Spearman; N = 44 model and threshold configurations).*

Relationship	Spearman ρ
ρ(T-CPS, CPS)	0.999
ρ(T-CPS, CV)	0.392

4.3.6 Balance Score

Balance Score ranks configurations by stability-adjusted improvement per unit of variability (Balance Score = (T-CPS improvement % / 100) / CV; Section 3.5.4). Table 4.10 reports the top 10 statistically significant positive configurations. Mistral 7B v0.3 dominates this ranking because of its larger T-CPS gains combined with moderate variability, whereas DeepSeek R1 8B remains competitive through smaller improvements paired with lower CV.

Table 4.10 *Balance Score ranking (top 10 statistically significant positive configurations).*

Rank	Model	Threshold	T-CPS Impr. %	CV	Balance Score	Sig.
1	Mistral 7B v0.3	0.95	+4.54	0.134	0.339	***
2	Mistral 7B v0.3	0.9	+2.36	0.131	0.18	**
3	Mistral 7B v0.3	0.7	+2.17	0.128	0.17	*
4	Mistral 7B v0.3	0.75	+2.06	0.128	0.161	*
5	Mistral 7B v0.3	0.6	+1.92	0.135	0.142	*
6	Mistral 7B v0.3	0.55	+1.77	0.132	0.134	*
7	Granite 3.2 8B	0.95	+1.25	0.124	0.101	**
8	Llama 3.1 8B	0.9	+1.48	0.148	0.1	*
9	Granite 3.2 8B	0.8	+1.2	0.122	0.098	**
10	DeepSeek R1 8B	0.65	+0.79	0.085	0.093	**

Table 4.11 summarizes the agreement among the thresholds selected by CPS, T-CPS, and Balance Score: full alignment is observed for Mistral 7B v0.3, Granite 3.2 8B, and Llama 3.1 8B, while DeepSeek R1 8B again shows divergence, with CPS favoring 0.90 and the stability-aware criteria favoring 0.65.

Table 4.11 *Threshold Alignment Across Selection Criteria by Model.*

Model	Best CPS Threshold	Best T-CPS Threshold	Best Balance Score Threshold	Alignment
Mistral 7B v0.3	0.95	0.95	0.95	Perfect
Granite 3.2 8B	0.95	0.95	0.95	Perfect
Llama 3.1 8B	0.9	0.9	0.9	Perfect
DeepSeek R1 8B	0.9	0.65	0.65	Divergent

Figure 4.4 visualizes the agreement between CPS and T-CPS across thresholds for each model.

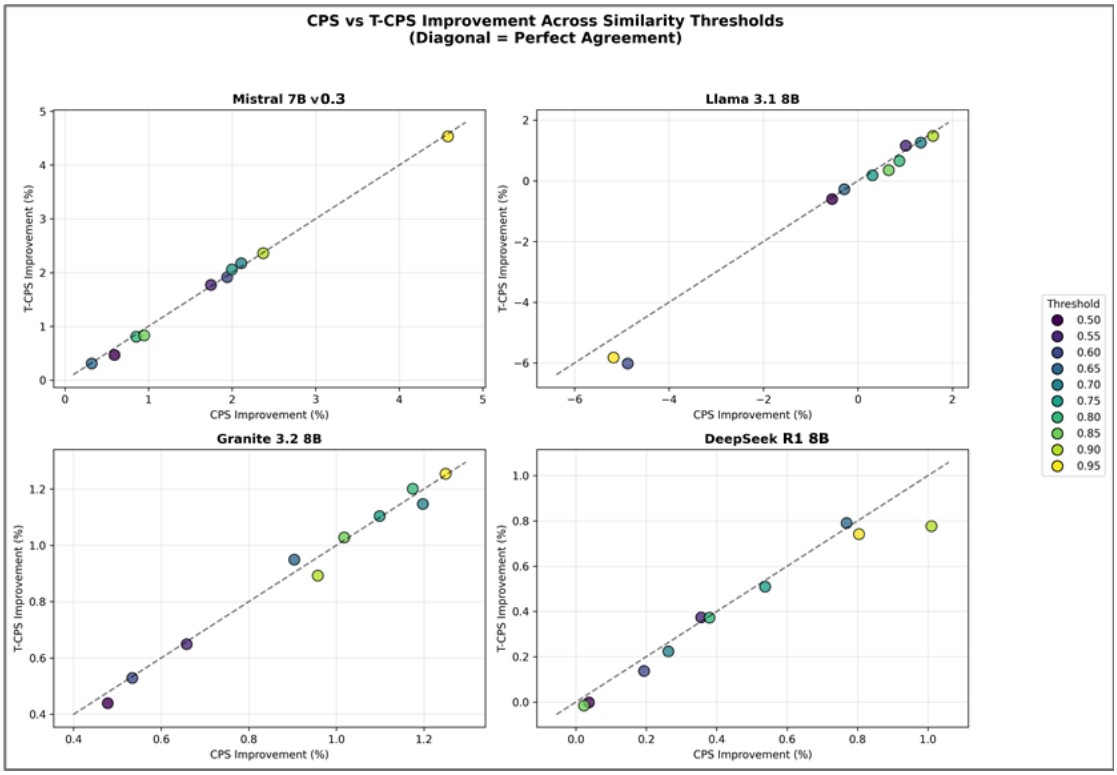

Figure 4.4 *Phase III (Agriculture, N = 369): CPS–T-CPS Agreement Across Thresholds per Model (Diagonal = Perfect Agreement).*

4.3.7 Statistical Significance

Two-tailed paired t-tests compared per-question CPS at each threshold against baseline without multiple-comparison correction. Significant positive improvements were most frequent for Granite 3.2 8B (7 of 10 thresholds) and Mistral 7B v0.3 (6 of 10), indicating broader effective threshold ranges under the tested configuration. DeepSeek R1 8B showed significant improvement at two thresholds, while Llama 3.1 8B showed one significant improvement together with two significant decreases. Table 4.12 summarizes the significance distribution, and Figure 4.5 presents the per-threshold heatmap.

Table 4.12 *Significance Distribution by Model (Agriculture Domain).*

Model	Significant Positive	Significant Negative	Not Significant
Mistral 7B v0.3	6	0	4
Granite 3.2 8B	7	0	3
Llama 3.1 8B	1	2	7
DeepSeek R1 8B	2	0	8

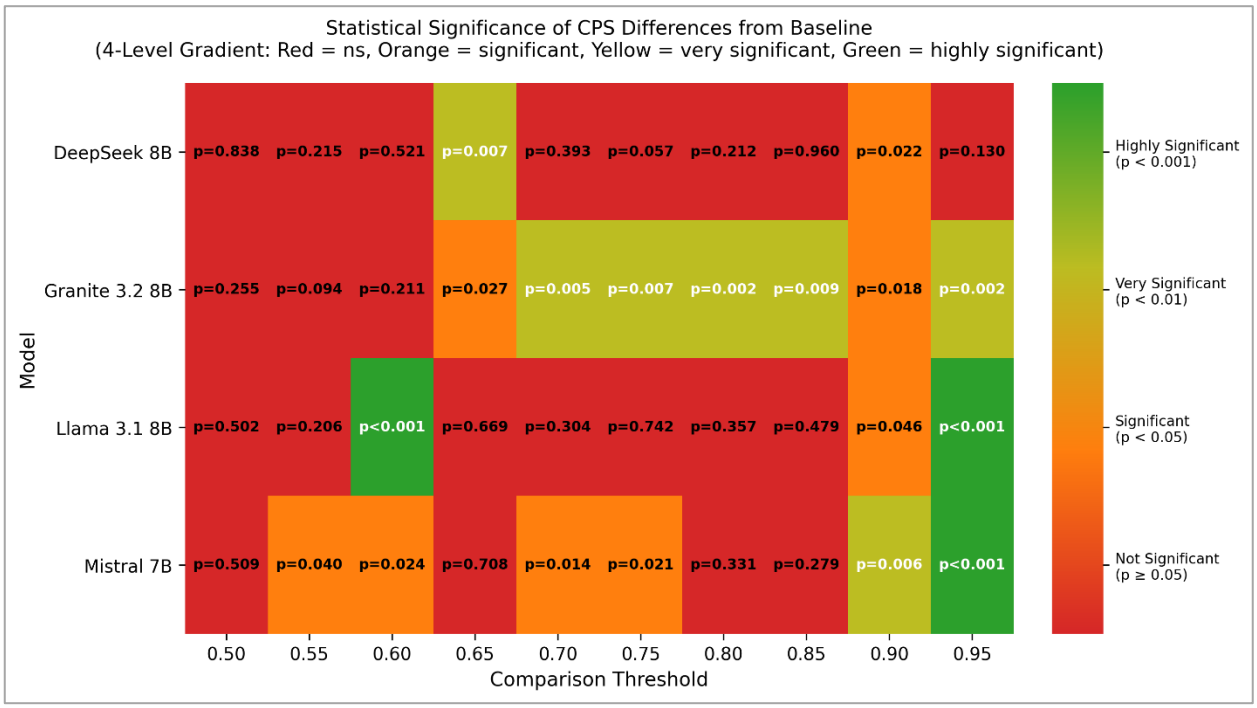

Figure 4.5 *Statistical significance heatmap of CPS differences from baseline across similarity thresholds (Phase III).*

4.3.8 Model-Specific Similarity Threshold Sensitivity Patterns

The Phase III results indicate distinct threshold-response profiles across models. Mistral 7B v0.3 shows the strongest overall pattern, with significant improvements from 0.55 to 0.95 and its highest performance at 0.95. Granite 3.2 8B also maintains a broad effective range, from 0.65 to 0.95, combined with relatively stable variability. In contrast, Llama 3.1 8B exhibits a narrower effective range and greater sensitivity to threshold selection. DeepSeek R1 8B shows more limited improvements and a divergence between the thresholds favored by mean performance and by stability-aware criteria.

4.3.9 Phase III Summary

Table 4.13 summarizes the best-performing configurations for each model together with their significance markers. The results show that similarity-threshold effects are model-dependent: Mistral 7B v0.3 and Granite 3.2 8B perform best at 0.95, Llama 3.1 8B at 0.90, and DeepSeek R1 8B shows different optima depending on whether performance alone or stability-aware criteria are emphasized. These findings provide evidence for RQ1 and RQ2 within the agricultural domain, while their generalizability beyond the tested configuration is examined in Phase IV.

Table 4.13 *Best-Performing Configurations Summary (Agriculture Domain).*

Model	Peak Threshold	CPS Impr. %	T-CPS Impr. %	CV	Balance Score	Sig.
Mistral 7B v0.3	0.95	4.58	4.54	0.134	0.339	***
Granite 3.2 8B	0.95	1.25	1.25	0.124	0.101	**
Llama 3.1 8B	0.9	1.58	1.48	0.148	0.1	*
DeepSeek R1 8B	0.65 / 0.90	0.77 / 1.01	0.79 / 0.78	0.085 / 0.108	0.093 / 0.072	** / *

** DeepSeek R1 8B shows divergent optima: similarity threshold 0.65 prioritizes stability-aware selection (T-CPS and Balance Score), while similarity threshold 0.90 prioritizes mean performance (CPS).*

4.4 Phase IV: Cross-Domain Evaluation (Biodiversity)

Phase IV extends the Phase III similarity-threshold analysis to a biodiversity domain in order to examine cross-domain generalization. The model set, threshold sweep (0.50-0.95, step 0.05), retrieval definitions, prompt format, chunking settings, embedding model, temperature (0.2), and evaluation framework remained unchanged. The main changes were the knowledge corpus, the biodiversity test set (N

= 426 question-answer pairs), and the use of a single execution environment (M1 Mac Mini). The full 24-metric set was computed for each run. The aim was to determine whether threshold effects and model-dependent sensitivity patterns persist when corpus characteristics differ, including vocabulary and embedding similarity distributions. Phase IV addresses RQ1, RQ2, and RQ3 by repeating the Phase III procedure in a second domain and comparing the resulting threshold-response patterns.

4.4.1 Experimental Design

All experiments were executed on a single M1 Mac Mini using a fixed 16,000-token context buffer across models and thresholds. This standardization removes context-buffer variation as a confounding factor and strengthens the interpretation of observed differences as domain-related rather than hardware-related. The biodiversity corpus comprised 426 question-answer pairs derived from authoritative sources, including the Convention on Biological Diversity [95] and the EU Biodiversity Strategy [96]. Reference answers were extracted from the source documents, and questions were generated with Claude Opus using the same general procedure as in Phase I. In total, 18,744 evaluations were conducted (4 models × 11 configurations × 426 question-answer pairs).

4.4.2 CPS Weighting Scheme

The same 9-metric weighting scheme used in Phase III (Table 4.6) was retained without modification. The weights preserve the four-construct evaluation framework: lexical overlap (30%), semantic similarity (25%), fluency and accuracy (25%), and language modeling (20%). T-CPS was again computed with $\alpha = 0.1$ and $\beta = 0.05$.

4.4.3 CPS Performance Overview

CPS performance across thresholds 0.50-0.95 was evaluated for all four models in the biodiversity domain. Table 4.14 reports the top three CPS improvement configurations for each model. Compared with the agriculture domain, the biodiversity results show larger CPS gains and lower peak thresholds across all models. The highest CPS improvements were observed for Mistral 7B v0.3 at 0.80 (+13.32%), DeepSeek R1 8B at 0.55 (+8.45%), Granite 3.2 8B at 0.80 (+6.95%), and Llama 3.1 8B at 0.85 (+2.06%).

Table 4.14 *Top CPS Improvement Configurations by Model (Top 3 Biodiversity)*

Model	Rank	Threshold	Mean CPS	Improvement %
Mistral 7B v0.3	1	0.8	0.4911	**13.32**
	2	0.65	0.4764	9.94
	3	0.7	0.4747	9.53
Granite 3.2 8B	1	0.8	0.4473	**6.95**
	2	0.95	0.4422	5.73
	3	0.55	0.4413	5.51
Llama 3.1 8B	1	0.85	0.4713	**2.06**
	2	0.7	0.4606	−0.25
	3	0.8	0.4567	−1.11
DeepSeek R1 8B	1	0.55	0.5094	**8.45**
	2	0.6	0.4906	4.45
	3	0.7	0.4775	1.66

4.4.4 T-CPS Performance and Stability

Table 4.15 summarizes the top T-CPS configurations for each model, incorporating variability through the coefficient of variation (CV). The stability-aware rankings remain model-dependent but largely coincide with the CPS results: Mistral 7B v0.3 and Granite 3.2 8B peak at 0.80, Llama 3.1 8B at 0.85, and DeepSeek R1 8B at 0.55. Among the top configurations, DeepSeek R1 8B shows the lowest variability (CV = 0.129-0.158), whereas the other models remain in a higher range (CV = 0.233-0.254), indicating more stable output quality across queries. This pattern is also visible in Figure 4.6, which compares threshold-wise CPS and T-CPS

improvements per model and highlights where stability adjustment changes the ranking relative to raw CPS improvement.

Table 4.15 *Top T-CPS Improvement Configurations by Model (Top 3 Biodiversity)*

Model	Rank	Threshold	T-CPS	T-CPS Impr. %	CV	Interpretation
Mistral 7B v0.3	1	0.8	0.5254	14.23	0.242	Large improvement
	2	0.65	0.5102	10.93	0.233	Large improvement
	3	0.7	0.5078	10.41	0.24	Large improvement
Granite 3.2 8B	1	0.8	0.4785	7.25	0.239	Moderate improvement
	2	0.55	0.4723	5.87	0.235	Moderate improvement
	3	0.95	0.472	5.8	0.254	Moderate improvement
Llama 3.1 8B	1	0.85	0.5042	2.29	0.24	Small improvement
	2	0.7	0.4928	−0.03	0.24	Minimal change
	3	0.8	0.4884	−0.92	0.241	Minimal change
DeepSeek R1 8B	1	0.55	0.5529	8.75	0.129	Large improvement
	2	0.6	0.5306	4.38	0.158	Moderate improvement
	3	0.7	0.5165	1.59	0.157	Small improvement

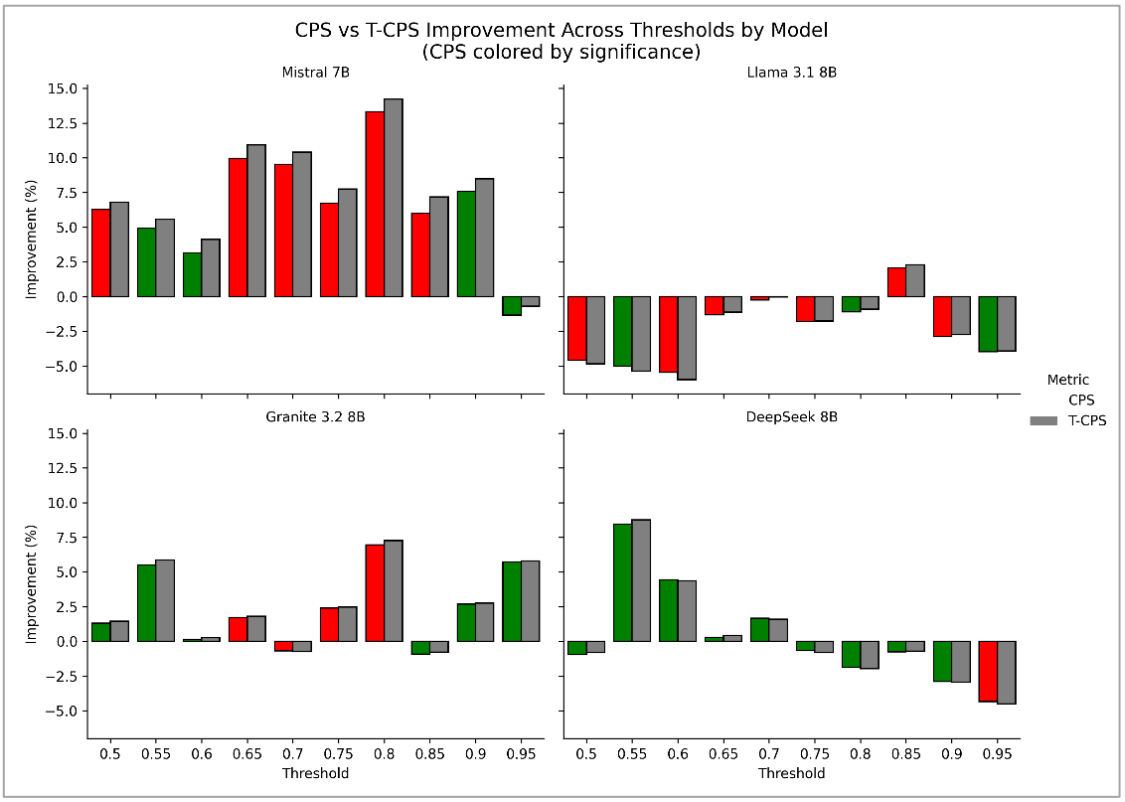

Figure 4.6 *Phase IV (Biodiversity, N = 426): Threshold-wise CPS and T-CPS improvements per model. CPS bars indicate percent improvement relative to baseline and are colored by significance at p < 0.05 (uncorrected). T-CPS bars (gray) indicate stability-aware improvement.*

4.4.5 Correlation Analysis

Correlation analysis was used to assess metric redundancy and to determine whether T-CPS contributes information beyond mean CPS. Figure 4.7 presents the Spearman correlation matrix for the Phase IV component metrics pooled across thresholds 0.50-0.95 plus baseline. Table 4.16 shows that T-CPS remains strongly associated with CPS (ρ ≈ 0.992), while also showing a substantial inverse association with CV (ρ ≈ -0.724). This indicates that, in Phase IV, T-CPS preserves mean-performance ranking while also expressing a stronger stability signal than in Phase III. The correlation structure additionally suggests tighter clustering among lexical overlap metrics and a domain-dependent shift in perplexity relationships, indicating that metric interdependencies are sensitive to corpus characteristics.

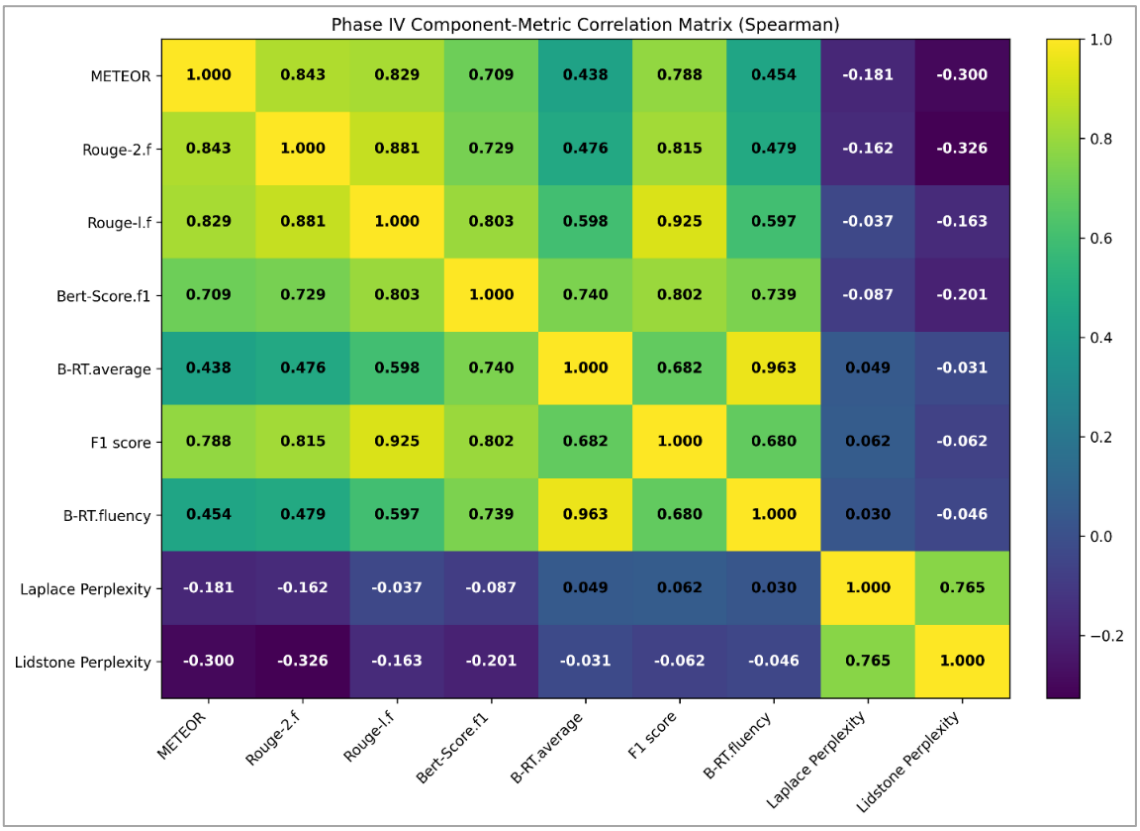

Figure 4.7 *Phase IV component metric correlation matrix (Spearman), pooled across models and similarity thresholds 0.50–0.95 plus baseline using per-question results.*

Table 4.16 *Phase IV associations between stability-aware scoring T-CPS, mean CPS, and variability CV (Spearman; N = 44 model and threshold configurations).*

Relationship	Spearman ρ
ρ(T-CPS, CPS)	0.992
ρ(T-CPS, CV)	-0.724

4.4.6 Balance Score

Balance Score ranks configurations by stability-adjusted improvement magnitude (Balance Score = (T-CPS improvement % / 100) / CV; Section 3.5.4). Table 4.17 reports the top 10 statistically significant positive configurations. DeepSeek R1 8B at threshold 0.55 achieves the highest Balance Score because it combines strong improvement with the lowest variability, while Mistral 7B v0.3 occupies several top positions through consistently large gains across thresholds.

Table 4.17 *Balance Score Ranking (top 10 statistically significant positive configurations)*

Rank	Model	Threshold	T-CPS Impr. %	CV	Balance Score	Sig.
1	DeepSeek R1 8B	0.55	+8.75	0.129	0.678	***
2	Mistral 7B v0.3	0.8	+14.23	0.242	0.588	***
3	Mistral 7B v0.3	0.65	+10.93	0.233	0.469	***
4	Mistral 7B v0.3	0.7	+10.41	0.24	0.434	***
5	Mistral 7B v0.3	0.9	+8.49	0.236	0.36	***
6	Mistral 7B v0.3	0.75	+7.74	0.228	0.34	***
7	Mistral 7B v0.3	0.85	+7.17	0.217	0.33	***
8	Granite 3.2 8B	0.8	+7.25	0.239	0.303	***
9	DeepSeek R1 8B	0.6	+4.38	0.158	0.277	***
10	Mistral 7B v0.3	0.5	+6.8	0.26	0.262	***

Table 4.18 shows full agreement among CPS, T-CPS, and Balance Score for all four models in the biodiversity domain, making threshold selection more straightforward than in Phase III.

Table 4.18 *Threshold Alignment Across Selection Criteria by Model*

Model	Best CPS Threshold	Best T-CPS Threshold	Best Balance Score Threshold	Alignment
Mistral 7B v0.3	0.8	0.8	0.8	Perfect
Granite 3.2 8B	0.8	0.8	0.8	Perfect
Llama 3.1 8B	0.85	0.85	0.85	Perfect
DeepSeek R1 8B	0.55	0.55	0.55	Perfect

Figure 4.8 plots CPS improvement against T-CPS improvement for each threshold per model. Points on the diagonal indicate that stability adjustment does not change the result, while deviations from the diagonal show where variability shifts T-CPS relative to mean CPS.

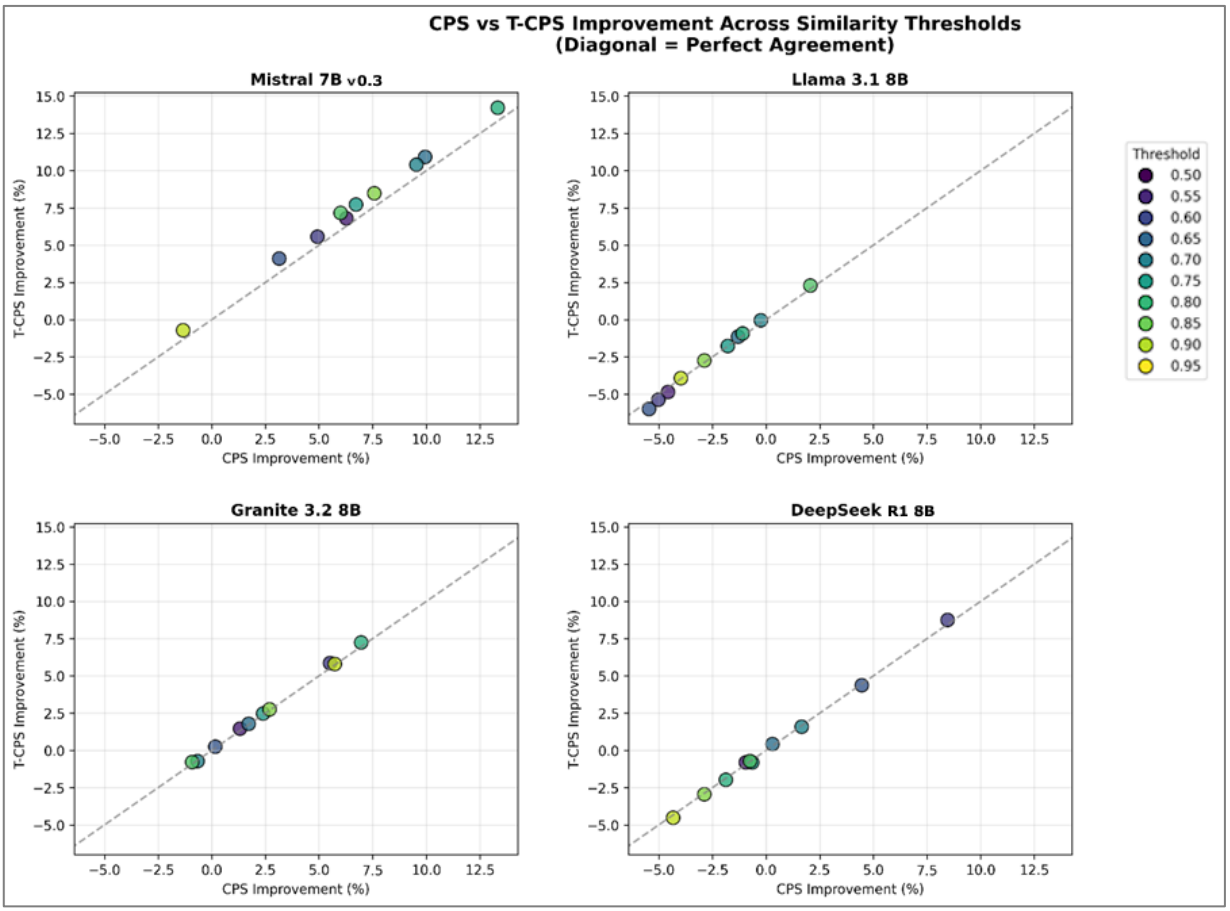

Figure 4.8 *Phase IV (Biodiversity, N = 426): CPS–T-CPS Agreement Across Thresholds per Model (Diagonal = Perfect Agreement).*

4.4.7 Statistical Significance

Two-tailed paired t-tests compared per-question CPS values at each similarity threshold against baseline without multiple-comparison correction. Table 4.19 summarizes the significance distribution by model, and Figure 4.9 presents the corresponding heatmap. Mistral 7B v0.3 shows significant positive improvements at 9 of 10 thresholds, indicating the broadest effective range in this domain. Granite 3.2 8B shows significant improvement at 3 thresholds, DeepSeek R1 8B at 2 thresholds together with 2 significant negative effects at stricter settings, and Llama 3.1 8B shows only 1 significant positive result alongside multiple significant decreases.

Table 4.19 *Significance Distribution by Model (Biodiversity Domain)*

Model	Significant Positive	Significant Negative	Not Significant
DeepSeek R1 8B	2	2	6
Granite 3.2 8B	3	0	7
Llama 3.1 8B	1	5	4
Mistral 7B v0.3	9	0	1

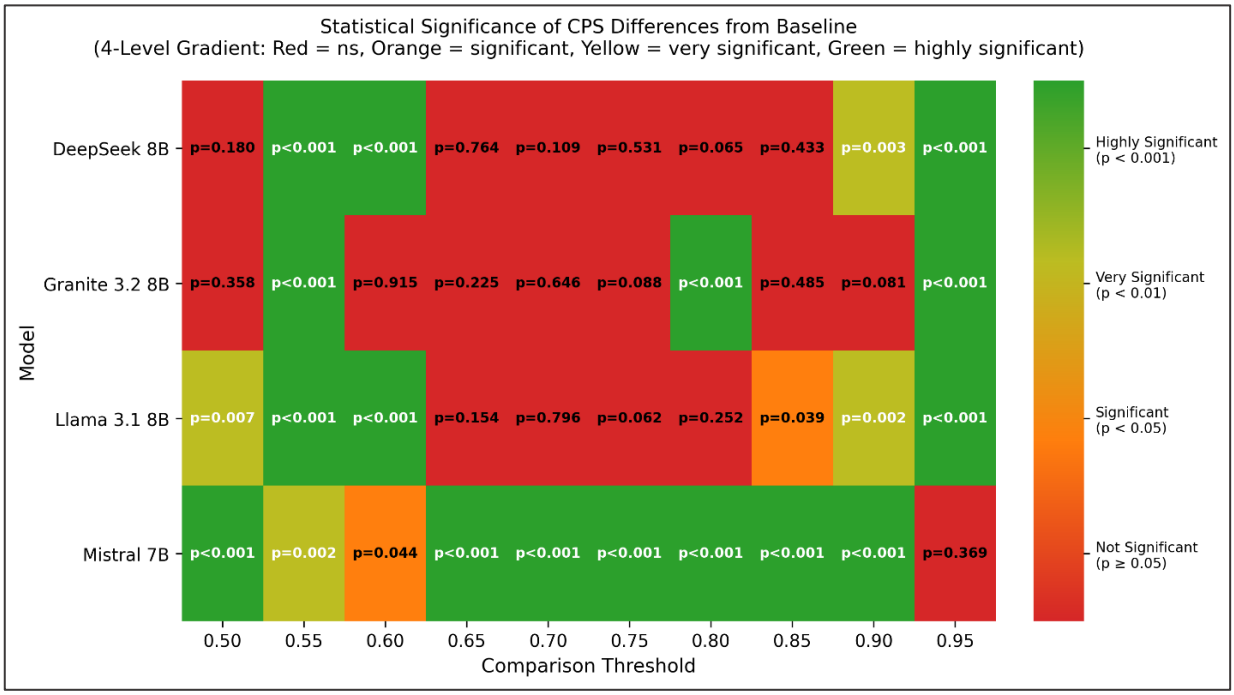

Figure 4.9 *Statistical significance heatmap of CPS differences from baseline across similarity thresholds (Phase IV).*

4.4.8 Model-Specific Similarity Threshold Sensitivity Patterns

Model-dependent sensitivity profiles remain visible under domain shift, although the effective threshold ranges and improvement magnitudes change. In the biodiversity domain, Mistral 7B v0.3 shows the broadest positive response, with peak performance at 0.80 (+13.32% CPS; +14.23% T-CPS) and improvements across most of the tested thresholds. Granite 3.2 8B shows a more moderate response, with significant gains at a smaller set of thresholds and its highest performance also at 0.80 (+6.95% CPS; +7.25% T-CPS). Llama 3.1 8B remains highly sensitive to threshold selection, with only one significant positive threshold at 0.85 and several thresholds associated with significant degradation. DeepSeek R1 8B shows an effective range concentrated at lower thresholds, especially 0.55-0.60, with strong gains at 0.55 but negative effects under stricter filtering. Owing to its lower variability, it also achieves the highest Balance Score in this phase.

4.4.9 Cross-Domain Comparison

Comparison between Phase IV and Phase III shows domain-dependent shifts in peak-performing thresholds and improvement magnitudes. Across all four models, the CPS-optimal thresholds are lower in the biodiversity domain than in agriculture: Mistral 7B v0.3 and Granite 3.2 8B shift from 0.95 to 0.80, Llama 3.1 8B from 0.90 to 0.85, and DeepSeek R1 8B from 0.90 to 0.55. Improvement magnitudes also increase in biodiversity, indicating stronger threshold sensitivity in that corpus. Threshold selection becomes more consistent as well: all four models show full agreement across CPS, T-CPS, and Balance Score, whereas in

agriculture DeepSeek R1 8B required a trade-off between mean performance and stability-aware ranking. Taken together, these results show that best-performing similarity thresholds are domain-dependent and should be calibrated for the target corpus rather than transferred unchanged across domains.

Table 4.20 *Cross-Domain Comparison: Agriculture (Phase III) vs. Biodiversity (Phase IV).*

Model	Agriculture Most Favorable Threshold	Agriculture CPS improvement %	Bio Most Favorable Threshold	Bio CPS improvement %	Shift
Mistral 7B v0.3	0.95	+4.58%	0.80	+13.32%	-0.15
Granite 3.2 8B	0.95	+1.25%	0.80	+6.95%	-0.15
Llama 3.1 8B	0.90	+1.58%	0.85	+2.06%	-0.05
DeepSeek R1 8B	0.90	+1.01%	0.55	+8.45%	-0.35

** Agriculture and biodiversity thresholds reflect CPS peaks. For DeepSeek R1 8B, the stability-aware optimum (T-CPS/Balance Score) in agriculture is 0.65; see Table 4.13.*

4.4.10 Phase IV Summary

Phase IV evaluated similarity-threshold sensitivity across four open-source language models using 426 question-answer pairs from the biodiversity domain under standardized hardware conditions. The best-performing configurations are summarized in Table 4.21: Mistral 7B v0.3 peaks at 0.80 (CPS +13.32%; T-CPS +14.23%), Granite 3.2 8B at 0.80 (+6.95%; +7.25%), Llama 3.1 8B at 0.85 (+2.06%; +2.29%), and DeepSeek R1 8B at 0.55 (+8.45%; +8.75%). DeepSeek R1 8B also achieves the highest Balance Score (0.678), reflecting the combination of strong gains and low output variability. At their peak thresholds, Mistral 7B v0.3, Granite 3.2 8B, and DeepSeek R1 8B show strong statistical support, while Llama 3.1 8B records a smaller but still positive gain. These results are bounded by the tested configuration of four models, 426 questions, the biodiversity domain, and a single standardized hardware environment. Within those limits, Phase IV provides evidence for RQ1 and RQ2 in the biodiversity domain, while the cross-domain implications for RQ3 are addressed in Section 4.4.9.

Table 4.21 Best-Performing Configurations Summary (Biodiversity Domain). Significance markers summarize paired t-test results;

Model	Peak Threshold	CPS Impr. %	T-CPS Impr. %	CV	Balance Score	Sig.
Mistral 7B v0.3	0.8	13.32	14.23	0.242	0.588	***
Granite 3.2 8B	0.8	6.95	7.25	0.239	0.303	***
Llama 3.1 8B	0.85	2.06	2.29	0.24	0.095	*
DeepSeek R1 8B	0.55	8.45	8.75	0.129	0.678	***

4.5 Chapter Summary

Chapter 4 examined how similarity threshold affects RAG generation quality across seven open-source language models and two knowledge domains using the PaSSER platform described in Chapter 2 and the evaluation framework defined in Chapter 3. Phase I established baseline system behavior under fixed top-k retrieval and compared runtime and quality results for three 7B models across two hardware environments. Phase II introduced threshold-based retrieval in a pilot sweep from 0.50 to 0.80 and showed that CPS varies by model. Phase III extended the analysis to four models and thresholds up to 0.95 in the agriculture domain, demonstrating model-dependent sensitivity under CPS, T-CPS, and Balance Score, supported by paired statistical testing against baseline. Phase IV repeated this threshold analysis in the biodiversity domain under standardized hardware and showed lower peak thresholds, larger improvement magnitudes, and full agreement among CPS, T-CPS, and Balance Score. Taken together, the results show that threshold effects are model-dependent and domain-sensitive, and that best-performing thresholds require corpus-specific calibration. Across all phases, the evaluation procedure, infrastructure, and experimental design components of Objective 1 were applied consistently, fulfilling Objective 4 and providing empirical support for Deficiency 1, Deficiency 2, and Deficiency 3.

CHAPTER 5: DISCUSSION AND FUTURE WORK

The dissertation is guided by one research aim and four objectives. Its aim is to develop an evaluation framework for Retrieval-Augmented Generation that supports evidence-based retrieval configuration decisions for RAG systems with open-source LLMs, with particular focus on similarity threshold configuration. Objective 1 defines and implements the core components of the framework: a threshold-aware evaluation procedure, the PaSSER platform as reproducibility infrastructure, and a controlled experimental design. Objectives 2 and 3 establish model selection criteria and define the evaluation metrics and computation procedures. Objective 4 applies these foundations through controlled testing and analysis across models and domains. Chapter 1 identifies the research gaps and corresponding deficiencies. Chapters 2 and 3 present the platform, model selection rationale, metrics, and computation procedures. Chapter 4 reports the experimental results. The findings are synthesized below: Section 5.1 addresses the research questions, Section 5.2 outlines the scientific-applied contributions, Section 5.3 discusses limitations, and Section 5.4 indicates future research directions.

5.1 Answers to Research Questions

Sections 5.1.1-5.1.3 synthesize the evidence from Phases II-IV in relation to the three research questions.

5.1.1 Threshold Effects on Generation Quality (RQ1)

Similarity threshold configuration produced statistically significant effects on generation quality. Across the agricultural and biodiversity corpora, threshold variation within the tested range of 0.50-0.95 yielded CPS improvements of up to +4.58% in Phase III and +13.32% in Phase IV relative to baseline.

The Phase II pilot provided the first indication that CPS varies across threshold settings even within a relatively narrow range. Using 101 question-answer pairs and three 7B-parameter models, it showed that peak-performing thresholds already differed by model, with Mistral 7B and Llama 2 7B peaking at 0.55 and Orca 2 7B at 0.65. These pilot results justified the broader threshold sweep and expanded model set used in the later phases.

Phase III extended the analysis to four models, 369 questions, and thresholds from 0.50 to 0.95, with statistical testing against baseline. Under these conditions, Granite 3.2 8B and Mistral 7B v0.3 showed the clearest evidence of systematic threshold sensitivity, with significant positive effects at 7 of 10 and 6 of 10 thresholds, respectively. DeepSeek R1 8B and Llama 3.1 8B showed fewer significant gains, and their threshold behavior was more constrained or mixed.

Phase IV applied the same threshold-evaluation procedure to the biodiversity corpus. The results again showed that threshold choice affects generation quality, but the magnitude and direction of those effects differed by model. Mistral 7B v0.3 showed significant improvement at 9 of 10 thresholds, whereas Granite 3.2 8B contracted from a broad effective range in Phase III to only 3 significant thresholds in Phase IV. DeepSeek R1 8B showed positive effects at lower thresholds and degradation at stricter ones, while Llama 3.1 8B showed predominantly negative threshold effects. Output variability also increased in Phase IV across all models, as reflected in higher coefficient of variation values.

Taken together, these findings provide an affirmative answer to RQ1: similarity threshold configuration produced measurable and, in many cases, statistically significant effects on generation quality. Improvements of up to +4.58% in agriculture and +13.32% in biodiversity were achieved through threshold calibration alone, without model retraining or architectural changes. At the same time, the best-performing threshold was not fixed across models or domains, indicating that threshold selection should be established empirically for the target configuration rather than assumed in advance.

5.1.2 Model-Dependent Similarity Threshold Sensitivity (RQ2)

The results show substantial model-dependent variation in similarity-threshold sensitivity. The models differed not only in their best-performing threshold values, but also in the breadth of their effective threshold ranges, the magnitude of their CPS improvements, and the consistency of their responses across

configurations.

Mistral 7B v0.3 showed the broadest and most consistent positive response. In Phase III, it achieved significant improvements across thresholds from 0.55 to 0.95, and in Phase IV this responsiveness expanded further, with significant gains at 9 of 10 tested thresholds and a peak CPS improvement of +13.32% at 0.80. Among the models evaluated in Phases III and IV, it was the only one to maintain a broad positive threshold-response profile across both domains.

Granite 3.2 8B showed a different pattern. In the agricultural domain, it exhibited a broad plateau of significant gains across thresholds from 0.65 to 0.95. In the biodiversity domain, however, that range narrowed substantially to 3 significant thresholds. This makes Granite 3.2 8B the clearest example of contraction in sensitivity breadth under domain change, although the interpretation remains qualified by the Phase III-IV procedural differences discussed in Section 5.3.2.

Llama 3.1 8B showed a much narrower positive range in agriculture and predominantly negative effects in biodiversity. In Phase IV, 6 of 10 thresholds were significant, but only one of these was positive. This reversal suggests that threshold calibration for Llama 3.1 8B is especially sensitive to corpus conditions and may become unstable when retrieval characteristics change.

DeepSeek R1 8B showed comparatively limited responsiveness in Phase III, but a more distinctive pattern in Phase IV. In biodiversity, it improved at lower thresholds and degraded at stricter ones, producing a bifurcated threshold-response profile. Although its mean CPS gains were not the largest overall, its lower coefficient of variation contributed to the highest Balance Score observed in Phase IV. This indicates that DeepSeek R1 8B becomes more competitive when stability-aware evaluation is taken into account.

These results provide an answer to RQ2: threshold sensitivity differed substantially across models in both magnitude and form. Some models showed broad and stable positive response, others showed narrow or contracting effective ranges, and others changed direction across domains. Threshold selection therefore cannot be generalized across models, since a setting that improves one model may have little effect or even a negative one for another

5.1.3 Cross-Domain Generalization (RQ3)

Threshold sensitivity was compared between the agricultural corpus in Phase III and the biodiversity corpus in Phase IV. The comparison revealed systematic cross-domain differences, although these must be interpreted cautiously because corpus, question-generation procedure, and hardware conditions were not all held constant between phases.

The clearest pattern is that the best-performing thresholds shifted downward in Phase IV for all four models. In biodiversity, each model reached its highest CPS at a lower threshold than in agriculture, with the largest downward shift observed for DeepSeek R1 8B. Improvement magnitudes also increased for most models. Mistral 7B v0.3, Granite 3.2 8B, and DeepSeek R1 8B all achieved substantially larger gains in Phase IV than in Phase III, whereas Llama 3.1 8B remained comparatively weak and showed predominantly negative threshold effects. The relative ranking of models also changed across domains: DeepSeek R1 8B moved from the weakest peak CPS result in agriculture to the second strongest in biodiversity, while Llama 3.1 8B dropped from second to fourth.

The cross-domain shift was visible not only in the peak thresholds and improvement magnitudes, but also in the structure of the metric relationships. The correlation patterns among component metrics differed between Phases III and IV, indicating that the interaction between overlap, semantic similarity, fluency, and variability was sensitive to corpus conditions. In addition, threshold selection behavior became more consistent in biodiversity, where CPS, T-CPS, and Balance Score aligned for all four models, unlike the agriculture phase where DeepSeek R1 8B showed divergence between mean-performance and stability-aware criteria.

Collectively, these results support the answer to RQ3: threshold-response patterns did not transfer unchanged across domains. The direction of the shift was systematic, with lower best-performing thresholds

and larger gains in the biodiversity phase, but the exact effects depended on the model. This indicates that threshold calibration should be treated as corpus-specific rather than assumed to generalize automatically from one knowledge domain to another.

5.2 Scientific-applied Contributions

Three scientific-applied contributions are defined as layers of an integrated evaluation framework that address the deficiencies identified in the Introduction: a threshold-aware evaluation procedure with composite scoring (C1, Evaluation Procedure layer, addressing D1), reproducibility infrastructure with blockchain-based provenance logging (C2, Infrastructure layer, addressing D2), and comparative empirical guidance for open-source LLM deployment (C3, Evidence layer, addressing D3). Sections 5.2.1-5.2.3 summarize these contributions.

5.2.1 Threshold-aware Evaluation Procedure

Section 1.4 showed that existing evaluation frameworks such as RAGAS, RGB, the TREC RAG Track, and TruLens typically assess RAG outputs under fixed retrieval settings and do not treat similarity threshold as an explicit experimental variable. In many implementations, threshold selection remains a heuristic configuration choice, often adopted without systematic justification. Existing approaches also tend to emphasize mean performance and rarely evaluate how output quality varies across queries.

To address this gap, the present work developed a threshold-aware evaluation procedure in which similarity threshold is treated as an independent variable and system behavior is evaluated across a retrieval-selectivity range rather than at a single operating point. This procedure was implemented through the PaSSER platform and applied across Phases II-IV. Phase II introduced pilot threshold variation in the range 0.50-0.80. Phase III extended the sweep to 0.95 and incorporated statistical testing in the agricultural domain. Phase IV repeated the same procedure in the biodiversity domain to examine cross-domain variation.

The results show that threshold effects are substantial and cannot be reduced to a fixed default. In Phase III, peak-performing thresholds by CPS fell within 0.90-0.95, while stability-aware criteria expanded the effective range to 0.65-0.95. In Phase IV, the best-performing thresholds shifted downward to 0.55-0.85. Performance differences across thresholds reached +4.58% CPS improvement in agriculture and +13.32% in biodiversity, indicating that evaluation at a single threshold may fail to represent system behavior across the retrieval range.

This contribution also includes three composite scoring formulations. CPS aggregates heterogeneous metrics through weighted summation after normalization, using a 9-metric panel distributed across lexical overlap (30%), semantic similarity (25%), fluency and accuracy (25%), and language modeling (20%). T-CPS extends CPS by incorporating the coefficient of variation (CV), thereby accounting for output stability across queries. Balance Score further evaluates the relation between improvement and variability.

Across Phases III and IV, T-CPS and CPS were highly correlated ($\rho > 0.99$), indicating that mean performance remained the dominant ranking signal in the tested configurations. This does not make T-CPS redundant. Rather, it shows that stability-aware ranking usually aligned with mean performance, while still revealing cases in which the preferred threshold changed once variability was considered. The clearest example is DeepSeek R1 8B in Phase III, where CPS peaked at 0.90 but T-CPS favored 0.65 because lower variability shifted the stability-aware optimum.

Overall, this contribution addresses Deficiency 1 by replacing fixed-threshold evaluation with a threshold-aware procedure that captures both performance level and performance stability. The reported findings remain exploratory, and confirmatory studies with stronger statistical control would further strengthen generalizability.

5.2.2 Reproducibility Infrastructure

The second contribution is the PaSSER platform, developed to address reproducibility problems in RAG evaluation. PaSSER provides a browser-based environment for controlled experimentation, integrating retrieval configuration, model execution, metric calculation, and result export within a single workflow.

A central feature of this infrastructure is blockchain-based provenance logging. For each experimental run, the platform records evaluation metrics, timing data, and run identifiers on the Antelope blockchain, while the backend preserves the associated configuration context, including the model identifier, retrieval parameters, decoding settings, and dataset identifiers. This design supports immutable records, verifiable timestamps, and later retrieval of experiment metadata for independent inspection.

Blockchain logging does not guarantee exact replication of the full computational environment. However, it strengthens reproducibility by preserving the experimental conditions and recorded outputs in a verifiable form. In this way, the contribution addresses Deficiency 2 not by claiming perfect replicability, which would be convenient but false, but by improving transparency, traceability, and post hoc verification of evaluation runs.

The materials associated with this dissertation are publicly available in two GitHub locations. The phase-organized thesis archive, containing the experimental results and supporting research materials, is available at https://github.com/M33rschaum/passer-thesis-archive. The current PaSSER implementations, including the original PaSSER platform and its related repositories such as maPaSSER and the PaSSER-SR, are available through the GitHub organization at https://github.com/scpdxtest.

5.2.3 Practical Guidance for Open-Source Deployments

The third contribution is comparative empirical guidance for selecting and configuring open-source LLMs under RAG conditions. This contribution addresses the lack of systematic evidence for open-source deployment candidates, especially where model choice and retrieval configuration must be made under security, cost, or infrastructure constraints.

Using the model selection criteria and metric procedures defined in Chapter 3, and the controlled experiments reported in Chapter 4, the dissertation generated more than 38,000 evaluations across four phases, two domains, and multiple hardware settings. These results make threshold sensitivity visible and comparable across deployment candidates in the 7-8B parameter range.

Four findings are especially relevant for practice. First, models differ substantially in threshold-response profile: some show broad and stable positive sensitivity, while others exhibit narrow, unstable, or bifurcated behavior. Second, the effective threshold ranges identified in Phases III and IV provide starting points for calibration rather than reliance on generic defaults. Third, cross-domain shifts show that threshold settings should be re-evaluated when the corpus changes. Fourth, Balance Score provides an additional basis for deployment decisions when consistency matters alongside mean performance.

The observed threshold effects are especially relevant to two retrieval-related failure points described in prior work [47]: exclusion of relevant evidence under overly strict filtering, and admission of weakly related material under overly permissive filtering. The present results suggest that model susceptibility to these failure points differs across systems, although the causal mechanisms were not isolated in this dissertation and would require dedicated ablation studies.

Taken as a whole, this contribution addresses Deficiency 3 by providing empirical guidance for threshold calibration and model comparison in open-source RAG deployment scenarios, with attention to both average quality and stability under changing corpus conditions.

5.3 Limitations

Several factors limit the extent to which the reported findings can be generalized beyond the evaluated corpora, models, and experimental settings.

5.3.1 Scope Constraints

The experimental analysis covers two domains: agriculture in Phase III and biodiversity in Phase IV. Both domains are characterized by technical and regulatory content, formal language, and relatively well-defined terminology. It therefore remains unclear whether similar threshold-sensitivity patterns would appear in domains with less structured language, broader topical variation, or more ambiguous reference material.

The evaluated model set was limited to open-source systems in the 7-8B parameter range in order to keep experimentation feasible on mid-range hardware. Threshold sensitivity may differ for smaller, larger, or more heavily specialized models, particularly where retrieval use, context utilization, or decoding behavior changes substantially.

5.3.2 Experimental Design Limitations

The experiments were conducted under hardware conditions that varied across phases. Phase III used M1, M2, and CPU-only environments together with context buffers ranging from 2,048 to 10,000 tokens. Phase IV standardized execution to a single M1 environment with a fixed 16,000-token context buffer. While this removed hardware and context-buffer variation within Phase IV, it also introduced simultaneous differences relative to Phase III in domain, question-generation procedure, and execution setup.

Vector stores were built using Mistral 7B embeddings through the Ollama embedding endpoint, with fixed chunking parameters of 1,024 characters and 50-character overlap across all phases. This ensured consistent retrieval inputs across generators, but it may also have introduced embedding-generator alignment effects. Because Mistral 7B served both as the embedding model and as one of the evaluated generators, a possible alignment confound cannot be excluded. Isolating that effect would require experiments with independent embedding models.

Question generation also differed across phases. Agricultural questions in Phases I-III were generated with Mistral 7B, whereas Phase IV used Claude Opus. This provides some procedural variation, but it does not isolate the contribution of question generation to the observed threshold patterns.

As a result, the comparison between Phases III and IV differs in several respects beyond domain content alone, including question-generation model, hardware standardization, and corpus characteristics. The cross-domain findings should therefore be interpreted as preliminary evidence that calibration may need to be corpus-specific, rather than as a clean test of domain effects alone.

5.3.3 Measurement and Analysis Limitations

Some of the evaluation metrics also impose interpretive limits. The perplexity metrics, Laplace and Lidstone, were computed with NLTK n-gram language models rather than transformer token probabilities and therefore function as model-independent proxies rather than direct estimates from the evaluated generators. The B-RT metric suite uses [CLS]-based projections and has not yet been validated against human judgments.

Phases III and IV report uncorrected p-values in order to preserve statistical power. Each phase includes 40 paired comparisons against baseline, meaning that at $\alpha = 0.05$ approximately two significant results per phase would be expected by chance under the null hypothesis. For that reason, interpretation should focus on broader patterns across thresholds and models rather than on isolated p-values. The reported findings are therefore exploratory and should not be treated as definitive threshold prescriptions.

5.3.4 Causal Interpretation

The experimental design establishes empirical associations between similarity-threshold configuration and generation performance, including model-dependent optima, cross-domain shifts, and stability-performance trade-offs captured through Balance Score. However, it does not isolate the mechanisms responsible for these patterns. Multiple factors vary simultaneously across conditions, including corpus characteristics, question formulation, hardware configuration, and model architecture. This prevents attribution of the observed effects to any single cause. Mechanistic claims would require controlled ablation studies in which one factor is varied at a time while the remaining components are held constant.

5.4 Future Work

Future work can be grouped into three directions: scope extensions, procedural extensions, and validation and verification.

5.4.1 Scope Extensions

Future studies should extend threshold-sensitivity analysis beyond the agriculture and biodiversity domains and beyond open-source models in the 7-8B parameter range. Controlled domain variation, in which

only the corpus changes, would help isolate domain effects from the design confounds described in Section 5.3.2. Model coverage could also be expanded to include smaller and larger models, fine-tuned variants, and, where feasible, proprietary systems evaluated under matched conditions.

5.4.2 Procedural extensions

Further work should investigate adaptive threshold selection, where retrieval selectivity varies according to query characteristics rather than remaining fixed. The PaSSER platform could also be extended to record retrieval-level provenance, including retrieved passages and similarity scores, and to integrate IPFS for content-addressed storage of corpora and datasets. Another useful direction is the analysis of efficiency-quality trade-offs through systematic measurement of latency, memory use, and throughput across threshold settings.

5.4.3 Validation and Verification

Future research should also include human validation studies to determine whether thresholds selected by CPS and T-CPS correspond to improvements perceived by human evaluators, and to assess the validity of B-RT against human judgments. Controlled ablation studies are also needed to isolate the effects of embedding alignment, chunking strategy, question generation, hardware conditions, and domain-only changes. In addition, alternative measurement approaches should be explored, including transformer-native perplexity where logits are available, together with multiple-comparison corrections such as Bonferroni or Benjamini-Hochberg false discovery rate control in confirmatory studies.

5.5 Chapter Summary

Chapter 5 interpreted the experimental findings in relation to the three research questions, outlined the three scientific-applied contributions of the dissertation, and identified the main limitations affecting scope, design, measurement, and causal interpretation. It also defined directions for future work aimed at expanding domain and model coverage, improving retrieval-level instrumentation, and strengthening validation through human assessment and controlled ablation. These elements position the reported results as a structured basis for continued investigation of threshold-aware RAG evaluation rather than as fixed prescriptions for all deployment settings.

CONCLUSION – RESUME OF THE OBTAINED RESULTS

An evaluation framework for retrieval-augmented generation (RAG) was developed, integrating three layers: an Evaluation Procedure layer introducing a threshold-aware evaluation procedure, an Infrastructure layer implementing reproducibility infrastructure, and an Evidence layer producing practical guidance for open-source deployments. The framework addresses three deficiencies identified in current RAG evaluation practice: the absence of threshold-aware evaluation, insufficient reproducibility infrastructure, and the lack of practical guidance for open-source RAG deployments.

The PaSSER platform was designed and implemented as a browser-accessible, open-source application supporting configurable retrieval parameter testing, multi-metric evaluation with composite scoring, and blockchain-based provenance logging via the Antelope ledger. Three composite scoring instruments were developed: the Composite Performance Score (CPS) for unified threshold comparison, the Threshold-aware Composite Performance Score (T-CPS) incorporating output consistency through a coefficient-of-variation-based reward-penalty structure, and the Balance Score quantifying the stability-performance trade-off.

Seven open-source language models in the 7–8 billion parameter range were evaluated across four experimental phases, two application domains (agriculture and biodiversity), and over 38,000 individual evaluations. Phase I validated end-to-end platform functionality under fixed top-k retrieval. Phase II introduced threshold-aware evaluation and provided initial evidence that threshold sensitivity varies across models. Phase III extended the analysis to four newer models across a broader threshold range (0.50–0.95) with statistical validation, revealing CPS improvements of up to 4.58% in the agricultural domain and identifying two distinct sensitivity profiles: broad improvement zones and narrow effective ranges. Phase IV

replicated the evaluation in the biodiversity domain, where substantially larger threshold effects were observed — peak CPS improvements reached 13.32%. Peak-performing threshold configurations shifted downward for all four models when moving from agriculture to biodiversity, with shifts ranging from −0.05 (Llama 3.1 8B) to −0.35 (DeepSeek R1 8B). Output variability increased by 67–105% (coefficient of variation) relative to agriculture, confirming that threshold sensitivity is domain-dependent and not solely a model-intrinsic property.

CPS and T-CPS alignment analysis demonstrated that mean performance and consistency-aware assessment can yield divergent threshold recommendations, with two of four models showing different optima under the two scoring instruments.

All four objectives were addressed. Objective 1 was realized through three components: the reproducibility infrastructure with blockchain-based provenance logging (b, Chapter 2), the threshold-aware evaluation procedure with composite scoring (a, Chapters 3–4), and the controlled experimental design producing comparative evidence across models and domains (c, Chapter 4). Objective 2 was addressed through definition of model selection criteria aligned with deployment feasibility, licensing, and computational requirements (Chapter 3). Objective 3 was fulfilled through definition and consistent implementation of metric computation procedures across five evaluation constructs, with aggregation into three composite scoring instruments (Chapter 3). Objective 4 was achieved through controlled experimentation across four phases, two domains, and over 38,000 evaluations under systematic threshold variation (Chapter 4).

Three practical conclusions emerge from the experimental evidence. First, threshold calibration produces measurable and statistically significant improvements, but effective configurations depend on both model architecture and knowledge domain - no single threshold setting generalizes across all conditions. Second, consistency-aware scoring is recommended over mean-performance-only assessment, as it prevents selection of high-performing but unstable configurations. Third, systematic threshold evaluation should be repeated when changing the application domain, as peak-performing thresholds shifted downward for all four tested models when moving from agriculture to biodiversity.

Three scientific-applied contributions — together constituting the evaluation framework — result from the research. The end-to-end traceability from identified deficiencies through research questions and objectives to contributions is shown in Figure C.1.

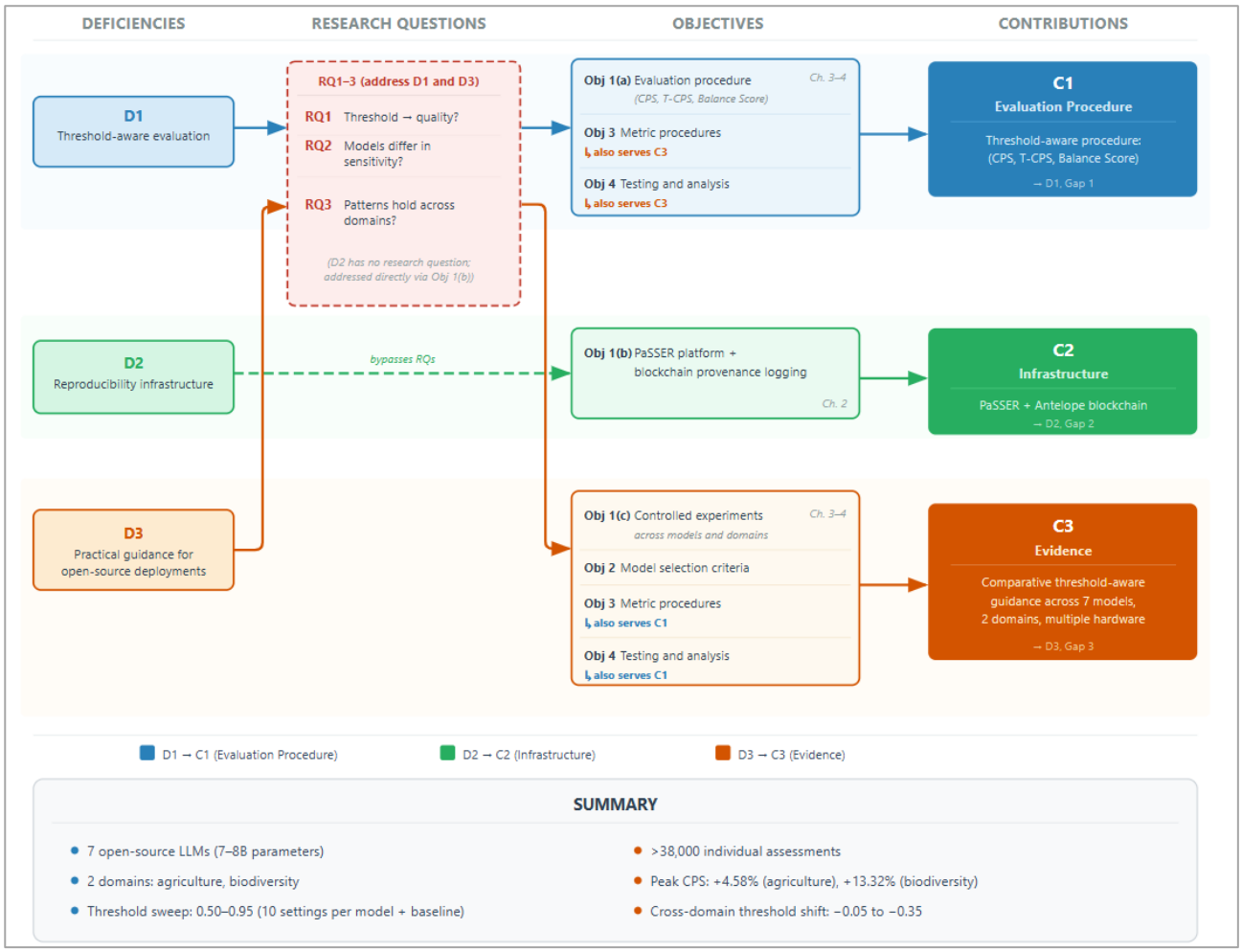

Figure C.1 *End-to-End Traceability Map*

SUPPORTING PUBLICATIONS

Five peer-reviewed publications underpin the reported research:

[15] I. Radeva, I. Popchev, and M. Dimitrova, **"Similarity thresholds in retrieval-augmented generation,"** Proceedings of the 12th IEEE International Conference on Intelligent Systems - IS'24, 29-31 August 2024, Varna, Bulgaria, IEEE Xplore, 2024, ISBN:979-8-3503-5098-2, ISSN:2832-4145, DOI:10.1109/IS61756.2024.10705214, 1-7.
This work supports the CPS formulation and threshold sensitivity analysis presented in Chapter 4 (Phase II).

[16] M. Dimitrova, I. Popchev, and I. Radeva, **"PaSSER: A platform for evaluating LLMs in RAG,"** Proceedings of the 9th IEEE International Conference on Big Data, Knowledge and Control Systems Engineering – BdKCSE'2025, 06-07 November 2025, Bankya, Bulgaria, IEEE Xplore, 2025, ISSN:979-8-3315-8712-3, DOI:10.1109/BdKCSE67969.2025.11300500, 1-7.
This work describes the PaSSER platform architecture and functionalities detailed in Chapter 2.

[17] I. Radeva, I. Popchev, L. Doukovska, and M. Dimitrova, **"Web application for retrieval-augmented generation: Implementation and testing,"** Electronics, 13, 7, MDPI, Basel, Switzerland, 2024, ISSN:2079-9292, DOI:10.3390/electronics13071361, 1-31. SJR (Scopus):0.64, JCR-IF (Web of Science):2.9.
This work presents the PaSSER platform and metrics discussed in Chapter 2.

[18] I. Radeva, I. Popchev, L. Doukovska, and M. Dimitrova, **"Multi-agent coordination strategies vs. retrieval-augmented generation in LLMs: A comparative evaluation,"** Electronics, 14, 24, MDPI, 2025, ISSN:2079-9292, DOI:10.3390/electronics14244883, SJR (Scopus):0.62.
This work documents the T-CPS and Balanced Score reported in Chapter 4 (Phase IV).

[20] M. Dimitrova, **"Retrieval-augmented generation (RAG): Advances and challenges,"** Problems of Engineering Cybernetics and Robotics, 83, Prof. Marin Drinov Academic Publishing House, 2025, ISSN:2738-7356, DOI:10.7546/PECR.83.25.03, 32-57.
This work provides the RAG literature review and frameworks analysis that form the foundation of Chapter 1.

Publications [17] and [18] are indexed in JCR-IF (Web of Science) and SJR (Scopus). Conference papers [15] and [16] are indexed in IEEE Xplore Digital Library. Article [20] is published by Prof. Marin Drinov Academic Publishing House of the Bulgarian Academy of Sciences.

CITATION RECORD

[1] I. Radeva, I. Popchev, L. Doukovska, and M. Dimitrova, Web Application for Retrieval-Augmented Generation: Implementation and Testing. Electronics, 13, 7, MDPI, Basel, Switzerland, 2024, ISSN:2079-9292, DOI:10.3390/electronics13071361, 1-31. SJR (Scopus):0.64, JCR-IF (Web of Science):2.9.

Cited by:

[1.1] H. Andersson, "Retrieval-augmented generation with Azure OpenAI," M.S. thesis, Malardalen Univ., 2024.

[1.2] S. D'Urso, B. Martini, and F. Sciarrone, "A Novel LLM Architecture for Intelligent System Configuration," in Proc. Int. Conf. Information Visualisation (IV), Coimbra, Portugal, 2024, pp. 326-331, doi:10.1109/IV64223.2024.00063.

[1.3] D. Firdaus, I. Sumardi, and Y. Kulsum, "Integrating Retrieval-Augmented Generation With Large Language Model Mistral 7B for Indonesian Medical Herb," JISKA, vol. 9, no. 3, pp. 230-243, 2024, doi:10.14421/jiska.2024.9.3.230-243.

[1.4] H. Zhang, Z. Li, F. Liu, Y. He, Z. Cao, and Y. Zheng, "Design and Implementation of LangChain-based Chatbot," in Proc. Int. Seminar on AI, Computer Technology and Control Engineering (ACTCE), Wuhan, China, 2024, pp. 226-229, doi:10.1109/ACTCE65085.2024.00053.

[1.5] J. G. Ongris, E. Tjitrahardja, F. Darari, and F. J. Ekaputra, "Towards an Open NLI LLM-based System for KGs: A Case Study of Wikidata," in Proc. 7th Int. Seminar on Research of IT and Intelligent Systems (ISRITI), 2024, pp. 44-49, doi:10.1109/ISRITI64779.2024.10963661.

[1.6] C. K. Kitengera and M. K. Kasambya, "Developpement d'une plateforme web d'evaluation des enseignements...," Revue Internationale Multidisciplinaire Etincelle, vol. 25, no. 2, pp. 1-22, 2024, doi:10.61532/rime252117.

[1.7] B. Lu, "Evaluating LLMs on large contexts: a RAG approach on text comprehension," Master's thesis, Univ. de Liege, 2024.

[1.8] M. M. Li, I. Nikishina, O. Sevgili, and M. Semman, "Wiping out the limitations of Large Language Models: A Taxonomy for Retrieval Augmented Generation," arXiv:2408.02854, 2024, doi:10.48550/arXiv.2408.02854.

[1.9] M. Olaosegba, "Next-Gen AI Optimization Tools for AWS Cloud Cost Control," IJFMR, 2024.

[1.10] P. Phukon, Y. Lokhar, and P. P. Ray, "Localized Open-Source LLM Aware RAG of Legal Documents...," in Proc. Int. BIT Conf. (BITCON), 2024, pp. 1-6, doi:10.1109/BITCON63716.2024.10985396.

[1.11] S. Rani, S. G. Deepika, D. Devdharshini, and H. Ravindran, "Augmenting Code Sequencing with RAG...," in Proc. SSITCON, 2024, pp. 1-7, doi:10.1109/SSITCON62437.2024.10796587.

[1.12] S. Dudhmande et al., "Textual Compression Using Lamini-LM," IRJAEM, vol. 2, no. 5, pp. 1536-1540, 2024, doi:10.47392/IRJAEM.2024.0208.

[1.13] S. Bouzid and L. Piron, "Leveraging Generative AI in Short Document Indexing," Electronics, vol. 13, no. 17, 2024, doi:10.3390/electronics13173563.

[1.14] K. Traykov, "A Framework for Security Testing of Large Language Models," in Proc. 12th IEEE Int. Conf. on Intelligent Systems (IS), Varna, Bulgaria, 2024, pp. 1-7, doi:10.1109/IS61756.2024.10705238.

[1.15] L. Werkman, "Assessing the potential of leveraging LLaMA-2...," thesis, Lulea Univ. of Technology, 2024.

[1.16] W. Wilmi and N. Roslund, "Implementering av RAG for automatiserad analys av hallbarhetsrapportering...," thesis, KTH Royal Inst. of Technology, 2024.

[1.17] Y. Xu et al., "Development of an Enterprise Knowledge Base System Based on Elasticsearch," in Proc. ISPCEM, 2024, pp. 186-190, doi:10.1109/ISPCEM64498.2024.00039.

[1.18] Y. Song, Enhancing Classroom Dialogue Productiveness: Exploring the Potential of Artificial Intelligence. London, U.K.: Routledge, 2024, doi:10.4324/9781003543039.

[1.19] Z. Zhong et al., "Mix-of-Granularity: Optimize the Chunking Granularity for RAG," arXiv:2406.00456, 2024.

[1.20] J. O. Agada et al., "A Systematic Review of Key RAG Systems...," arXiv:2507.18910, 2025, doi:10.48550/arXiv.2507.18910.

[1.21] A. Guyyala et al., "RAG-based AI Agents for Multilingual Help Desks...," Int. J. Computer Applications, vol. 187, no. 56, pp. 15-28, 2025, doi:10.5120/ijca2025925964.

[1.22] O. Barcelos et al., "Technological Convergence Identification Model (TCIM)...," Revista E-TECH, vol. 18, no. 1, 2025, doi:10.18624/e-tech.v18i1.1444.

[1.23] C. Yu et al., "Safety Devolution in AI Agents," 2025, doi:10.48550/arXiv.2505.14215.

[1.24] D. Costa et al., "Mycroft: Retrieval Augmented Generation for SDK Documentation," in Proc. NATL, 2025, doi:10.5121/csit.2025.152211.

[1.25] R. Dayarathne et al., "Comparing the Performance of LLMs in RAG-Based QA...," in AI in Education Technologies, LNDECT, vol. 228. Singapore: Springer, 2025, doi:10.1007/978-981-97-9255-9_26.

[1.26] E. H. Omoush et al., "Advancing Arabic Medical QA Systems with RAG...," in Proc. ICTCS, 2025, pp. 511-516, doi:10.1109/ICTCS65341.2025.10989446.

[1.27] Y. Fan et al., "Research on the Online Update Method for RAG Model...," in Proc. NNICE, 2025, pp. 1740-1744, doi:10.1109/NNICE64954.2025.11063821.

[1.28] F. Shen et al., "Development of a Convenient Accounting System Based on SpringBoot+Vue," in Proc. CITSC, 2025, pp. 167-171, doi:10.1109/CITSC64390.2025.00038.

[1.29] F. Ehrlich-Sommer et al., "ForestGPT and Beyond...," Electronics, vol. 14, no. 18, p. 3583, 2025, doi:10.3390/electronics14183583.

[1.30] H. Mahfoud et al., "AI Chatbots for Healthcare Maintenance...," TQM J., 2025, doi:10.1108/TQM-10-2024-0394.

[1.31] G. Iieva and G. A. Tsihrintzis, "Editorial Note to Special Issue...," Electronics, vol. 14, no. 10, p. 1925, 2025, doi:10.3390/electronics14101925.

[1.32] L. A. Sanjani et al., "Performance Analysis of LLM Models with RAG and Fine-Tuning T5...," in Proc. ICoCSETI, 2025, pp. 152-157, doi:10.1109/ICoCSETI63724.2025.11018908.

[1.33] B. T. Mahardika and A. M. Hasan, "Application of GPT in Chatbots...," Eduvest, vol. 5, no. 6, pp. 6235-6247, 2025, doi:10.59188/eduvest.v5i6.51321.

[1.34] N. A. Akbar et al., "Novel Approach for Leveraging Agent-Based Experts...," in AIxIA 2024, LNCS, vol. 15450. Cham, Switzerland: Springer, 2025, doi:10.1007/978-3-031-80607-0_2.

[1.35] B. M. Praneeth et al., "Optimization of Customer Feedback Summarization...," IEEE Access, vol. 13, pp. 124319-124332, 2025, doi:10.1109/ACCESS.2025.3588337.

[1.36] P. Pany, "Reasoning Engine with Pre-Trained LLMs: An Operation GPT," IJRASET, vol. 13, no. 4, pp. 2452-2463, 2025, doi:10.22214/ijraset.2025.68761.

[1.37] S. K. Mahjour and S. S. Mahjour, "Intelligent Reservoir Decision Support...," 2025, doi:10.48550/arXiv.2509.11376.

[1.38] S. Chen et al., "Customized large-scale model for human-AI collaborative operation...," Appl. Energy, vol. 393, pp. 126-169, 2025, doi:10.1016/j.apenergy.2025.126169.

[1.39] T. Jung and I. Joe, "An Intelligent Docent System with a Small Language Model (sLLM) Based on RAG," Appl. Sci., vol. 15, no. 17, p. 9398, 2025, doi:10.3390/app15179398.

[1.40] C.-N. Tirpescu and E. Velescu, "Enhancing Veterinary Education...," Procedia Comput. Sci., vol. 270, pp. 3828-3837, 2025, doi:10.1016/j.procs.2025.09.508.

[1.41] W. Ke et al., "Large Language Models in Document Intelligence: A Comprehensive Survey...," ACM Trans. Inf. Syst., vol. 44, no. 1, 2025, doi:10.1145/3768156.

[1.42] A. J. Winata et al., "Utilizing Large Language Models for Developing Automatic Question Generation in Education," in Proc. ICADEIS, 2025, doi:10.1109/ICADEIS65852.2025.10933227.

[1.43] Y. Benitez-Morejon et al., "Question-Answering Systems for Tourism...," in MISNC 2025, CCIS, vol. 2729. Cham, Switzerland: Springer, doi:10.1007/978-3-032-09945-7_22.

[1.44] J. Qi, Mitigating Translation Hallucinations in Large Language Models: A Chain of Thought and RAG-Based Approach, Ph.D. research proposal, The Chinese Univ. of Hong Kong, 2024-2025.

[1.45] R. Kumar and Y. Qu, "Utilizing Large Language Model Enabled Agents to Streamline Business Decision Making," Eur. J. Electr. Eng. Comput. Sci., vol. 9, no. 5, pp. 14-21, Sep. 2025, doi:10.24018/ejece.2025.9.5.717.

[1.46] I. P. A. E. Pratama, I. M. O. Widyantara, Linawati, and N. Gunantara, "Bibliometric Analysis of AI-Based Prototype Proposal for User Security Awareness in Healthcare," JOIV: Int. J. Informatics Visualization, vol. 9, no. 3, pp. 982-994, May 2025, doi:10.62527/joiv.9.3.3319.

[1.47] S. Gandla, Automated Test Code Generation from Textual Descriptions Using Generative AI, Master's thesis, Blekinge Inst. of Technology, 2024.

[1.48] N. S. Patil, A. J. Koyande, A. V. Thakur, P. B. Kadam, and P. G. Moholkar, "RAG Chatbots: Implementing Large Language Models in Retrieval-Augmented Generations," in Smart Trends in Computing and Communications, LNNS, vol. 1363, pp. 401-410, 2025, doi:10.1007/978-981-96-2885-8_33.

[1.49] I. Balen, Sustav korisnicke podrske temeljen na bazi znanja i koristenju alata umjetne inteligencije za brzo odgovaranje na ponavljajuca pitanja korisnika, Undergraduate thesis (Zavrsni rad), Faculty of Electrical Engineering and Computing (FER), Univ. of Zagreb, Jun. 2024.

[1.50] Y. Jiao, S. Ouyang, M. Zhong, Y. Zhang, L. Ding, S. Zhou, and J. Han, "Retrieval and Structuring Augmented Generation with LLMs for Web Applications," in Companion Proc. ACM Web Conf. 2025 (WWW '25 Companion), pp. 25-28, May 2025, doi:10.1145/3701716.3715870.

[1.51] Z. Liu, Design and Implementation of an AI-based Agent to Inform Best Practices on Test Case Execution Routines, Master's thesis, Univ. of Zurich, Jun. 29, 2025, doi:10.5167/uzh-278942.

[2] I. Radeva, I. Popchev, and M. Dimitrova, Similarity Thresholds in Retrieval-Augmented Generation. Proceedings of the 12th IEEE International Conference on Intelligent Systems - IS'24, 29-31 August 2024, Varna, Bulgaria, IEEE Xplore, 2024, ISBN:979-8-3503-5098-2, ISSN:2832-4145, DOI:10.1109/IS61756.2024.10705214, 1-7.

Cited by:

[2.1] D. Ayepah-Mensah et al., "A RAG-Assisted DRL Framework for Microservices Deployment in 6G Vehicular Networks," in Proc. WiMob 2025, Marrakesh, Morocco, 2025, pp. 1-6, doi:10.1109/WiMob66857.2025.11257559.

[2.2] Y. Bondalapati and H. N. BM, "Scalable RAG with Kubernetes for Enhanced Document Intelligence," in Proc. CICCS 2025, Bengaluru, India, 2025, pp. 1-6, doi:10.1109/CICCS66437.2025.11280266.

[2.3] A. Jadhav et al., "AI-Driven Diagnosis Predictive Chatbot for Healthcare," in Proc. WorldSUAS 2025, 2025, doi:10.1109/WorldSUAS66815.2025.11199219.

[2.4] J. Van Nooten et al., "One Size Does Not Fit All: Exploring Variable Thresholds for Distance-Based Multi-Label Text Classification," arXiv:2510.11160, 2025, doi:10.48550/arXiv.2510.11160.

[2.5] X. Sun, C. Liang, Q. Wang, et al., "Mesh RAG: Retrieval Augmentation for Autoregressive Mesh Generation," arXiv:2511.16807, 2025.

[2.6] K. Traykov and Y. Kolova, "Analysis of Methods for Evaluating Responses of LLMs in Retrieval-Augmented Generation," in Proc. Int. Conf. on Big Data, Knowledge and Control Systems Engineering, 2025, pp. 1-6.

[2.7] A. Kosar, W. Daelemans, and G. De Pauw, Dont Make Me Guess: Automatically Detecting and Naming Topics in Large Collections of Text. Antwerp, Belgium: Univ. of Antwerp, Faculty of Arts, 2025.

[2.8] J. Van Nooten and W. Daelemans, The Many Faces of a Text: Applications and Enhancements of Multi-Label Text Classification Algorithms. Antwerp, Belgium: Univ. of Antwerp, Faculty of Arts, 2025.

[2.9] T. Bosi, Design, Implementation and Benchmarking of a Retrieval-Augmented Chatbot for the Insurance Sector, Master's thesis (Laurea magistrale), Univ. of Bologna, 2025.

[2.10] J. Karkoush and M. Ali, Kallgranskning med RAG och smasprakmodeller, Student thesis (Basic level, 15 HE credits), Univ. of Gavle, 2025, 46 pp., URN: urn:nbn:se:hig:diva-47778.

[3] I. Radeva, I. Popchev, L. Doukovska, and M. Dimitrova, Multi-Agent Coordination Strategies vs Retrieval-Augmented Generation in LLMs: A Comparative Evaluation. Electronics, 14, 24, MDPI, 2025, ISSN:2079-9292, DOI:10.3390/electronics14244883, SJR (Scopus):0.62.

Cited by:

[3.1] M. Nababan and G. Simarmata, "Model Matematika Dalam Pemilihan Mekanisme Koordinasi...," Jurnal Ilmiah Matematika (JIMAT), vol. 6, no. 2, pp. 891-902, Dec. 2025, doi:10.63976/jimat.v6i2.1201.

[3.2] S. Schmulling and G. Sanrocco, "Ensembles of Small Language Models as an Efficient Alternative to Large Language Models," course report (II2202, Fall 2025, Period 1/Period 1-2), KTH Royal Inst. of Technology, Stockholm, Sweden, Jan. 14, 2026.

[4] M. Dimitrova, Retrieval-Augmented Generation (RAG): Advances and Challenges. Problems of Engineering Cybernetics and Robotics, 83, Prof. Marin Drinov Academic Publishing House, 2025, ISSN:2738-7356, DOI:10.7546/PECR.83.25.03, 32-57.

Cited by:

[4.1] M. E. Koutsiaki, M. Delianidi, C. Mizeli, K. Diamantaras, I. Grigoropoulos, and N. Koutlianos, "From Textbook to Talkbot: A Case Study of a Greek-Language RAG-Based Chatbot in Higher Education," arXiv:2601.14265, 2025

BIBLIOGRAPHY

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.
[2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 1877–1901, 2020.
[3] Z. Ji, N. Qiu, S. Xu, D. Young, F. Tao, L. Lyu, C. Chen, C. Gu, R. Li, L. Yang, and P. Fung, "Survey of hallucination in natural language generation," *ACM Comput. Surv.*, vol. 55, no. 12, pp. 1–38, Dec. 2023, doi: 10.1145/3571730.

[4] K. Shuster, S. Poff, M. Chen, D. Kiela, and J. Weston, "Retrieval augmentation reduces hallucination in conversation," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 3784–3803, Nov. 2021, doi: 10.18653/v1/2021.findings-emnlp.320.
[5] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-T. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, pp. 9459–9474, 2020.
[6] N. Rossi, G. M. Gupta, S. Agarwal, S. Srinivasan, J. Liu, S. Han, and Y. Gao, "Relevance filtering for embedding-based retrieval," in *Proc. ACM Int. Conf. on Information and Knowledge Management (CIKM)*, 2024, doi: 10.1145/3627673.3680095.
[7] A. Pal, L. K. Umapathi, and M. Sankarasubbu, "Med-HALT: Medical domain hallucination test for large language models," in *Proc. 27th Conf. on Computational Natural Language Learning (CoNLL)*, pp. 314–334, 2023, doi: 10.18653/v1/2023.conll-1.21.
[8] J. Cui, Z. Li, Y. Yan, B. Chen, and L. Yuan, "ChatLaw: Open-source legal large language model with integrated external knowledge bases," *arXiv preprint arXiv:2306.16092*, Jun. 2023, doi: 10.48550/arXiv.2306.16092.
[9] Y. Bang, S. Cahyawijaya, N. Lee, W. Dai, D. Su, B. Wilie, H. Lovenia, Z. Ji, T. Yu, W. Chung, Q. V. Do, Y. Xu, S. Barezi, P. Pascual, H. Li, R. Shick, S. Joty, B. Shin, and P. Fung, "A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity," in *Proc. Int. Joint Conf. on Natural Language Processing and the Asia-Pacific Chapter of the ACL (IJCNLP-AACL)*, 2023.
[10] Y. Gao, Y. Xiong, X. Wang, J. Wang, Z. Jiang, H. Li, Y. Wen, K. Jiang, N. Meng, L. Shao, and P. Sethi, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint arXiv:2312.10997*, Dec. 2023, doi: 10.48550/arXiv.2312.10997.
[11] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008.
[12] M. Baker, "1,500 scientists lift the lid on reproducibility," *Nature*, vol. 533, no. 7604, pp. 452–454, May 2016, doi: 10.1038/533452a.
[13] R. Pradeep, N. Thakur, S. Upadhyay, D. Campos, N. Craswell, and J. Lin, "Initial nugget evaluation results for the TREC 2024 RAG track with the AutoNuggetizer framework," *arXiv preprint arXiv:2411.09607*, Nov. 2024, doi: 10.48550/arXiv.2411.09607.
[14] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proc. AAAI Conf. on Artificial Intelligence (AAAI)*, vol. 38, no. 16, pp. 17754–17762, 2024.
[15] I. Radeva, I. Popchev, and M. Dimitrova, "Similarity thresholds in retrieval-augmented generation," in *2024 IEEE 12th Int. Conf. on Intelligent Systems (IS)*, Aug. 2024, pp. 1–7, doi: 10.1109/IS61756.2024.10705214.
[16] M. Dimitrova, I. Popchev, and I. Radeva, "PaSSER: A platform for evaluating LLMs in RAG," in *2025 IEEE BdKCSE*, 2025, p. 7, doi: 10.1109/BdKCSE67969.2025.11300500.
[17] I. Radeva, I. Popchev, L. Doukovska, and M. Dimitrova, "Web application for retrieval-augmented generation: Implementation and testing," *Electronics*, vol. 13, no. 7, p. 1361, Apr. 2024, doi: 10.3390/electronics13071361.
[18] I. Radeva, I. Popchev, L. Doukovska, and M. Dimitrova, "Multi-agent coordination strategies vs. retrieval-augmented generation in LLMs: A comparative evaluation," *Electronics*, vol. 14, no. 24, p. 4883, Jan. 2025, doi: 10.3390/electronics14244883.
[19] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed. draft. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/ (accessed Jan. 20, 2026).
[20] M. Dimitrova, "Retrieval-augmented generation (RAG): Advances and challenges," Problems of Engineering Cybernetics and Robotics (PECR), vol. 83, Jul. 2025, doi: 10.7546/PECR.83.25.03.
[21] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *ACM Trans. Inf. Syst.*, vol. 43, no. 2, pp. 1–55, Mar. 2025, doi: 10.1145/3703155.
[22] J. Huang, X. Chen, S. Mishra, H. S. Liao, J. J. Chung, H. G. Song, and D. Zhou, "Large language models cannot self-correct reasoning yet," *arXiv preprint arXiv:2310.01798*, Oct. 2023, doi: 10.48550/arXiv.2310.01798.
[23] V. Karpukhin, B. Oğuz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-T. Yih, "Dense passage retrieval for open-domain question answering," in *Proc. EMNLP*, Nov. 2020, pp. 6769–6781, doi: 10.18653/v1/2020.emnlp-main.550.

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186, doi: 10.18653/v1/N19-1423.
[25] A. Shrivastava and P. Li, "Asymmetric LSH (ALSH) for sublinear time maximum inner product search (MIPS)," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2014.
[26] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with GPUs," *IEEE Trans. Big Data*, vol. 7, no. 3, pp. 535–547, Sep. 2021, doi: 10.1109/TBDATA.2019.2921572.
[27] Y. A. Malkov and D. A. Yashunin, "Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 824–836, Apr. 2020, doi: 10.1109/TPAMI.2018.2889473.
[28] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguist. (ACL)*, 2020, doi: 10.18653/v1/2020.acl-main.703.
[29] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2014.
[30] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, J. Polosukhin, J. Devlin, K. Lee, K. N. Toutanova, L. Jones, M. Kelcey, M. W. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov, "Natural questions: A benchmark for question answering research," *Trans. Assoc. Comput. Linguist.*, vol. 7, pp. 452–466, 2019, doi: 10.1162/tacl_a_00276.
[31] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer, "TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguist. (ACL)*, 2017, pp. 1601–1611, doi: 10.18653/v1/P17-1147.
[32] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "FEVER: A large-scale dataset for fact extraction and VERification," in *Proc. NAACL-HLT*, 2018, pp. 809–819, doi: 10.18653/v1/N18-1074.
[33] P. Bajaj, D. Campos, N. Craswell, L. Deng, C. Majumder, X. Qu, B. de Rossi, A. Rodriguez, B. Bhaskar, R. Lin, S. Sayyaparaju, and J. Shao, "MS MARCO: A human generated MAchine reading COmprehension dataset," *arXiv preprint arXiv:1611.09268*, Nov. 2016, doi: 10.48550/arXiv.1611.09268.
[34] A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-RAG: Learning to retrieve, generate, and critique through self-reflection," *arXiv preprint arXiv:2310.11511*, Oct. 2023, doi: 10.48550/arXiv.2310.11511.
[35] S.-Q. Yan, J.-C. Gu, Y. Zhu, and Z.-H. Ling, "Corrective retrieval augmented generation," *arXiv preprint arXiv:2401.15884*, Jan. 2024, doi: 10.48550/arXiv.2401.15884.
[36] S. Jeong, J. Baek, S. Cho, S. J. Hwang, and J. C. Park, "Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity," in *Proc. NAACL-HLT (Long Papers)*, 2024, pp. 7036–7050.
[37] Z. Jiang, F. F. Xu, L. Gao, J. Sun, Q. Liu, J. Dwivedi-Yu, Y. Yang, C. Callison-Burch, and G. Neubig, "Active retrieval augmented generation," in *Proc. EMNLP*, 2023, pp. 7969–7992, doi: 10.18653/v1/2023.emnlp-main.495.
[38] "Reducing false positives in retrieval-augmented generation (RAG) semantic caching: A banking case study," *InfoQ*. Accessed: Jan. 21, 2026. [Online]. Available: https://www.infoq.com/articles/reducing-false-positives-retrieval-augmented-generation/
[39] A. Celikyilmaz, E. Clark, and J. Gao, "Evaluation of text generation: A survey," *arXiv preprint arXiv:2006.14799*, Jun. 2020, doi: 10.48550/arXiv.2006.14799.
[40] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated evaluation of retrieval augmented generation," in *Proc. EACL (Demos)*, 2024.
[41] R. Pradeep, N. Thakur, S. Upadhyay, D. Campos, N. Craswell, and J. Lin, "Ragnarök: A reusable RAG framework and baselines for TREC 2024 retrieval-augmented generation track," *arXiv preprint arXiv:2406.16828*, Jun. 2024, doi: 10.48550/arXiv.2406.16828.
[42] "Getting Started," *TruLens*. Accessed: Jan. 31, 2026. [Online]. Available: https://www.trulens.org/getting_started/
[43] "LangSmith Evaluation," *LangChain Docs*. Accessed: Jan. 31, 2026. [Online]. Available: https://docs.langchain.com/langsmith/evaluation
[44] "Observability concepts," *LangChain Docs*. Accessed: Jan. 31, 2026. [Online]. Available: https://docs.langchain.com/langsmith/observability-concepts

[45] "Home," *Arize Phoenix*. Accessed: Jan. 31, 2026. [Online]. Available: https://phoenix.arize.com/
[46] "DeepEval," *Confident AI*. Accessed: Jan. 31, 2026. [Online]. Available: https://deepeval.com/
[47] S. Barnett, S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek, "Seven failure points when engineering a retrieval augmented generation system," *arXiv preprint arXiv:2401.05856*, Jan. 2024, doi: 10.48550/arXiv.2401.05856.
[48] T. Yu, S. Zhang, and Y. Feng, "Auto-RAG: Autonomous retrieval-augmented generation for large language models," *arXiv preprint arXiv:2411.19443*, Nov. 2024, doi: 10.48550/arXiv.2411.19443.
[49] D. Edge, H. Trinh, B. Cheng, J. Bradley, N. Chao, A. Mody, S. Truitt, and J. Larson, "From local to global: A graph RAG approach to query-focused summarization," *arXiv preprint arXiv:2404.16130*, Apr. 2024, doi: 10.48550/arXiv.2404.16130.
[50] Z. Guo, L. Xia, Y. Yu, T. Ao, and C. Huang, "LightRAG: Simple and fast retrieval-augmented generation," *arXiv preprint arXiv:2410.05779*, Oct. 2024, doi: 10.48550/arXiv.2410.05779.
[51] Z. Wang, J. Cho, S. S. Kim, S. J. Hwang, S. Lee, and J. G. Park, "Speculative RAG: Enhancing retrieval augmented generation through drafting," *arXiv preprint arXiv:2407.08223*, Jul. 2024, doi: 10.48550/arXiv.2407.08223.
[52] I. Popchev, L. Doukovska, and I. Radeva, "A prototype of blockchain/distributed file system platform," in *2022 IEEE 11th Int. Conf. Intell. Syst. (IS)*, Oct. 2022, pp. 1–7, doi: 10.1109/IS57118.2022.10019715.
[53] I. Popchev, L. Doukovska, and I. Radeva, "A framework of blockchain/IPFS-based platform for smart crop production," in *2022 Int. Conf. Automatics and Informatics (ICAI)*, Oct. 2022, pp. 265–270, doi: 10.1109/ICAI55857.2022.9960070.
[54] I. Popchev and I. Radeva, "Decentralized application (dApp) development and implementation," *Cybernetics and Information Technologies*, vol. 24, no. 2, pp. 122–141, Jun. 2024, doi: 10.2478/cait-2024-0019.
[55] AntelopeIO, "Antelope," GitHub repository. Accessed: Jun. 14, 2025. [Online]. Available: https://github.com/AntelopeIO
[56] IPFS, "IPFS Documentation," Accessed: Jun. 14, 2025. [Online]. Available: https://docs.ipfs.tech/
[57] I. Popchev, I. Radeva, and L. Doukovska, "Oracles integration in blockchain-based platform for smart crop production data exchange," *Electronics*, vol. 12, no. 10, Art. no. 2244, Jan. 2023, doi: 10.3390/electronics12102244.
[58] Greymass, "greymass/anchor: Antelope Desktop Wallet and Authenticator," GitHub repository. Accessed: Jan. 08, 2026. [Online]. Available: https://github.com/greymass/anchor
[59] EOSio Support, "Anchor Wallet Overview," Accessed: Mar. 15, 2023. [Online]. Available: https://eosio.support/anchor-wallet-overview/
[60] Chroma, "Chroma," Accessed: Jan. 21, 2026. [Online]. Available: https://www.trychroma.com
[61] Chroma Research, "Evaluating chunking strategies for retrieval," Accessed: Feb. 04, 2026. [Online]. Available: https://research.trychroma.com/evaluating-chunking
[62] LangChain, "LangChain," Accessed: Jun. 14, 2025. [Online]. Available: https://www.langchain.com
[63] Ollama, "Ollama," Accessed: Jan. 21, 2026. [Online]. Available: https://ollama.com
[64] PyPI, "pyntelope," Accessed: Jun. 14, 2025. [Online]. Available: https://pypi.org/project/pyntelope/
[65] Modal, "How much VRAM do I need for LLM inference?," Accessed: Feb. 04, 2026. [Online]. Available: https://modal.com/blog/how-much-vram-need-inference
[66] J. Manchanda, L. Boettcher, M. Westphalen, and J. Jasser, "The open source advantage in large language models (LLMs)," *arXiv preprint arXiv:2412.12004*, Dec. 2024, doi: 10.48550/arXiv.2412.12004.
[67] AI21, "What is a long context window? Benefits & use cases," Accessed: Feb. 04, 2026. [Online]. Available: https://www.ai21.com/knowledge/long-context-window/
[68] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022.
[69] Mistral AI, "Announcing Mistral 7B," Accessed: Jan. 21, 2026. [Online]. Available: https://mistral.ai/news/announcing-mistral-7b/
[70] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B.

Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Itani, K. Jiotomo, M. Levy, W. Luo, T. Scialom, G. Sun, K. S. Balaji, A. Sagun, E. Grave, S. Goyal, T. Izacard, A. Kushman, P. Luc, S. Iyer, A. Lomeli, Y. Low, J. Martin, P. Bhargava, M. Sastry, S. Singh, M. Singh, T. Majid, R. Williams, T. Scialom, and J. Zettlemoyer, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, Jul. 2023, doi: 10.48550/arXiv.2307.09288.

[71] A. Mitra, H. S. Liao, M. Moussawi, A. S. Atanasova, A. S. Sestari, H. Song, J. G. Park, J. J. Chung, and J. Huang, "Orca 2: Teaching small language models how to reason," *arXiv preprint arXiv:2311.11045*, Nov. 2023, doi: 10.48550/arXiv.2311.11045.

[72] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Castro, M. S. Lauw, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, "Mistral 7B," *arXiv preprint arXiv:2310.06825*, Oct. 2023, doi: 10.48550/arXiv.2310.06825.

[73] R. Rastogi, "Papers explained: Mistral 7B," DAIR.AI (Medium). Accessed: Mar. 06, 2024. [Online]. Available: https://medium.com/dair-ai/papers-explained-mistral-7b-b9632dedf580

[74] D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt, "Measuring massive multitask language understanding," in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2021.

[75] GSM8K, "openai/grade-school-math," GitHub repository. Accessed: Feb. 04, 2026. [Online]. Available: https://github.com/openai/grade-school-math

[76] M. Suzgun, N. S. Abid, A. Adam, E. Ahumada, A. Bansal, T. B. Brown, W. J. Child, E. Choi, D. S. Weld, and L. Zettlemoyer, "Challenging BIG-Bench tasks and whether chain-of-thought can solve them," *arXiv preprint arXiv:2210.09261*, Oct. 2022, doi: 10.48550/arXiv.2210.09261.

[77] IBM, "IBM Granite 3.2: open source reasoning and vision," Accessed: Jan. 21, 2026. [Online]. Available: https://www.ibm.com/new/announcements/ibm-granite-3-2-open-source-reasoning-and-vision

[78] DeepSeek-AI, "deepseek-ai/DeepSeek-R1," GitHub repository. Accessed: Jan. 21, 2026. [Online]. Available: https://github.com/deepseek-ai/DeepSeek-R1

[79] Meta AI, "Introducing Llama 3.1: Our most capable models to date," Accessed: Jan. 21, 2026. [Online]. Available: https://ai.meta.com/blog/meta-llama-3-1/

[80] Mistral AI, "Mistral 7B," Accessed: Jan. 21, 2026. [Online]. Available: https://docs.mistral.ai/models/mistral-7b-0-3

[81] DeepSeek-AI, C. Guo, M. Yang, Z. Bi, K. Zhou, F. Wang, W. Liu, Z. Shao, D. Wang, and G. Dai, "DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning," *arXiv preprint arXiv:2501.12948*, Jan. 2025, doi: 10.48550/arXiv.2501.12948.

[82] A. Grattafiori, J. Santua, K. Stone, P. Albert, S. Batra, K. J. Chen, A. Chou, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Itani, K. Jiotomo, A. Kushman, P. Luc, M. Martin, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. Canton Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, B. Fuller, C. Gao, V. Goswami, and N. Goyal, "The Llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, Jul. 2024, doi: 10.48550/arXiv.2407.21783.

[83] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[84] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proc. ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, MI, USA, Jun. 2005, pp. 65–72.

[85] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, Barcelona, Spain, Jul. 2004, pp. 74–81.

[86] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguist. (ACL)*, Philadelphia, PA, USA, Jul. 2002, pp. 311–318, doi: 10.3115/1073083.1073135.

[87] B. Li and L. Han, "Distance weighted cosine similarity measure for text classification," in *Intell. Data Eng. Autom. Learn. (IDEAL 2013)*, *Lecture Notes in Computer Science*, vol. 8206, pp. 611–618, 2013, doi: 10.1007/978-3-642-41278-3_74.

[88] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with BERT," in *Proc. Int. Conf. Learn. Representations (ICLR)*, 2020.

[89] H. Kane, M. Y. Kocyigit, A. Abdalla, P. Ajanoh, and M. Coulibali, "NUBIA: NeUral based interchangeability assessor for text generation," in *Proc. 1st Workshop on Evaluating NLG Evaluation*, Dec. 2020, pp. 28–37.

[90] T. Ito, K. van Deemter, and J. Suzuki, "Reference-free evaluation metrics for text generation: A survey," *arXiv preprint arXiv:2501.12011*, Jan. 2025, doi: 10.48550/arXiv.2501.12011.
[91] D. C. Montgomery, *Statistical Quality Control: A Modern Introduction*, 6th ed. Hoboken, NJ, USA: Wiley, 2010.
[92] J. Cohen, *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed. New York, NY, USA: Psychology Press, 2009.
[93] Regulation (EU) 2018/848 of the European Parliament and of the Council of 30 May 2018 on organic production and labelling of organic products and repealing Council Regulation (EC) No 834/2007. Accessed: Jan. 21, 2026. [Online]. Available: http://data.europa.eu/eli/reg/2018/848/oj
[94] FAO, "Climate Smart Agriculture Sourcebook," Accessed: Jan. 21, 2026. [Online]. Available: https://www.fao.org/climate-smart-agriculture-sourcebook/en/
[95] Convention on Biological Diversity, "The Convention on Biological Diversity," Accessed: Jan. 07, 2026. [Online]. Available: https://www.cbd.int/convention
[96] European Commission, "Biodiversity Strategy for 2030," Accessed: Jan. 07, 2026. [Online]. Available: https://environment.ec.europa.eu/strategy/biodiversity-strategy-2030_en