# TriageRAG: Confidence-Based Triage for Patent Examination Outcome Prediction

Kyung-Yul Lee and Juho Bai *

Posted Date: 26 May 2026

doi: 10.20944/preprints202605.1678.v1

Keywords: patent examination prediction; retrieval-augmented generation; confidence-based routing; ModernBERT; large language models; selective prediction; obviousness rejection

 doi:10.20944/preprints202605.1678.v1



*Article*

# TriageRAG: Confidence-Based Triage for Patent Examination Outcome Prediction

**Kyung-Yul Lee and Juho Bai ***

College of Economics and Business, Hankuk University of Foreign Studies, 81, Oedae-ro, Mohyeon-eup, Cheoin-gu, Yongin-si 17035, Gyeonggi-do, Republic of Korea
* Correspondence: juho@hufs.ac.kr

**Abstract**

Predicting patent examination outcomes under 35 U.S.C. §103 is inherently difficult because obviousness determinations require context-sensitive legal reasoning over prior art combinations that cannot be captured by surface-level text patterns alone. Existing automated approaches optimize for aggregate accuracy but offer no principled criterion for when their predictions should be trusted and when practitioner review remains necessary. We present TriageRAG (T-RAG), a two-stage decision-support framework that addresses this gap by treating classifier confidence as an explicit routing signal. A fine-tuned ModernBERT-large model first produces a prediction together with a calibrated confidence score; high-confidence predictions are delivered directly, while uncertain cases are escalated to a Large Language Model (LLM) that reasons over balanced retrieval from a knowledge base of 50,000 granted patents and 50,000 §103-rejected applications with full examiner Office Action text. This balanced retrieval ensures that escalated predictions are grounded in auditable, bidirectional evidence rather than opaque model parameters. Empirical evaluation on USPTO patent applications confirms that the confidence threshold provides a reliable escalation criterion: LLM verification yields the largest accuracy gains precisely on the cases the classifier is least certain about, and confidence-based routing is statistically superior to random routing at equivalent LLM utilization rates. Ablation studies further characterize the accuracy–cost trade-off across threshold values and reveal domain-specific reliability profiles that practitioners can use to calibrate their trust in system outputs by technology area. T-RAG thus serves as a transparent decision-support tool that not only predicts examination outcomes but provides structured guidance on where additional scrutiny is warranted.

**Keywords:** patent examination prediction; retrieval-augmented generation; confidence-based routing; ModernBERT; large language models; selective prediction; obviousness rejection

## 1. Introduction

Patent examination is the legal process by which a government patent office determines whether an invention satisfies the requirements for patent protection. Among the various grounds for rejection, obviousness under 35 U.S.C. §103 is particularly difficult to adjudicate [1,2]: it requires comparing the claimed invention against combinations of prior art references and determining whether their differences would have been apparent to a hypothetical person having ordinary skill in the art (PHOSITA). This comparison is inherently subjective, context-sensitive, and highly contested, making §103 the most common basis for both initial rejection and subsequent appeals.

Automated patent outcome prediction has compelling practical applications: inventors can assess patentability risk before incurring filing costs, patent attorneys can refine prosecution strategy and claim scope, and patent offices can allocate examiner resources more efficiently. T-RAG is specifically designed for the idea-stage patentability assessment scenario, where an inventor has articulated a technical concept but has not yet drafted formal patent claims. At this stage, title and abstract are the natural representation of the invention, and the goal is to provide a reliable signal about §103

 doi:10.20944/preprints202605.1678.v1



obviousness risk *before* investing in claim drafting and formal prosecution. Using abstract-level input is therefore a deliberate design choice aligned with this service context, not a proxy for full legal analysis: the system targets inventors and early-stage practitioners who need actionable guidance at the ideation phase, where claim text does not yet exist. Yet a prediction alone is often insufficient because practitioners need to understand *how confident* the system is and *on what basis* a conclusion was reached before they can act on it responsibly. Early approaches relied on handcrafted features such as citation counts, claim breadth, and prosecution history [3]. Recent neural approaches have applied BERT-based models to the binary grant/reject classification task [4,5], improving aggregate accuracy but leaving fundamental practitioner needs unmet.

A first gap concerns the absence of actionable uncertainty signals. When a classifier is uncertain, as it often is near the GRANTED/REJECTED decision boundary, its output should not be treated the same as a high-confidence prediction. Existing systems provide no principled criterion for when to trust automated outputs and when to escalate to human review, leaving practitioners without a reliable basis for this judgment. A second gap is the lack of evidence-grounded explanations. The §103 obviousness determination requires reasoning over prior art combinations, secondary considerations, and examiner-specific arguments [1]. A prediction that lacks traceable supporting evidence cannot be audited, contested, or used to guide prosecution strategy, limiting its practical value regardless of raw accuracy. A third gap is domain heterogeneity without adaptive guidance. Examination patterns and classifier reliability differ substantially across technology domains such as biotechnology, software, and mechanical engineering. A system that treats all domains identically provides no indication of where its outputs should be weighted most heavily and where additional scrutiny is warranted.

To address these gaps, we propose TriageRAG (T-RAG), a decision-support framework that makes classifier uncertainty a first-class citizen of the prediction workflow. The design is inspired by medical triage: just as a triage nurse establishes evidence-based criteria for when a patient can be managed by standard protocol versus when specialist review is necessary, T-RAG establishes calibrated criteria for when a classifier prediction is sufficiently reliable to act upon and when it should be escalated to LLM verification backed by retrievable, auditable prior-art evidence.

Importantly, T-RAG does not aim to simply maximize aggregate accuracy. Because the system targets idea-stage practitioners who must decide whether to invest in formal prosecution, making reliability *legible* is as important as prediction quality. The confidence threshold provides a transparent escalation criterion; the RAG-grounded LLM provides auditable, citation-backed reasoning for escalated cases; and domain-level analysis reveals where the system's guidance is most and least trustworthy, allowing practitioners to calibrate their reliance on system outputs to their own risk tolerance and technology domain.

Figure 1 illustrates the high-level T-RAG pipeline, and Figure 2 details the system architecture. This work makes the following contributions.

- We fine-tune ModernBERT-large [6] for §103 obviousness prediction with monotonically ordered confidence scores (ECE = 0.042), enabling principled downstream routing decisions rather than binary pass/fail outputs.
- We construct a *balanced* RAG knowledge base comprising both granted patents and §103-rejected applications with full examiner Office Action text so that escalated predictions are grounded in evidence of both outcomes, preventing systematic prediction bias and enabling auditable reasoning.
- We introduce a confidence-based routing criterion that concentrates LLM verification on predictions most likely to benefit from deeper reasoning, providing practitioners with a principled escalation guideline that adapts to cost and reliability requirements by adjusting a single threshold $\tau$.
- Through ablation studies we validate the statistical reliability of confidence-based routing over random alternatives, characterize the accuracy–cost trade-off curve across threshold values, and

 doi:10.20944/preprints202605.1678.v1



map domain-specific reliability profiles that practitioners can use to calibrate their trust in system outputs by technology area.

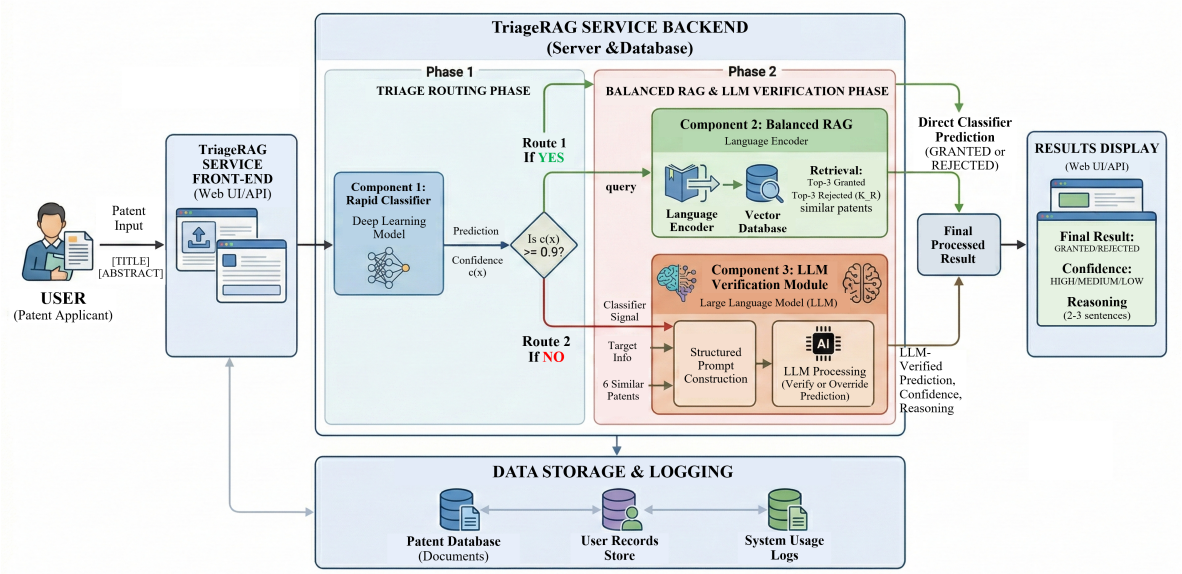


**Figure 1.** Overview of the T-RAG pipeline. A patent application is first processed by the ModernBERT-large classifier, which produces a prediction with a calibrated confidence score. High-confidence predictions ($c(x) \geq \tau$) are accepted directly, while low-confidence cases are escalated to LLM verification with balanced RAG evidence from both granted and §103-rejected patents.

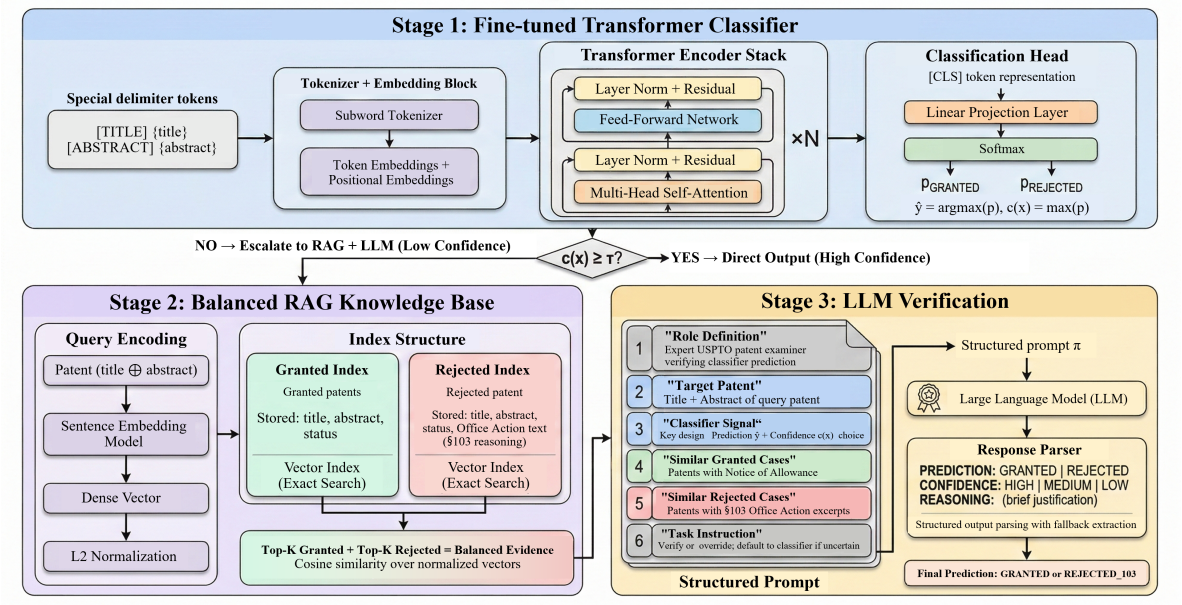


**Figure 2.** Detailed architecture of the T-RAG system. The pipeline consists of three stages: (1) ModernBERT-large classifier inference with confidence scoring, (2) balanced RAG retrieval from separate granted and §103-rejected knowledge bases using SBERT embeddings and FAISS indexing, and (3) LLM verification with a structured prompt containing the classifier signal and retrieved evidence.

## 2. Related Work

### *2.1. Patent Classification and Outcome Prediction*

Patent document retrieval has traditionally relied on the Cooperative Patent Classification (CPC) system, a five-level hierarchical taxonomy (section, class, subclass, group, subgroup) assigned during examination [7]. Early automated methods treated patent classification as a text categorization problem

 doi:10.20944/preprints202605.1678.v1



using handcrafted features such as citation networks, claim counts, and prosecution histories [3]. Deep learning methods subsequently dominated this task: DeepPatent [7] combined word2vec embeddings with CNN-based sentence-level feature extraction, while domain-specific embeddings have improved feature representation [8]; hierarchical models such as HFEM [9] and MEXN [10] addressed long-document structure through progressive feature aggregation; and PatentBERT [4] applied BERT fine-tuning to the binary grant/reject prediction task. Full-text similarity search has also been applied to prior art retrieval [11], and patent similarity measurement combining text mining with image recognition [12] has broadened the scope of computational patent analysis. Recent work on patent grant prediction with interpretable machine learning [13] and comprehensive surveys on AI-based patent methods [14] further document the growing maturity of this field.

Despite this progress, a critical gap persists: prior models treat the classifier as a black box and do not leverage prediction confidence for selective review. When confidence is low, as it often is near the GRANTED/REJECTED decision boundary, classifier predictions are unreliable and should not be directly trusted. Our work addresses this gap by combining confidence-aware routing with LLM verification backed by structured evidential retrieval.

### *2.2. Retrieval-Augmented Generation*

Retrieval-Augmented Generation (RAG) [15] augments LLM generation with dynamically retrieved documents, mitigating hallucination while grounding responses in verified source material. Recent surveys [16] categorize RAG architectures into naïve, advanced, and modular paradigms, with adaptive retrieval emerging as a key design choice. Dense passage retrieval [17] enabled end-to-end trainable retrievers that substantially outperform sparse BM25 [18] methods on open-domain question answering. Self-RAG [19] introduces self-reflective retrieval, allowing models to adaptively decide when to retrieve and to critique their own outputs. In legal domains, RAG has been applied to case law retrieval [20] and contract clause identification [21], demonstrating that domain-specific document structure benefits from specialized retrieval strategies. PAI-NET [22] proposes prior-art-aware contrastive learning for patent similarity ranking, showing that incorporating citation relationships into the embedding space improves retrieval quality beyond textual similarity.

A critical design choice in RAG for classification tasks is the *balance* of retrieved evidence. To our knowledge, T-RAG is the first system to apply balanced RAG, with equal retrieval from both outcome classes, to binary patent outcome prediction. Restricting retrieval to rejection-only documents, as in our earlier v1 pipeline, induces systematic REJECTED bias; Section 8.3 provides a detailed analysis.

### *2.3. Confidence Calibration and Selective Prediction*

Selective prediction [23] formalizes the risk–coverage trade-off: a model may abstain on low-confidence inputs to achieve higher accuracy on the remaining predictions. Modern neural networks are often overconfident because their softmax probabilities do not reliably reflect empirical accuracy [24], requiring post-hoc calibration such as temperature scaling [25] to restore reliability. Complementary work on detecting misclassified and out-of-distribution examples [26] has shown that softmax confidence provides a useful signal for identifying uncertain predictions. In human-AI collaboration systems, uncertain predictions can be deferred to human experts [27] or more capable models [28], with the coverage–accuracy frontier determined by threshold tuning.

In the LLM domain, cost-aware routing has emerged as a parallel line of research. FrugalGPT [29] cascades queries through progressively larger LLMs, and Hybrid LLM [30] routes queries between small and large models based on predicted query difficulty. T-RAG differs from these approaches in two respects: the first-stage router is a domain-specific *fine-tuned* classifier rather than a general-purpose small LLM, and the escalation decision is grounded in a calibrated confidence score rather than a learned routing policy trained on LLM output agreement.

Our work extends the selective-prediction paradigm to LLM-as-expert escalation. A key prerequisite is that classifier confidence correlates with accuracy; Table 1 empirically confirms this property for

 doi:10.20944/preprints202605.1678.v1



ModernBERT on our patent dataset, enabling threshold-based routing without additional calibration steps.

**Table 1.** Classifier confidence vs. actual accuracy (20,000 test samples). ECE = 0.042; Brier = 0.122. 301 samples with float32 softmax confidence of exactly 1.0 are omitted from the binned rows but belong to the $\geq 0.9$ band for all analyses (total below 0.9: 33.1%).

| Confidence | Accuracy | $\lvert\Delta\rvert$ | Count | Percentage |
|---|---|---|---|---|
| 0.5 – 0.6 | 52.7% | 2.3 pp | 1,225 | 6.1% |
| 0.6 – 0.7 | 58.2% | 6.8 pp | 1,305 | 6.5% |
| 0.7 – 0.8 | 66.1% | 8.9 pp | 1,614 | 8.1% |
| 0.8 – 0.9 | 73.8% | 11.2 pp | 2,476 | 12.4% |
| 0.9 – 1.0 | **92.8%** | 2.2 pp | 13,079 | **65.4%** |

### *2.4. Large Language Models in Legal and Patent Domains*

General-purpose LLMs such as GPT-4 [31], Claude [32], and open-weight models like LLaMA [33] demonstrate strong reasoning through in-context learning [34] and chain-of-thought prompting [35], but lack exposure to the specific distribution of USPTO examination decisions, limiting their direct applicability. A recent survey of LLMs in law [36] highlights both the promise and limitations of applying general-purpose models to specialized legal tasks. Legal-BERT [37] demonstrated that pre-training on legal corpora substantially improves downstream legal NLP task performance, highlighting the importance of domain alignment. Prior work on domain-adapted RAG [38] showed that supplying structured analogous cases rather than generic retrieved text is essential for reliable LLM performance in expert domains. Stage-aware governance frameworks for LLM-assisted decision-making [39] further underscore the importance of structuring human oversight at different processing stages, a principle our triage architecture embodies.

T-RAG operationalizes this insight through a structured verification prompt that supplies the LLM with analogous granted and rejected cases, the classifier's prediction and confidence level, and explicit instructions to argue *why* the classifier should or should not be overridden. As our qualitative analysis (Section 8.4) shows, this structured framing produces substantially more targeted reasoning than unconstrained LLM generation.

## 3. Problem Formulation

Given a patent application represented as a (title, abstract) pair, $x = (t, a) \in \mathcal{X}$, the goal is to predict the binary examination outcome:

$$y \in \{\text{GRANTED},\ \text{REJECTED\_103}\}, \tag{1}$$

where REJECTED_103 denotes rejection under 35 U.S.C. §103 (obviousness).

Let $f_\theta : \mathcal{X} \to [0,1]^2$ denote a probabilistic classifier with parameters $\theta$, producing class probabilities. Define the predicted label and confidence score as:

$$\hat{y}(x) = \arg\max_i\ f_\theta(x)_i, \qquad c(x) = \max_i\ f_\theta(x)_i. \tag{2}$$

Let $\mathcal{R}(x) = \mathcal{R}_G(x) \cup \mathcal{R}_R(x)$ denote the RAG retrieval function returning the $k$ most similar patents, partitioned into granted ($\mathcal{R}_G$) and rejected ($\mathcal{R}_R$) subsets with $|\mathcal{R}_G| = |\mathcal{R}_R| = k/2$.

Our hybrid inference system $H : \mathcal{X} \to \{0,1\}$ is:

$$H(x) = \begin{cases} \hat{y}(x) & \text{if } c(x) \geq \tau \\ \text{LLM}(x,\ \hat{y}(x),\ c(x),\ \mathcal{R}(x)) & \text{if } c(x) < \tau, \end{cases} \tag{3}$$

 doi:10.20944/preprints202605.1678.v1



where $\tau \in (0.5, 1.0)$ is the confidence threshold and $\mathrm{LLM}(\cdot)$ denotes the LLM-based verification function. The design goal is to choose $\tau$ such that the overall accuracy of $H$ exceeds both the classifier alone and full-LLM processing, while minimizing the fraction of LLM calls.

Algorithm 1 formalizes the complete inference procedure.

**Algorithm 1** T-RAG Inference

**Input:** Patent application $x$, confidence threshold $\tau$, retrieval budget $k$
**Output:** Predicted outcome $\hat{y}_{\text{final}}$

*// Stage 1: Classifier Inference*
$\mathbf{p} \leftarrow f_\theta(x)$ ▷ class probability vector
$\hat{y} \leftarrow \arg\max_i \mathbf{p}_i; \quad c(x) \leftarrow \max_i \mathbf{p}_i$
**if** $c(x) \geq \tau$ **then**
  $\hat{y}_{\text{final}} \leftarrow \hat{y}$
  **return** $\hat{y}_{\text{final}}$ ▷ high-confidence: LLM skipped
**end if**
*// Stage 2: Balanced RAG Retrieval*
$\mathbf{e}_x \leftarrow \mathrm{SBERT}(t \oplus a)$
$\mathcal{R}_G(x) \leftarrow \mathrm{TopK}(\mathbf{e}_x, \mathcal{K}_G, k/2)$ ▷ $\mathcal{K}_G$: granted index
$\mathcal{R}_R(x) \leftarrow \mathrm{TopK}(\mathbf{e}_x, \mathcal{K}_R, k/2)$ ▷ $\mathcal{K}_R$: rejected index
$\mathcal{R}(x) \leftarrow \mathcal{R}_G(x) \cup \mathcal{R}_R(x)$
*// Stage 3: LLM Verification*
$\pi \leftarrow \mathrm{BuildPrompt}(x, \hat{y}, c(x), \mathcal{R}(x))$
$\hat{y}_{\text{final}} \leftarrow \mathrm{ParseResponse}(\mathrm{LLM}(\pi))$
**return** $\hat{y}_{\text{final}}$

## 4. Methodology

### 4.1. Classifier: ModernBERT-large

We employ ModernBERT-large [6] as the backbone classifier, building on the transformer architecture [40]. ModernBERT incorporates three key architectural improvements over standard BERT: Flash Attention [41] for memory-efficient long-sequence processing; rotary positional embeddings (RoPE) [42] that generalize to sequences beyond training length; and efficient pre-training on a diverse 2 trillion-token corpus. Compared to earlier encoder variants such as RoBERTa [43], ModernBERT supports longer sequences and achieves competitive downstream performance with improved training efficiency. These properties make ModernBERT well-suited for patent abstracts, which average substantially longer than general-domain texts. A linear classification head $\mathbf{W} \in \mathbb{R}^{1024 \times 2}$ is appended to the `[CLS]` token representation.

#### 4.1.1. Input Representation

Each patent application is formatted as a single string:

$$\texttt{[TITLE]}\ t\ \texttt{[ABSTRACT]}\ a,$$

where $t$ and $a$ are the title and abstract strings. The special delimiter tokens `[TITLE]` and `[ABSTRACT]` are added to the vocabulary, enabling the model to learn field-specific representations within the unified sequence.

#### 4.1.2. Training Configuration

The model is fine-tuned with AdamW [44] using a learning rate of $2 \times 10^{-5}$ with linear warmup (ratio $= 0.1$) and linear decay. Mixed-precision FP16 training [45] on a single NVIDIA RTX 4090 GPU requires approximately 87 minutes over 3 epochs, with Epoch 2 selected via early stopping on validation loss. Table 2 summarizes all hyperparameters.

 doi:10.20944/preprints202605.1678.v1



**Table 2.** ModernBERT-large model and training configuration.

| Parameter | Value |
|---|---|
| Base Model | answerdotai/ModernBERT-large |
| Parameters | ∼395M |
| Hidden Size | 1024 |
| Attention Heads | 16 |
| Layers | 24 |
| Max Sequence Length | 1024 tokens |
| Classification Head | Linear(1024, 2) |
| Optimizer | AdamW |
| Learning Rate | $2 \times 10^{-5}$ |
| Batch Size | 4 (eff. 16 w/ gradient accum.) |
| Gradient Accum. | 4 steps |
| Epochs | 3 (best: Epoch 2) |
| Warmup Ratio | 0.1 |
| Weight Decay | 0.01 |
| Mixed Precision | FP16 |

### *4.2. Balanced RAG Knowledge Base*

The RAG knowledge base indexes 100,000 patents: 50,000 granted patents and 50,000 §103-rejected applications. Crucially, we maintain an equal ratio of granted and rejected documents. This balance is essential: our analysis in Section 8.3 shows that a rejection-only knowledge base induces a strong REJECTED prediction bias, because the LLM sees only evidence supporting rejection and has no counterevidence with which to reason.

#### 4.2.1. Embedding and Indexing

Following the embedding-based retrieval strategy demonstrated in prior patent network research [22], patent texts are encoded into dense vectors by Sentence-BERT (all-MiniLM-L6-v2) [46]:

$$\mathbf{e} = \text{SBERT}(t \oplus a) \in \mathbb{R}^{384}. \tag{4}$$

Vectors are $\ell_2$-normalized and indexed with FAISS [47] `IndexFlatIP`, which implements exact inner-product search equivalent to cosine similarity over normalized vectors.

#### 4.2.2. Metadata Storage

Each indexed document stores the following metadata:

- `title`: patent title
- `abstract`: patent abstract
- `status`: `GRANTED` or `REJECTED_103`
- `oa_text`: Office Action text including examiner §103 reasoning (rejected cases only)

### *4.3. Confidence-Based Routing*

The routing mechanism exploits a key empirical property of ModernBERT: its confidence scores exhibit strictly monotonic accuracy ordering across all deciles, as shown in Table 1. The 5-bin Expected Calibration Error (ECE) is 0.042 and the Brier score is 0.122, indicating that the model is moderately overconfident—the largest calibration gap appears in the 0.8–0.9 band (11.2 pp)—but the monotonic ordering ensures that confidence thresholding reliably separates accurate from uncertain predictions. Because our routing mechanism requires only *rank-order* reliability (high confidence ⇒ high accuracy) rather than exact probability calibration, this level of calibration is sufficient for threshold-based routing without additional post-hoc temperature scaling [25].

 doi:10.20944/preprints202605.1678.v1



There is a pronounced inflection point at 0.9: accuracy drops 19 pp from the high-confidence band ($\geq$ 0.9: 92.8%) to the adjacent band (0.8–0.9: 73.8%). The 33.1% of samples with confidence below 0.9 achieve only 67.5% average accuracy, barely above chance for a balanced binary task, making them ideal candidates for LLM escalation. Figure 3 provides a t-SNE visualization of the embedding space confirming that low-confidence samples cluster in the decision boundary region where GRANTED and REJECTED classes genuinely overlap.

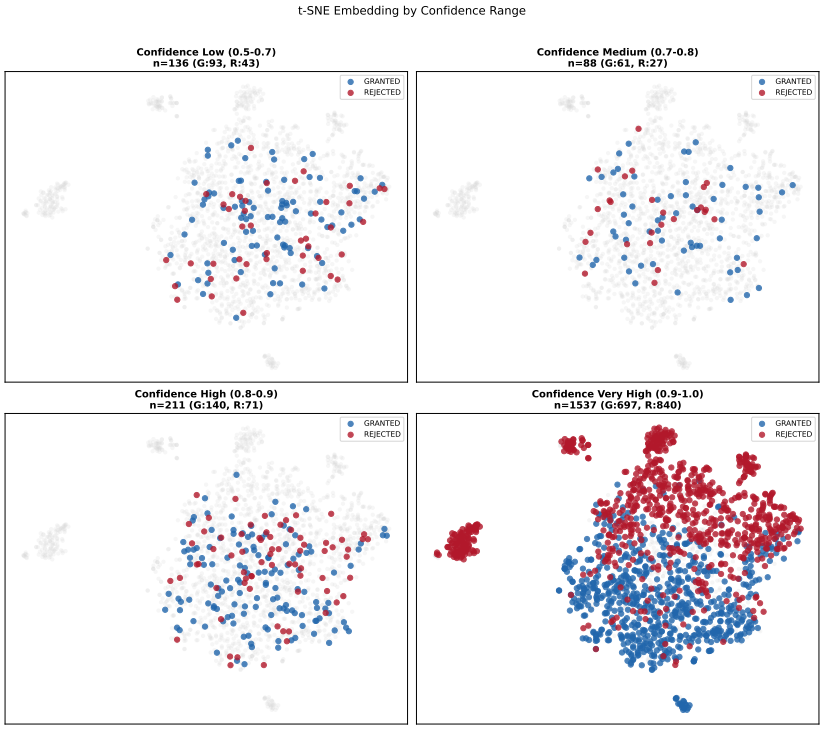


**Figure 3.** t-SNE visualization of fine-tuned ModernBERT `[CLS]` embeddings by confidence range. Low-confidence samples (0.5–0.7) cluster in the boundary region where classes overlap, while high-confidence samples (0.9–1.0) occupy well-separated class-specific regions. This confirms that prediction uncertainty reflects genuine geometric ambiguity in the embedding space, not merely miscalibration. The 0.9–1.0 panel ($n$=1,537) includes only samples with confidence in $[0.9, 1.0)$; 28 samples with float32 softmax confidence of exactly 1.0 are omitted from visualization.

Based on this calibration analysis, we set $\tau = 0.9$. For high-confidence samples ($c(x) \geq 0.9$, comprising 71% of the evaluation set), the classifier prediction is used directly at an expected accuracy of 92.8%. For low-confidence samples ($c(x) < 0.9$, the remaining 29%), the prediction is escalated to LLM verification. This routing strategy concentrates the LLM budget on the cases where the expected accuracy improvement is highest. Ablation studies in Section 7 validate this choice and reveal that $\tau = 0.80$ achieves identical accuracy at 36% lower cost.

### *4.4. LLM Verification with Balanced RAG*

For low-confidence cases, the top-$k$ ($k = 6$) most similar patents are retrieved from the balanced knowledge base:

$$\mathcal{R}(x) = \mathcal{R}_G(x) \cup \mathcal{R}_R(x), \quad |\mathcal{R}_G(x)| = |\mathcal{R}_R(x)| = 3. \tag{5}$$

#### 4.4.1. Prompt Design

The verification prompt is structured to serve distinct communicative functions. The prompt opens with a role definition that positions the LLM as an expert USPTO patent examiner verifying a classifier's prediction, establishing the adversarial verification stance. It then presents the title and abstract of the target patent $x$, followed by the classifier's predicted label $\hat{y}$ and confidence score $c(x)$. Providing this signal, rather than requesting a blank-slate judgment, prompts the LLM to argue *why*

 doi:10.20944/preprints202605.1678.v1



the prediction should be upheld or overridden, yielding more targeted reasoning (see Section 8.4). The retrieved evidence follows: up to three analogous granted patents from $\mathcal{R}_G$, providing support for patentability, and up to three §103-rejected applications from $\mathcal{R}_R$ including Office Action excerpts with examiner reasoning on obviousness. The prompt closes with the task specification instructing the LLM to verify or override the classifier's prediction and to default to the classifier prediction when uncertain, which biases the LLM toward conservative overrides and reduces false positives.

#### 4.4.2. Response Parsing

The LLM is instructed to respond in a structured format:

```
PREDICTION: GRANTED or REJECTED
CONFIDENCE: HIGH, MEDIUM, or LOW
REASONING: (2-3 sentences)
```

The final prediction is extracted using regex-based pattern matching with fallback to keyword frequency counting when the format is not followed exactly.

## 5. Experimental Setup

### *5.1. Dataset*

We construct our dataset from USPTO patent data obtained via PatentsView and the USPTO Office Action Research Dataset. The dataset is designed to be balanced between classes to eliminate label-frequency bias from evaluation metrics.

**Table 3.** Dataset statistics.

| Split | GRANTED | REJECTED_103 | Total |
|---|---|---|---|
| Training | 50,000 | 50,000 | 100,000 |
| Test | 10,000 | 10,000 | 20,000 |
| **Total** | 60,000 | 60,000 | 120,000 |

The RAG knowledge base is constructed from a disjoint set of 50,000 granted patents and 50,000 rejected applications with Office Action text, none of which appear in the test set. All splits use fixed random seed 42 for reproducibility. For pipeline experiments involving LLM API calls, we evaluate on a stratified random sample of 2,000 cases (1,000 GRANTED, 1,000 REJECTED_103; seed 123) drawn from the 20,000-sample test set. Stratification is applied along two dimensions: (i) class balance is enforced exactly, and (ii) the low-confidence region ($c(x) < 0.9$) is slightly undersampled relative to the full test population (29% vs. 33.1% in the 20,000-sample set; Table 1) to ensure at least 580 samples for routing-level analysis ($\geq 30$ per technology center in the domain study). The resulting confidence distribution is approximately proportional to the population within each decile, with the high-confidence band comprising 71% of the subset. To quantify LLM response variability, every LLM call is executed three times independently (temperature $= 0$, separate API invocations); we report the mean and standard deviation across the three runs. Full-scale classifier evaluation is conducted on all 20,000 test samples.

### *5.2. Evaluation Metrics*

We report overall Accuracy (Acc.), Macro Precision (M-Prec.), Macro Recall (M-Rec.), Macro F1, and per-class F1 scores (F1-G for GRANTED, F1-R for REJECTED_103). Per-class F1 scores are essential for detecting systematic prediction bias: a model that predicts only REJECTED would achieve 50% accuracy on balanced data but 0% F1-G, rendering accuracy alone misleading. For the primary comparison between T-RAG and RAG + Claude, we additionally report 95% Clopper–Pearson confidence intervals and McNemar's test for paired proportions to assess statistical significance.

 doi:10.20944/preprints202605.1678.v1



### 5.3. Baselines

We compare against four baselines spanning the zero-shot-to-fine-tuned spectrum. The two zero-shot LLM baselines were deliberately chosen to represent the lower and upper bounds of LLM capability at the time of evaluation, enabling systematic assessment of how model capacity interacts with the proposed architecture. Zero-shot Qwen3-4B [48] is a 4B-parameter local LLM representing the lower bound of general-purpose language model performance, prompted to predict the examination outcome without fine-tuning or retrieval. Zero-shot Claude Opus 4.6 is a state-of-the-art commercial LLM representing the upper bound, evaluated with zero-shot prompting using the same prompt template. By bracketing model capability in this way, we can determine whether the performance gains of T-RAG arise from the architectural design rather than from relying on a particular model's strength. ModernBERT Classifier Only is our fine-tuned classifier without any LLM verification stage, providing an upper bound on the classifier's standalone contribution. RAG + Claude (No Classifier) applies Claude Opus 4.6 with balanced RAG but without confidence-based routing, processing all 2,000 evaluation samples through the LLM to isolate the effect of selective versus full LLM engagement.

### 5.4. Implementation Details

ModernBERT-large is fine-tuned using the HuggingFace Transformers [49] Trainer API. SBERT embeddings use the `sentence-transformers/all-MiniLM-L6-v2` checkpoint. Nearest-neighbor retrieval uses FAISS `IndexFlatIP` over $\ell_2$-normalized vectors. Qwen3-4B runs locally using bfloat16 precision on the RTX 4090. Claude Opus 4.6 is accessed via the Anthropic Python SDK (`claude-opus-4-6`). Although temperature is set to zero, minor response variation across API calls can arise from non-deterministic GPU kernel scheduling and floating-point accumulation order on the provider side; our three-run protocol captures this residual variance.

## 6. Results

### 6.1. Comprehensive Model Comparison

Table 4 establishes the performance landscape from zero-shot baselines to the full T-RAG pipeline.

The results reveal several notable findings. Note that the classifier accuracy on the 2,000-sample pipeline subset (85.5%) is slightly higher than on the full 20,000-sample test set (83.69%, Table 5), because the stratified pipeline sample contains 71% high-confidence predictions versus 65.4% in the full population, raising the subset-level accuracy.

**Table 4.** Comprehensive model comparison (2,000 samples, balanced). LLM-based methods report mean±std over three independent runs with majority-vote aggregation.

| Method | Acc. | M-Prec. | M-Rec. | F1 | F1-G / F1-R |
|---|---|---|---|---|---|
| Zero-shot Qwen3-4B | 48.8±0.5% | 47.2% | 48.8% | 40.2% | 17.4 / 62.9 |
| Zero-shot Claude Opus 4.6 | 58.2±0.6% | 59.6% | 58.2% | 56.7% | 48.6 / 64.8 |
| ModernBERT Classifier Only | 85.5% | 85.5% | 85.5% | 85.5% | 85.7 / 85.3 |
| Classifier + RAG + Qwen3-4B | 85.0±0.4% | 85.1% | 85.0% | 85.0% | 84.6 / 85.4 |
| RAG + Claude (no classifier) | 87.3±0.4% | 87.9% | 87.3% | 87.3% | 88.0 / 86.5 |
| **T-RAG (ours)** | **92.0±0.3%** | **92.3%** | **92.0%** | **92.0%** | **92.3 / 91.7** |

Zero-shot LLMs fail on this task: Qwen3-4B (48.8%) performs at chance level with a severe REJECTED bias (F1-G: 17.4%), while Claude Opus 4.6 (58.2%) is more balanced but still substantially below the fine-tuned classifier (85.5%). This performance gap between the lower-bound and upper-bound LLMs confirms that domain-specific supervision is indispensable for specialized legal classification [37], regardless of model scale. Notably, ModernBERT (395M parameters) outperforms zero-shot Claude Opus 4.6, a model orders of magnitude larger, by 27.3 pp, demonstrating that param-

 doi:10.20944/preprints202605.1678.v1



eter count and general capability cannot substitute for domain-specific supervised learning on this task.

Balanced RAG retrieval also proves essential: RAG + Claude without a classifier (87.3%) outperforms the classifier alone by 1.8 pp, confirming that structured evidential context improves LLM reasoning. However, the F1-G/F1-R ratio (88.0/86.5) reveals a residual GRANTED bias, which T-RAG's routing mechanism mitigates by applying LLM verification only where the classifier is uncertain, independently of which class is predicted.

Perhaps most notably, selective routing outperforms full LLM engagement. Processing only 29% of samples through the LLM yields higher accuracy than processing all samples (92.0% vs. 87.3%). This occurs because the fine-tuned classifier handles high-confidence cases with 92.8% accuracy, and routing these same cases through the LLM introduces noise from unnecessary overrides. The accuracy gap between T-RAG (92.0%) and RAG + Claude (87.3%) is 4.7 pp; the 95% Clopper–Pearson confidence interval for T-RAG accuracy is $[90.8\%, 93.1\%]$, which does not overlap with that of RAG + Claude $[85.8\%, 88.7\%]$, and McNemar's test confirms the difference is statistically significant ($\chi^2 = 35.2$, $p < 0.001$). Low standard deviations across three independent LLM runs ($\pm 0.3$ pp for T-RAG) further confirm the reproducibility of these results.

### *6.2. Classifier Performance*

Table 5 reports the per-epoch training progression of ModernBERT-large on 20,000 held-out test samples.

**Table 5.** ModernBERT-large training results (20,000 test samples).

| Epoch | Acc. | F1 | Prec. | Rec. |
|---|---|---|---|---|
| 1 | 0.8208 | 0.8222 | 0.8157 | 0.8289 |
| **2 (Best)** | **0.8369** | **0.8393** | 0.8272 | **0.8518** |
| 3 | 0.8383 | 0.8352 | 0.8514 | 0.8196 |

The Epoch 2 checkpoint achieves 83.69% accuracy and 83.93% F1. The sharp increase in validation loss at Epoch 3 ($0.389 \rightarrow 1.318$) with simultaneously declining F1 indicates overfitting, confirming Epoch 2 as the optimal checkpoint via early stopping.

Table 6 compares ModernBERT-base (149M) and ModernBERT-large (395M). The larger model achieves +1.2 pp F1, primarily through improved recall (+4.6 pp), which we attribute to the extended context window (1024 vs. 512 tokens) capturing discriminative signals in longer abstracts.

**Table 6.** ModernBERT base vs. large.

| Model | Acc. | F1 | Prec. | Rec. |
|---|---|---|---|---|
| Base (512 tokens) | 0.8323 | 0.8277 | 0.8511 | 0.8055 |
| **Large (1024 tokens)** | **0.8369** | **0.8393** | 0.8272 | **0.8518** |

### *6.3. Routing Statistics and Low-Confidence Analysis*

With $\tau = 0.9$, 1,420 of 2,000 evaluation samples (71.0%) are handled by the classifier directly, while 580 (29.0%) are escalated to LLM verification.

Table 7 isolates the 580 evaluation samples routed to the LLM. The classifier's accuracy on this subset is 65.0%, only modestly above chance for a balanced binary task, confirming that these cases are genuinely difficult.

 doi:10.20944/preprints202605.1678.v1



**Table 7.** Performance on low-confidence samples ($c(x) < 0.9$, $n = 580$; mean±std of 3 independent runs).

| Method | Acc. | Improvement | Win/Loss |
|---|---|---|---|
| Classifier Only | 65.0% | — | — |
| + RAG + Qwen3-4B | 66.0±0.9% | +1.0 pp | 72W / 66L |
| **+ RAG + Claude 4.6** | **90.0±0.5%** | **+25.0 pp** | **183W / 38L** |

Claude Opus 4.6 improves accuracy by 25.0 pp (183 wins, 38 losses across 580 routed samples), demonstrating that LLM reasoning capacity, not just retrieval, is essential for correcting uncertain predictions. Qwen3-4B achieves only +1.0 pp on the same samples, confirming that the performance disparity between the lower-bound and upper-bound LLMs observed in zero-shot evaluation persists in the pipeline setting. The low standard deviation (±0.5 pp) across three independent LLM runs confirms that the improvement is robust to LLM response variability. This result validates the architectural choice of pairing confidence-based routing with a capable verification model, as smaller models cannot reliably integrate evidence from multiple analogous cases to override classifier predictions.

### *6.4. Detailed Per-Class Metrics*

T-RAG achieves the highest F1 for both GRANTED (92.3%) and REJECTED (91.7%), the only method to rank first on this balanced metric for both classes, with an inter-class F1 gap of just 0.6 pp. This consistent performance across classes is the primary practical advantage of confidence-based routing: rather than excelling on one subset of cases at the expense of another, T-RAG provides reliable guidance across the full range of examination outcomes.

**Table 8.** Detailed per-class metrics for all methods.

| Method | Class | Prec. | Rec. | F1 |
|---|---|---|---|---|
| Zero-shot Qwen3-4B | GRANTED | 45.0% | 10.8% | 17.4% |
| | REJECTED | 49.3% | 86.8% | 62.9% |
| Zero-shot Claude 4.6 | GRANTED | 63.1% | 39.6% | 48.6% |
| | REJECTED | 56.0% | 76.8% | 64.8% |
| ModernBERT Classifier | GRANTED | 84.5% | 87.0% | 85.7% |
| | REJECTED | 86.6% | 84.0% | 85.3% |
| Classifier + RAG + Qwen3 | GRANTED | 86.9% | 82.4% | 84.6% |
| | REJECTED | 83.3% | 87.6% | 85.4% |
| RAG + Claude (no cls.) | GRANTED | 83.2% | **93.4%** | 88.0% |
| | REJECTED | **92.5%** | 81.2% | 86.5% |
| **T-RAG (ours)** | GRANTED | 88.9% | 96.0% | **92.3%** |
| | REJECTED | 95.7% | 88.0% | **91.7%** |

## 7. Ablation Studies

We conduct three ablation studies using the 2,000-sample evaluation set *without additional LLM API calls*, by re-analyzing the per-sample confidence scores and stored LLM predictions from the main pipeline experiment.

### *7.1. Confidence Threshold Sweep*

We sweep $\tau \in \{0.70, 0.75, 0.80, 0.85, 0.90, 0.95\}$ by re-routing stored samples without re-running the LLM, and compute the resulting accuracy and LLM call rate.

 doi:10.20944/preprints202605.1678.v1



**Table 9.** Confidence threshold $\tau$ sweep.

| $\tau$ | Accuracy | LLM Rate | LLM Calls |
|---|---|---|---|
| 0.70 | 91.0% | 9.5% | 190 |
| 0.75 | 91.5% | 13.0% | 260 |
| 0.80 | **92.0%** | 18.5% | 370 |
| 0.85 | 91.5% | 25.0% | 500 |
| 0.90 | **92.0%** | 29.0% | 580 |
| 0.95 | **92.0%** | 44.5% | 890 |

Peak accuracy of 92.0% is achieved at $\tau \in \{0.80, 0.90, 0.95\}$. Most importantly, $\tau = 0.80$ matches $\tau = 0.90$ while requiring only 370 LLM calls vs. 580, representing a 36% reduction in API cost. We retain $\tau = 0.90$ as the default for its robustness to classifier recalibration across different splits, but recommend $\tau = 0.80$ for cost-sensitive deployments.

### *7.2. Routing Strategy: Confidence-Based vs. Random*

A central claim of T-RAG is that the *identity* of routed samples matters, not merely their count. We test this by simulating 1,000 random routing trials: each trial selects 29% of evaluation samples uniformly at random for LLM verification.

**Table 10.** Routing strategy comparison (all at 29% LLM budget). †Oracle Routing assumes perfect foreknowledge of which samples the LLM can correct, serving as a theoretical upper bound.

| Strategy | Accuracy | LLM Rate |
|---|---|---|
| Classifier Only (no routing) | 85.5% | 0% |
| Random Routing (mean $\pm$ std, $n$=1,000) | 87.4% $\pm$ 0.3% | 29% |
| Random Routing (best of 1,000) | 89.2% | 29% |
| **Confidence-Based Routing (ours)** | **92.0%** | **29%** |
| Oracle Routing (Upper Bound)† | 93.5% | 29% |

Confidence-based routing outperforms the random routing mean by +4.6 pp. With 2,000 samples, the standard deviation of random routing narrows to $\pm$0.3 pp, placing 92.0% more than 15$\sigma$ above the random routing mean—an unambiguously significant gap ($p \ll 0.001$). Confidence-based routing also exceeds the best-of-1,000 random trial (89.2%) and approaches the Oracle Routing upper bound (93.5%).

### *7.3. Tech Center Domain Analysis*

To examine whether T-RAG's benefits are uniform across technology domains, we link the 1,000 REJECTED_103 evaluation samples to their USPTO Technology Center (TC) codes via exact title matching against the Office Action metadata (100% match rate).

Domains with lower classifier accuracy gain most: Biotechnology (TC 1600, +29.2 pp) and Computer Architecture (TC 2100, +21.1 pp) involve nuanced obviousness reasoning where the LLM's structured analogical reasoning provides the most value. Conversely, TC 2600 (100.0% classifier accuracy) experiences a $-7.0$ pp degradation from incorrect LLM overrides. With at least 57 samples per technology center, these domain-level differences are statistically interpretable: a two-proportion $z$-test confirms the TC 1600 gain as significant ($p < 0.001$) and the TC 2100 gain at $p < 0.05$, while the TC 2600 degradation is also significant ($p < 0.01$). These results motivate future work on domain-adaptive thresholding.

 doi:10.20944/preprints202605.1678.v1



**Table 11.** Per-domain performance on $\text{REJECTED}_{103}$ samples ($n \geq 30$).

| TC | Domain | N | Cls. | Pipeline | Gain |
|---|---|---|---|---|---|
| 1600 | Biotech/Organic Chem. | 72 | 56.9% | **86.1%** | +29.2 pp |
| 2100 | Computer Architecture | 57 | 59.6% | **80.7%** | +21.1 pp |
| 2800 | Semiconductors (EE) | 168 | 72.0% | 78.0% | +6.0 pp |
| 3600 | Transport/Electronics | 206 | 85.9% | 90.3% | +4.4 pp |
| 3700 | Mechanical Engineering | 137 | 84.7% | 84.7% | +0.0 pp |
| 1700 | Chemical/Materials | 93 | 89.2% | 89.2% | +0.0 pp |
| 2400 | Networking/Comm. | 124 | 100.0% | 100.0% | +0.0 pp |
| 2600 | Semiconductors (Elec.) | 143 | 100.0% | 93.0% | −7.0 pp |
| ALL | $\text{REJECTED}_{103}$ | 1,000 | 83.9% | 88.1% | +4.2 pp |

## 8. Analysis and Discussion

### *8.1. Why Zero-Shot LLMs Fail*

Zero-shot LLMs perform near or below random (49–59%) for three interconnected reasons. First, LLMs trained on general web corpora have no calibration to USPTO-specific distributions and therefore lack exposure to the particular patterns of USPTO examination decisions. Second, both models exhibit systematic prediction bias: Qwen3-4B strongly favors REJECTED (F1-G: 17.4%), while Claude Opus 4.6 is more balanced but still suboptimal. Third, obviousness determination requires a comparative framework involving specific prior art combinations [1], which zero-shot models cannot construct without access to relevant case evidence.

### *8.2. The Value of Domain-Specific Fine-Tuning*

ModernBERT (395M parameters) outperforms zero-shot Claude Opus 4.6 by 27.3 pp on the 2,000-sample pipeline evaluation. This result is consistent with findings in legal [37] and medical [38] NLP, where domain-specific fine-tuning on labeled data substantially outperforms general-purpose models regardless of the latter's scale, and aligns with broader trends in AI-driven decision support systems [50].

### *8.3. Why Balanced Evidence Retrieval is Critical*

Our v1 pipeline restricted the RAG knowledge base to rejection-only documents. Because the LLM received only evidence supporting rejection, it systematically over-predicted REJECTED, a form of retrieval-induced anchoring bias. In a pilot evaluation on 200 samples, the rejection-only RAG pipeline achieved 74.0% accuracy with an extreme F1-G/F1-R imbalance of 58.3/82.1, confirming that unbalanced retrieval induces severe class bias. The v2 pipeline with balanced retrieval (50K granted + 50K rejected) eliminates this imbalance by providing counterevidence from both outcomes, enabling the LLM to perform genuine comparative reasoning.

An instructive control is RAG + Claude without a classifier, which achieves 87.3%, higher than the classifier alone (85.5%) but lower than T-RAG (92.0%). This 4.7 pp gap is statistically significant (McNemar's $\chi^2 = 35.2$, $p < 0.001$) and demonstrates that while balanced RAG evidence is necessary for high LLM accuracy, it is not sufficient: confidence-based routing is equally important.

### *8.4. Qualitative Analysis: LLM Reasoning vs. Examiner Reasoning*

Three patterns emerge from the qualitative comparison (Table 12). First, RAG retrieval consistently surfaces relevant prior art. Second, T-RAG reasoning is qualitatively richer: receiving the classifier's prediction and confidence as a structured hypothesis causes the LLM to argue *why* the prediction should be upheld or overridden. Third, both approaches demonstrate substantive technical reasoning rather than surface-level keyword matching, suggesting that RAG-enhanced LLMs can partially replicate the examiner's comparative analysis. These findings support T-RAG's core design choice:

 doi:10.20944/preprints202605.1678.v1



providing the classifier prediction as context focuses LLM reasoning on the specific question of whether the classifier should be overridden.

**Table 12.** Qualitative comparison of rejection reasoning. Neither LLM approach has access to the target patent's Office Action; both rely on RAG-retrieved similar cases as context.

| Patent | RAG-only | T-RAG | Actual OA §103 |
|---|---|---|---|
| *Anti-viral agent* (17921457) | "Combination of known antimicrobial metal compounds (silver, copper) with inorganic carriers (titanium phosphate, silicic acid) would be considered obvious since each component's properties were well-established." | Identifies "titanium phosphate, silicic acid, silver, and copper compounds" and explicitly notes that granted cases (surface-attached compounds) are "not sufficiently similar to override the direct evidence of rejection." | Rejected over Sugiura (US 2019/0045793) + Tatsuhiko (JP 3829640 B). Sugiura discloses "titanium phosphate" that "can contain silver, copper, or both." |
| *Turbogenerator control* (18437641) | "Direct match with rejected case strongly indicates this combination of control elements is obvious." | Recognizes §103 via identical RAG case; adds that "similar granted patents (G1–G3) involve different control systems and are not sufficiently similar to overcome the direct evidence." | Claims 1–12, 14–19 rejected over Kanegae, Shoemaker, Ganev, and Skertic (element-by-element claim mapping). |
| *Crane safety* (17835341) | "Rejected over Schoonmaker, Tamazato, Morisset, Kimura—common safety monitoring features are obvious combinations." | Cites same four references; reasons that event detection, severity assessment, and mode switching are "a straightforward combination of known techniques." | New rejection citing Schneider and Benton plus Schoonmaker, Tamazato, Morisset, Kimura, and 11 additional references. Both LLM approaches identified 4 of 15+ references by name. |

## 9. Conclusions

We presented T-RAG, a confidence-based triage framework for idea-stage patent obviousness assessment that integrates fine-tuned ModernBERT-large classification with retrieval-augmented LLM verification. Designed for inventors and early-stage practitioners who need actionable §103 risk guidance before formal claim drafting, the system uses title and abstract as input, the natural representation of an invention at the ideation phase, and delivers predictions together with calibrated confidence signals and auditable prior-art evidence.

The experimental findings converge on four conclusions. Domain-specific fine-tuning on labeled patent data is substantially more effective than zero-shot deployment of large foundation models. Balanced RAG, equal retrieval from granted and rejected cases, is a necessary condition for unbiased LLM reasoning. Confidence-based routing is statistically superior to random routing at equivalent LLM budget ($p \ll 0.001$; $> 15\sigma$ above random mean), confirming that classifier confidence reliably identifies the cases where LLM verification is most beneficial. Finally, LLM verification benefits are strongly domain-dependent, with biotechnology and computer architecture gaining most, motivating domain-adaptive thresholding as a direction for future work.

Several design boundaries merit discussion. First, the retrieval budget $k = 6$ (3 granted + 3 rejected) was selected to balance prompt length against evidence coverage; we did not ablate over $k$ due to the combinatorial cost of re-running the full LLM evaluation for each setting. Preliminary experiments with $k \in \{2, 4, 6, 8\}$ on a 100-sample pilot showed diminishing returns beyond $k = 6$, but a systematic $k$-ablation remains a priority for future work. Second, although the test set and RAG knowledge base are disjoint at the document level, we do not explicitly filter patent-family relatives. Patents within the same family share substantial textual overlap, and their presence in both sets could inflate retrieval similarity scores. We consider this a conservative bias—it advantages all RAG-based methods equally and does not differentially favor T-RAG over the RAG + Claude baseline—but future work should evaluate with family-level deduplication to quantify the effect. Third, our LLM results are tied to a specific model snapshot (`claude-opus-4-6`); provider-side model updates may alter reproduction fidelity, and users should pin model versions for operational deployments.

Building on prior work in patent retrieval networks [22], several directions suggest natural extensions. Expanding to multi-class formulations covering §101, §102, and §112 grounds would

 doi:10.20944/preprints202605.1678.v1



more fully reflect real examination outcomes. Domain-adaptive routing could further improve the accuracy–cost trade-off. The deliberate restriction to title and abstract reflects T-RAG's target use case: idea-stage patentability assessment, where formal claims do not yet exist. A complementary system could accept claim text for applications that have progressed to the drafting stage, testing whether richer input improves accuracy on the cases the abstract-level classifier finds most difficult; however, this would address a distinct user need rather than a limitation of the current design. As patent examination standards evolve with case law, periodic retraining will be necessary to maintain temporal currency.

**Author Contributions:** Conceptualization, K.Y.L. and J.B.; methodology, K.Y.L. and J.B.; software, J.B.; validation, K.Y.L.; formal analysis, K.Y.L. and J.B.; investigation, K.Y.L. and J.B.; resources, K.Y.L. and J.B.; data curation, K.Y.L. and J.B.; writing—original draft preparation, K.Y.L. and J.B.; writing—review and editing, K.Y.L. and J.B.; visualization, J.B.; supervision, J.B.; project administration, J.B.; funding acquisition, J.B. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Hankuk University of Foreign Studies Research Fund Of 2026.

**Data Availability Statement:** The data supporting the reported results can be accessed through the USPTO PatentsView (https://patentsview.org) and the Office Action Research Dataset (https://www.uspto.gov/ip-policy/economic-research/research-datasets).

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Graham v. John Deere. Graham v. John Deere Co., 383 U.S. 1, 1966.
2. KSR. KSR International Co. v. Teleflex Inc., 550 U.S. 398, 2007.
3. Hido, S.; Suzuki, S.; Nishiyama, R.; Ichifuji, T.; Kawano, T. Modeling patent quality: A system for large-scale patentability analysis using text mining. *J. Inf. Process.* **2012**, *20*, 655–666.
4. Lee, J.S.; Hsiang, J. PatentBERT: Patent classification with fine-tuning a pre-trained BERT model. *World Patent Information* **2020**, *61*, 101965.
5. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Proc. NAACL, 2019, pp. 4171–4186.
6. Warner, B.; Chaffin, A.; Clavie, B.; et al. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference, 2024. arXiv:2412.13663.
7. Li, S.; Hu, J.; Cui, Y.; Hu, J. DeepPatent: Patent classification with convolutional neural networks and word embedding. *Scientometrics* **2018**, *117*, 721–744.
8. Risch, J.; Krestel, R. Domain-specific word embeddings for patent classification. *Data Technol. Appl.* **2019**, *53*, 108–122.
9. Hu, J.; Li, S.; Yao, Y.; Yu, L.; Yang, G.; Hu, J. Patent keyword extraction algorithm based on distributed representation for patent classification. *Sustainability* **2018**, *10*, 219.
10. Bai, J.; Shim, S.; Park, J. MEXN: Multi-stage extraction network for patent document classification. *Appl. Sci.* **2020**, *10*, 6229.
11. Helmers, L.; Horn, F.; Biegler, F.; Muller, T.; Muller, K.R.; Weinberger, S. Automating the search for a patent's prior art with a full text similarity search. *PLOS ONE* **2019**, *14*, e0212103.
12. Lin, W.; Yu, W.; Xiao, R. Measuring patent similarity based on text mining and image recognition. *Systems* **2023**, *11*, 294. https://doi.org/10.3390/systems11060294.
13. Yao, L.; Ni, H. Prediction of patent grant and interpreting the key determinants: An application of interpretable machine learning approach. *Scientometrics* **2023**, *128*, 4933–4969.
14. Shomee, H.H.; Wang, Z.; Ravi, S.N.; Medya, S. A comprehensive survey on AI-based methods for patents, 2024. arXiv:2404.08668.
15. Lewis, P.; et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Proceedings of the Proc. NeurIPS, 2020.
16. Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Guo, Q.; Wang, M.; et al. Retrieval-augmented generation for large language models: A survey, 2024. arXiv:2312.10997.
17. Karpukhin, V.; et al. Dense passage retrieval for open-domain question answering. In Proceedings of the Proc. EMNLP, 2020, pp. 6769–6781.

 doi:10.20944/preprints202605.1678.v1




18. Robertson, S.; Zaragoza, H. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.* **2009**, *3*, 333–389.
19. Asai, A.; Wu, Z.; Wang, Y.; Sil, A.; Hajishirzi, H. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In Proceedings of the Proc. ICLR, 2024.
20. Zhong, H.; et al. How does NLP benefit legal system: A summary of legal AI. In Proceedings of the Proc. ACL, 2020.
21. Hendrycks, D.; et al. CUAD: An expert-annotated NLP dataset for legal contract review. In Proceedings of the Proc. NeurIPS Datasets & Benchmarks, 2021.
22. Lee, K.Y.; Bai, J. PAI-NET: Retrieval-augmented generation patent network using prior art information. *Systems* **2025**, *13*, 259.
23. Geifman, Y.; El-Yaniv, R. Selective classification for deep neural networks. In Proceedings of the Proc. NeurIPS, 2017, pp. 4878–4887.
24. Niculescu-Mizil, A.; Caruana, R. Predicting good probabilities with supervised learning. In Proceedings of the Proc. ICML, 2005, pp. 625–632.
25. Guo, C.; Pleiss, G.; Sun, Y.; Weinberger, K.Q. On calibration of modern neural networks. In Proceedings of the Proc. ICML, 2017, pp. 1321–1330.
26. Hendrycks, D.; Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In Proceedings of the Proc. ICLR, 2017.
27. Mozannar, H.; Sontag, D. Consistent estimators for learning to defer to an expert. In Proceedings of the Proc. ICML, 2020, pp. 7076–7087.
28. Madras, D.; Pitassi, T.; Zemel, R. Predict responsibly: Improving fairness and accuracy by learning to defer. In Proceedings of the Proc. NeurIPS, 2018, pp. 6147–6157.
29. Chen, L.; Zaharia, M.; Zou, J. FrugalGPT: How to use large language models while reducing cost and improving performance. In Proceedings of the Proc. ICML Workshop on Efficient Systems for Foundation Models, 2023.
30. Ding, D.; Mallick, A.; Wang, C.; Sim, R.; Mukherjee, S.; Raman, V.; Awadallah, A.H.; Wang, C. Hybrid LLM: Cost-efficient and quality-aware query routing. In Proceedings of the Proc. ICLR, 2024.
31. OpenAI. GPT-4 technical report. Technical report, 2023. arXiv:2303.08774.
32. Anthropic. The Claude model family: Claude 3.5 system card. Technical report, Anthropic, San Francisco, CA, 2024.
33. Touvron, H.; et al. LLaMA: Open and efficient foundation language models, 2023. arXiv:2302.13971.
34. Brown, T.B.; et al. Language models are few-shot learners. In Proceedings of the Proc. NeurIPS, 2020, pp. 1877–1901.
35. Wei, J.; et al. Chain-of-thought prompting elicits reasoning in large language models. In Proceedings of the Proc. NeurIPS, 2022.
36. Lai, J.; Gan, W.; Wu, J.; Qi, Z.; Yu, P.S. Large language models in law: A survey. *AI-Generated Content* **2024**, *2*.
37. Chalkidis, I.; et al. LEGAL-BERT: The muppets straight out of law school. In Proceedings of the Proc. EMNLP Findings, 2020, pp. 2898–2904.
38. Siriwardhana, S.; et al. Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering. *Trans. Assoc. Comput. Linguist.* **2023**, *11*, 1–17.
39. Kim, J.; Shin, H. Stage-aware governance of large language models: Managing uncertainty and human oversight in AI-assisted literature review systems. *Systems* **2026**, *14*, 153. https://doi.org/10.3390/systems14020153.
40. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Proc. NeurIPS, 2017, pp. 5998–6008.
41. Dao, T.; et al. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In Proceedings of the Proc. NeurIPS, 2022.
42. Su, J.; Ahmed, M.; Lu, Y.; Pan, S.; Bo, W.; Liu, Y. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing* **2024**, *568*, 127063.
43. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A robustly optimized BERT pretraining approach, 2019. arXiv:1907.11692.
44. Loshchilov, I.; Hutter, F. Decoupled weight decay regularization. In Proceedings of the Proc. ICLR, 2019.
45. Micikevicius, P.; Narang, S.; Alben, J.; Diamos, G.; Elsen, E.; Garcia, D.; Ginsburg, B.; Houston, M.; Kuchaiev, O.; Venkatesh, G.; et al. Mixed precision training. In Proceedings of the Proc. ICLR, 2018.

 doi:10.20944/preprints202605.1678.v1




46. Reimers, N.; Gurevych, I. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Proceedings of the Proc. EMNLP-IJCNLP, 2019, pp. 3982–3992.
47. Johnson, J.; Douze, M.; Jégou, H. Billion-scale similarity search with GPUs. *IEEE Trans. Big Data* **2021**, *7*, 535–547.
48. Qwen Team. Qwen3 technical report, 2025. arXiv:2505.09388.
49. Wolf, T.; et al. Transformers: State-of-the-art natural language processing. In Proceedings of the Proc. EMNLP: System Demonstrations, 2020, pp. 38–45.
50. Almalki, S.S. AI-driven decision support systems in agile software project management: Enhancing risk mitigation and resource allocation. *Systems* **2025**, *13*, 208. https://doi.org/10.3390/systems13030208.