Iterative Prompt Refinement via Knowledge Alignment: A Case Study in Systematic Review Screening

Adrian Kuiper
Delft University of Technology

Abstract

Applying Large Language Models (LLMs) to high-stakes classification tasks like systematic review screening is challenged by prompt sensitivity and a lack of transparency. We introduce IMAPR (Iterative Multi-signal Adaptive Prompt Refinement), a novel framework where a single LLM uses its own internal signals to iteratively refine its prompts, improving classification robustness and reliability. Unlike black-box optimizers that tune the prompts using only external scores, IMAPR is a white-box approach that diagnoses why a prediction failed using three internal signals: model confidence, a rationale, and a knowledge alignment score that checks whether the evidence cited in the rationale actually covers the user-defined inclusion criteria. We evaluate IMAPR on a real-world biomedical screening task, comparing it against strong baselines including GPO and StraGo. IMAPR outperforms the best baseline (GPO) by 8.8% in Macro-F$_1$ while maintaining high, stable recall across runs. Across seven LLMs, IMAPR yields an average 9.2% improvement in Macro-F$_1$. An ablation shows that knowledge-alignment acts as a recall safeguard: removing it leaves Macro-F$_1$ similar but degrades recall, reducing reliability for screening. These results suggest that diagnostic, signal-driven prompt refinement is a practical alternative to black-box optimization for transparent, dependable LLM screening systems.

1 Introduction

Systematic reviews are essential for rigorous research across numerous fields. In high-stakes domains like evidence-based medicine, they are particularly critical as they form the foundation for clinical guidelines and healthcare policy (Moosapour et al., 2021). However, their early-stage screening process remains a major bottleneck. Reviewers manually inspect thousands of titles and abstracts to identify studies that satisfy strict inclusion criteria. This task is time-consuming, prone to error, and increasingly difficult to scale as the volume of biomedical literature continues to grow. As a result, the screening phase is labor-intensive, yet yields only a small fraction of relevant studies. For example, in a recent systematic review conducted at Erasmus Medical Center, 5,730 PubMed records were screened to select just 179 studies for inclusion.

Large language models (LLMs) present an appealing opportunity to automate this screening by framing it as a binary classification task (Khraisha et al., 2024; Blaizot et al., 2022). In this task, the model must predict whether each study is 'Relevant' for inclusion or 'Irrelevant' for exclusion based on its title and abstract. While promising, using LLMs for this high-stakes classification faces several challenges. First, predictions are highly sensitive to prompt phrasing, with small changes often resulting in large differences in model behavior (Sclar et al., 2024). Second, LLM decisions are not inherently explainable, making it difficult to assess whether classifications are based on the correct evidence (Bruynseels et al., 2025). This creates a risk of shortcut learning (Du et al., 2023): the model can latch onto superficial cues and even echo phrases from the prompt, rather than grounding its predictions in the content. Finally, inclusion criteria in medical screening tasks, such as whether a study is 'double-blinded' or 'placebo controlled', can be expressed in subtle, domain-specific ways that general-purpose models often miss. These challenges highlight the need for a new approach beyond simple prompt tuning, one that can instill and verify domain-specific reasoning.

In this paper, we introduce IMAPR, a modular framework that makes LLM-based classification more robust and interpretable. The core of IMAPR is its ability to function as a white-box optimizer around a single, fixed LLM (no fine-tuning), diagnosing the root cause of its own reasoning failures instead of merely observing a drop in performance.

To accomplish this sophisticated self-diagnosis, the framework leverages a trio of internal signals generated with each prediction: the model's confidence, a rationale (a model-generated natural-language explanation), and a knowledge alignment score that validates the rationale against domain-specific inclusion criteria. This process allows IMAPR to generate targeted, corrective edits to its own prompt.

Furthermore, we address the challenge of adaptation in a real-world, label-scarce environment. A key contribution of our design is a novel extension where the system trains a 'correctness oracle' on its own performance during the training phase. This oracle, a gradient-boosted model trained with gold labels on the training split using the internal-signal tuples as features, is used to predict whether the LLM's classification is likely flawed. This enables IMAPR to continue its refinement process during test-time, adapting to new data without requiring any additional human-provided labels. Our approach marks a significant shift from the dominant paradigm of Automatic Prompt Optimization (APO) (Pryzant et al., 2023). Most APO frameworks operate as black-box systems, using a single external performance score to guide refinement; IMAPR's diagnostic, signal-driven process provides a more transparent and targeted method for improvement.

We evaluate our method on a biomedical screening task using the Erasmus MC dataset introduced in §1. Our experiments include an ablation study to assess the contribution of our knowledge alignment component. Results show that IMAPR outperforms state-of-the-art black-box optimizers (Zhou et al., 2023; Tang et al., 2025; Wu et al., 2024) in overall Macro-F_1 score while maintaining high and stable recall, which is critical for reducing manual review load in a screening environment.

Our main contributions are:

- We introduce a diagnostic, white-box prompt-refinement method: a lightweight framework that uses multi-signal diagnostics around a single, fixed LLM (no fine-tuning) to apply targeted prompt edits.

- We present a learned correctness oracle, a gradient-boosted error predictor trained with gold labels at train-time on internal signal features, which at test-time uses its predicted error probability to gate when a rewrite is attempted, enabling label-free refinement.

- We introduce a knowledge alignment module and show via ablation that it acts as a recall safeguard: removing it keeps Macro-F1 similar but reduces recall from 0.962 ± 0.012 to 0.861 ± 0.231.

- We provide empirical validation and label efficiency: IMAPR outperforms strong APO baselines while maintaining high, stable recall (≥ 0.94) even with limited training data (see §4.3).

2 Related work

2.1 Automating Systematic Reviews with AI

Researchers have explored automating different stages of the review pipeline using artificial intelligence (AI). Early tools such as Rayyan, Covidence, and EPPI-Reviewer applied classical machine learning to prioritize records for manual screening. For instance, Rayyan uses support vector machines to rank abstracts based on reviewer feedback, reducing workload but often failing to fully capture complex inclusion criteria from limited training data (Valizadeh et al., 2022).

Building on these efforts, systems such as Research Screener (Chai et al., 2021) use deep learning with user-provided seed articles to rank abstracts and iteratively re-order the queue as more feedback arrives. While the tool achieves substantial workload reductions across multiple reviews, it remains a black-box ranking system that offers no explanations, depends on seed selection, and stops short of full-document classification.

More recently, large language models (LLMs) have been proposed as flexible alternatives due to their strong generalization capabilities. (Khraisha et al., 2024) investigated fine-tuned LLMs for abstract screening and found that performance was highly sensitive to prompt wording, highlighting the need for careful task-specific prompting. (Gartlehner et al., 2023) applied generative LLMs to data extraction, reporting competitive results with human annotators, but also noting issues with transparency and reliability.

Therefore, a key challenge remains: verifying whether LLM predictions are based on relevant evidence. (Smirnova et al., 2024) demonstrated that explanation techniques (XAI), paired with human rationales, can expose gaps in model reasoning. Inspired by this, our work addresses this gap by making the model's reasoning process transparent.

Its use of self-generated rationales and a knowledge alignment check provides a verifiable audit trail for each classification, which is essential for building trust in high-stakes medical applications. This entire classification and refinement process is achieved with a fixed LLM, which does not require model fine-tuning, ranking heuristics, or user-curated seed examples.

2.2 Prompt Engineering

Prompt design plays a critical role in the performance of large language models (LLMs), particularly in zero- and few-shot settings. Minor changes in phrasing, formatting, or task description can lead to large shifts in model behavior (Zhao et al., 2021; Lu et al., 2022). To improve robustness, several studies explore structured prompting strategies, including instruction tuning (Wang et al., 2022). Chain-of-thought prompting has also shown benefits for complex reasoning by decomposing tasks into intermediate steps (Wei et al., 2023).

Recent surveys have begun to frame these efforts within the broader discipline of Context Engineering (Mei et al., 2025), which distinguishes the art of prompt design from the science of building systems to optimize an LLM's inputs. This approach is characterized by its emphasis on the optimization process itself. While this can involve managing various inputs like retrieved knowledge or memory, a primary focus remains on engineering the system instructions themselves. Our work, IMAPR, contributes a novel engineering process for this instructional component, standing in contrast to the black-box methods described below.

To move beyond manually crafted static prompts, the dominant paradigm is Automatic Prompt Optimization (APO), where an LLM itself is employed as a black-box optimizer to refine prompts based on performance on a validation set. Early methods like Automatic Prompt Engineer (APE) (Zhou et al., 2023) perform a one-shot search, generating a pool of candidate prompts and selecting the single best performer. More recently, state-of-the-art frameworks use an iterative approach. GPO (Tang et al., 2025), for instance, draws an analogy to gradient-based optimization, using a trajectory of past prompts and their external performance scores to guide the refinement process. Addressing the common issue of prompt drifting, StraGo (Wu et al., 2024) analyzes both successful and failed examples from a validation set to generate an explicit, actionable strategy for the optimizer LLM. Despite their increasing sophistication, these methods fundamentally treat the task LLM as an opaque system, using a single external performance score as the sole signal for improvement.

Our work takes a different approach from this black-box paradigm. We introduce IMAPR, an interpretable, white-box framework that refines prompts by diagnosing why the prompt itself is failing. Instead of relying on an external performance metric, the evidence for IMAPR's diagnosis comes from a trio of internal signals generated alongside an incorrect prediction: the model's confidence score, a generated textual explanation, and a domain-specific knowledge alignment score. This allows for targeted, self-correcting prompt revisions that directly address flaws in the model's decision-making, rather than optimizing for an aggregate score.

2.3 Explanation and Feedback Loops in LLMs

Recent work explores frameworks where LLMs iteratively critique and refine their own outputs. ReAct combines chain-of-thought reasoning with action steps so an agent can revise earlier decisions during a multi-turn interaction (Yao et al., 2023). Reflexion adds a short memory of self-critiques that guides future turns and reduces repeated errors (Shinn et al., 2023). Self-Refine shows that even in single-turn tasks a model can answer, critique the answer, and rewrite it; two or three such cycles raise scores on summarisation, question-answering, and extraction without extra training data (Madaan et al., 2023). Skill-set optimisation variants push the same idea to few-shot domains, selecting transferable tools on the basis of prior errors (Nottingham et al., 2024).

These feedback loops focus on open-ended generation and assume either a running dialogue or a black-box reward signal, conditions that do not hold for single-turn, high-precision screening tasks. Closer to our setting, XCrowd uses per-instance crowd-sourced rationales to diagnose feature misuse and predict errors in relation extraction models (Smirnova et al., 2024). It does not automatically revise the model or prompts; improvements are left to future work.

In contrast, IMAPR adapts iterative refinement to explicitly target the system prompt for single-pass binary screening, where each document is considered independently. Instead of relying on external memory or reward models, IMAPR updates the prompt solely based on transparent domain-specific feedback signals derived directly from the model's predictions. By keeping the language-model weights fixed, IMAPR offers a transparent and lightweight refinement loop, generalizable across diverse systematic review domains.

3 IMAPR: Iterative Multi-signal Adaptive Prompt Refinement

We present IMAPR, a modular framework that improves LLM-based abstract screening through iterative prompt refinement. The process is triggered when a correctness oracle flags a prediction as incorrect. Once an error is identified, the framework diagnoses the failing prompt by providing the refiner module with three internal signals from the decision-making process: (1) the model's confidence score, (2) a generated rationale (natural-language explanation), and (3) a knowledge alignment score. This score is calculated by comparing user-supplied inclusion terms (*should-know*) against the evidence tokens the model cites in its rationale (*really-know*). Adapting to a new review topic only requires updating the inclusion-criteria list and replacing the initial prompt; no code changes are needed. Based on this diagnostic information, a new candidate prompt is generated and accepted only if it raises macro-F_1 and preserves recall on relevant abstracts. Throughout this process, all model weights remain fixed. The framework consists of four modules described below: Classifier (§3.1), Explainer (§3.2), Assessor (§3.3), and Prompt Refiner (§3.4). A complete overview of this framework and its components is illustrated in Figure 1.

3.1 Classifier

We use an LLM model as the screening classifier. Given the current prompt plus the title and abstract of the article, the model produces a binary label Relevant, Irrelevant. The log-probability assigned to the chosen label serves as a confidence score (Kauf et al., 2024) and is later fed to the assessor. When no refinement loop is applied, this single LLM call defines our baseline screening system.

3.2 Explainer

After the classification step, we issue a second, separately prompted call to the same LLM model. The call returns:

- **Free-text rationale** – short text explaining why the paper was judged relevant or irrelevant.
- **Evidence tokens** – the exact word pieces from the title or abstract that the model cites as evidence for its decision. We call this list the *really-know*.

To avoid "prompt echo" the evidence list is filtered to tokens that actually occur in the input. Any phrase that appears only in the prompt is discarded.

The rationale text is forwarded to the Prompt Refiner, while the *really-know* are passed to the alignment check in the Assessor.

3.3 Assessor

The assessor produces two signals, one for knowledge alignment and one for prediction correctness. The overall logic of this module during the training phase is summarized in Algorithm 1.

For **knowledge alignment**, the Assessor verifies if the model's reasoning is grounded in the required domain criteria. This is achieved with a single, structured LLM call that compares the evidence tokens from the Explainer (*really-know*) against a fixed set of user-defined inclusion criteria (*should-know*). These criteria are embedded within a system prompt that instructs the LLM to act as a "domain checker." The LLM receives the list of really-know as evidence and, in a single forward pass, evaluates whether each criterion is supported by that evidence. The model's output is a structured JSON object containing a boolean flag for each criterion, which directly serves as the alignment vector. The vector informs the Prompt Refiner of any specific criteria that were missed. The system prompt for this alignment task is provided in Appendix A.3.

For **correctness**, the system operates in two modes. During training, we use the available gold labels to determine if each classification is correct or incorrect. As we process the training data, we log the outcome of every decision, creating a new dataset where each entry consists of a feature vector and a ground-truth label. The feature vector combines numerical signals (e.g., model confidence, the alignment vector) with sentence-transformer embeddings of both the input text (title and abstract) and the prompt itself. The label for this vector is 1 if the LLM's classification was correct and 0 otherwise.

After the training phase is complete, this logged dataset is used to train a gradient-boosted tree

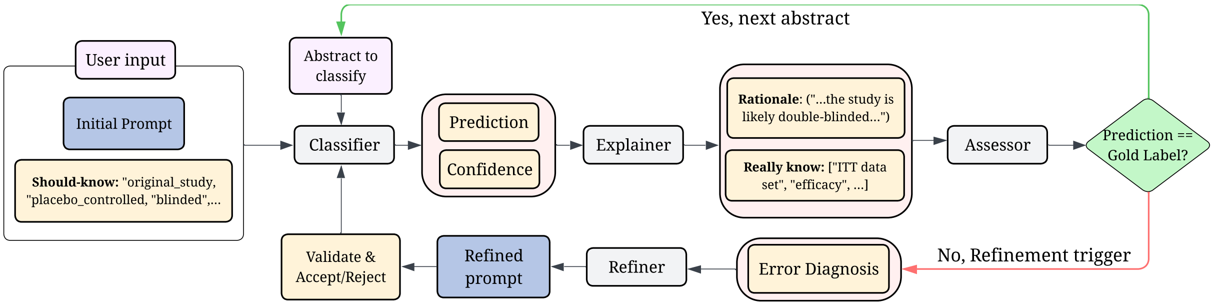
Yes, next abstract
User input
Initial Prompt
Should-know: "original_study, "placebo_controlled", "blinded",...
Abstract to classify
Classifier
Prediction
Confidence
Explainer
Rationale: ("...the study is likely double-blinded...")
Really know: ["ITT data set", "efficacy", ...]
Assessor
Prediction == Gold Label?
No, Refinement trigger
Error Diagnosis
Refiner
Refined prompt
Validate & Accept/Reject

Figure 1: An overview of the IMAPR framework. The flowchart illustrates the iterative refinement loop. For each abstract, the system generates a prediction and internal signals. A correctness check (using a gold label during training or a learned oracle at test-time) determines the path: correct predictions continue to the next sample (green path), while incorrect predictions trigger a prompt refinement cycle (red path).

Algorithm 1 Assessor (train-time)

```
procedure ASSESS
    Inputs: $\hat{y}$ (prediction), $y$ (gold), $c$ (confidence), $r$ (rationale), $E$ (really-knows), $S$ (should-knows)
    if $\hat{y} \neq y$ then
        $a \leftarrow \text{ALIGN}(E, S)$
        $m \leftarrow \text{MISSING}(a)$
        $d \leftarrow \text{SUMMARIZE}(c, r, m)$
        TRIGGERREFINE($d$)
    end if
end procedure
```

Algorithm 2 Prompt Refiner (acceptance policy)

```
procedure REFINE($p$)
    Inputs: rolling window $W$, diagnosis $d$, current prompt $p$
    $p' \leftarrow \text{LLMREFINE}(p, d)$
    $(\text{F1}, \text{Rec}) \leftarrow \text{EVAL}(p, W)$;
    $(\text{F1}', \text{Rec}') \leftarrow \text{EVAL}(p', W)$;
    if $\text{F1}' > \text{F1}$ and $\text{Rec}' \geq \text{Rec}$ then
        return $p'$                                    ▷ accept
    else
        return $p$                                     ▷ reject
    end if
end procedure
```

model to act as a correctness oracle. We selected this class of models for its strong performance in tabular data that combines diverse features (i.e., numerical scores and dense embeddings) and its computational efficiency. At test-time, this trained model replaces the need for gold labels. For each new abstract, it predicts the probability of the main classifier's decision being flawed. If this predicted error probability exceeds a threshold (0.5), the instance is deemed "likely-incorrect," and the prompt-refinement step is invoked.

3.4 Prompt refiner

When the assessor labels a prediction incorrect (training) or likely incorrect (test), the Prompt Refiner is activated. The distinction between these two trigger mechanisms is illustrated in Figure 2. This module provides the LLM with a meta-prompt containing the faulty prompt and the full error diagnosis (the confidence score, rationale, and alignment vector). The meta-prompt instructs the model to make a minimal, targeted edit to the prompt to address the specific failure identified by the diagnosis. The full meta-prompt for this refinement task is detailed in Appendix A.5.

The resulting candidate prompt is then evaluated on a rolling window of the last 50 abstracts screened, a window size chosen to balance a stable performance estimate with responsiveness to recent prompt changes. During training, the edit is accepted only if it increases the macro-F$_1$ score and preserves or improves recall for the Relevant class; otherwise, the edit is discarded and the previous prompt is restored. This acceptance policy is detailed in Algorithm 2.

At test-time, gold labels are unavailable, so the edit is kept when it reduces the fraction of instances that the correctness oracle flags as likely-incorrect.

This selective procedure allows IMAPR to continually refine its prompt while maintaining overall accuracy and stability.

4 Experiments

In this section, we describe the full experimental methodology used to evaluate our framework, IMAPR. Our evaluation is designed to answer the following research questions:

RQ1. *How does IMAPR perform compared to relevant state-of-the-art baselines?* To answer this, we compare IMAPR, which is refined on the training set using gold-label feedback, against black-box prompt-optimization methods that select prompts on a 500-abstract validation subset. All

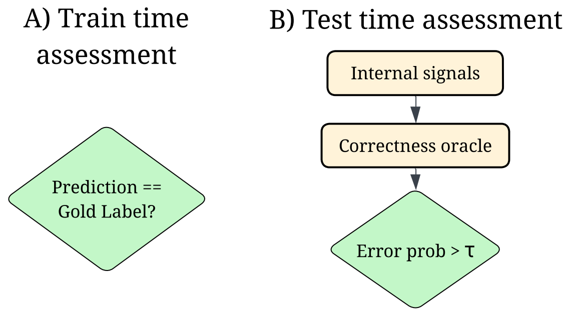

Figure 2: A comparison of the refinement trigger logic for training and test-time. (A) During training, the trigger is a direct mismatch between the model's prediction and the gold label. (B) During test-time, the trigger is a learned correctness oracle's predicted error probability exceeding a threshold (τ).

methods are then evaluated on the held-out test set.

RQ2. *What is the data efficiency of the IMAPR framework?* To answer this question, we evaluate IMAPR's performance in simulated low-resource settings with varying amounts of training data.

RQ3. *What is the contribution of the knowledge alignment component to the performance of IMAPR?* To answer this question, we conduct a targeted ablation study where we deactivate the knowledge alignment module to isolate its specific impact on the behavior of the framework.

RQ4. *How generalizable is IMAPR across different models?* We replicate our key experiments using several distinct Large Language Models to assess the framework's model-agnostic capabilities.

4.1 Experimental Setup

Dataset We evaluate IMAPR on a real-world biomedical corpus from a systematic review on placebo-controlled migraine trials, conducted at the Erasmus Medical Center (Erasmus MC). The corpus consists of 5,730 titles and abstracts retrieved from the PubMed database. The data set includes two sets of labels: initial "Stage 1" labels from the abstract screening and final "Stage 2" gold standard labels determined after a full-text review. For all experiments, we use the definitive Stage 2 labels as our ground truth. This choice is crucial as it tasks the model with predicting the final, correct outcome of the review, rather than mimicking the potentially noisy and error-prone intermediate screening process. Each record was annotated by human domain experts, resulting in 179 studies labeled *Relevant* and 5,551 labeled *Irrelevant* after screening. The 3.2% proportion of relevant studies reflects the significant class imbalance typically found in systematic review tasks.

For our experiments, we create a single, static 50/50 split of the data, stratified by label, to form a training set of 2,865 abstracts and a held-out test set of 2,865 abstracts. To evaluate data efficiency (RQ2), we create training subsets by randomly sampling 5%, 10%, 20%, 40%, and 100% of the full training set.

Large Language Models (LLMs) For our primary experiments, including the data efficiency and baseline comparisons, we use Meta's Llama 3.1 8B as the backbone model. This model was executed locally to ensure full control over the experimental environment. For all text generation steps, we set the temperature parameter to 0.0 to ensure deterministic outputs and support full reproducibility of these core results.

To assess the generalizability of our framework (RQ4), we then conduct comparative experiments with a diverse range of widely adopted models accessed via API. This includes other models from Meta's Llama family at various scales (LLaMA-3.2 3B, LLaMA-3.1 70B, LLaMA-3.3 70B, and LLaMA-3.1 405B), as well as models from other developers such as OpenAI's GPT-4o and DeepSeek-V3. For these API-based experiments, we also set temperature to 0.0 to minimize variance, though we note that full determinism cannot be guaranteed by API providers. This allows us to evaluate the method's transferability across diverse model architectures and capabilities.

Evaluation Protocol and Metrics. To measure the impact of our framework, we evaluate the performance of all comparison methods on a fixed, held-out test set. To ensure the stability and robustness of our findings, all reported results are the mean and standard deviation over five runs with different random seeds. Following best practices for high-stakes information retrieval tasks like medical screening, we select two primary evaluation metrics. Our most critical metric is Recall on the Relevant class ($\text{recall}_{\text{Rel}}$), as failing to identify a relevant study is significantly more costly than including an irrelevant one. For overall performance, we use the Macro-averaged F_1 score (Macro-F_1), which is well-suited for imbalanced datasets as it gives equal weight to both classes.

Implementation Details. Our IMAPR framework was configured with a rolling validation window of 50 abstracts and an error buffer of 30. Once the error buffer's threshold was met, the refinement was triggered using the diagnostic signals from the most recent error. These values were chosen empirically to ensure prompt updates are driven by stable error patterns rather than isolated mistakes. For the test-time refinement experiments, the correctness oracle was an `LightGBM` model, using features from the `all-MiniLM-L6-v2` Sentence-BERT model. All prompts used for IMAPR's modules are available in Appendix A and for the baselines in Appendix E.

4.2 Comparison Methods

To evaluate the effectiveness of our framework, we compare IMAPR against a hierarchy of baselines representing different levels of sophistication. We also analyze different variants of our own method to measure the impact of its learning stages.

Static Prompt Our primary baseline is the initial, human-engineered prompt. This prompt is fixed throughout the experiment and represents a standard zero-shot approach, measuring the raw capability of the LLM without any automated refinement.

APE (Automatic Prompt Engineer) As a representative one-shot optimization method discussed in Section 2.2, we implemented Automatic Prompt Engineer (APE) (Zhou et al., 2023). For our implementation, we prompted our local Llama 3.1 8B model with a meta-prompt containing three relevant and three irrelevant examples from the training set. We set the decoding temperature to 0.9 to generate a diverse pool of 50 candidate instructions. Each candidate was then scored on a fixed validation set of 500 abstracts, and the single prompt with the highest Macro-F_1 score was selected for the final comparison.

GPO (Gradient-inspired Prompt Optimizer) We compare IMAPR against GPO (Tang et al., 2025), a state-of-the-art iterative optimizer that uses a trajectory of past prompts and scores to guide refinement. In our implementation, we ran GPO for 12 iterations, generating 8 candidate prompts per iteration with a temperature of 0.9. To find the best prompt at each step, candidates were scored on a fixed validation set of 500 abstracts randomly sampled from the training data.

StraGo (Strategic-Guided Optimization) We also include StraGo (Wu et al., 2024), an advanced baseline designed to mitigate prompt drifting by generating explicit strategies from both successful and failed examples. For our faithful implementation, we ran the optimization for 5 iterations. At each step, we sampled 3 successful and 3 failed examples from a fixed 500-sample validation set. For each sampled example, the model generated multiple experiences (M = 3) to diagnose performance. Subsequently, a pool of candidate strategies (N = 3) was generated for each experience, with the best selected through a 5-pass LLM voting mechanism to guide the final prompt rewrite. The temperature for creative steps like strategy generation was set to 0.7.

IMAPR Variants. Finally, we evaluate two variants of our own method:

- **Train-refined:** This represents our core method and the primary configuration used for comparison against external baselines. The prompt is generated by running the full IMAPR refinement loop on the training set, using gold labels as the oracle.

- **Test-refined:** This represents an exploratory extension of our framework. It takes the final Train-refined prompt and allows IMAPR to continue adapting on the test set, using its trained correctness oracle for feedback to simulate a label-free deployment scenario.

4.3 Results

In this section, we present the empirical results of our experiments, which were designed to answer our four research questions. To ensure a fair and controlled comparison with our single-model framework, we configured all baselines to use the same Llama 3.1 8B instance for all their internal operations, including both task execution and prompt optimization. This choice isolates the effectiveness of the refinement methodology itself as the primary variable. We first analyze the data efficiency and generalizability of IMAPR, then compare its performance against our selected baselines.

Comparison with Baselines (RQ1) We compare the performance of our Train-refined IMAPR prompt against the hierarchy of external baselines in Table 1, with all results except the static prompt averaged over five runs. The results show that IMAPR significantly outperforms all baselines on the primary metric of Macro-F_1 score.

The automated baselines exhibit distinct and informative performance profiles. APE, despite slightly improving the Macro-F_1 score over the static prompt to 0.496 (±0.019), proves to be an unsuitable method for this task. Its recall for relevant studies is low (0.279) and unstable (±0.196), making it unreliable for a high-stakes screening environment. In contrast, StraGo successfully optimizes for the opposite objective, achieving a near-perfect and stable recall of 0.996 (±0.006). However, this comes at the expense of precision, resulting in a lower overall Macro-F_1 score of 0.458 (±0.024). GPO offers a more balanced improvement, increasing the Macro-F_1 score to 0.535 (±0.042) while maintaining a high, although less stable, recall of 0.937 (±0.097).

IMAPR distinguishes itself by achieving a state-of-the-art Macro-F_1 score of 0.582 (±0.018), a substantial improvement over the next best baseline (GPO). Importantly, it achieves this while maintaining a high and stable recall of 0.962 (±0.012). This demonstrates that by using a trio of internal, interpretable signals, IMAPR is able to navigate the precision-recall trade-off more effectively than black-box optimizers, delivering a solution that is both highly accurate and reliable for the critical task of systematic review screening.

Method	**Macro-F_1**	**Recall$_{\text{Rel}}$**
Static Prompt	0.439	0.989
APE	0.496 (±0.019)	0.279 (±0.196)
GPO	0.535 (±0.042)	0.937 (±0.097)
StraGo	0.458 (±0.024)	**0.996 (±0.006)**
IMAPR (Train-refined)	**0.582 (±0.018)**	0.962 (±0.012)

Table 1: Performance comparison of IMAPR against all external baselines on the full test set. All results except for the static prompt are the mean and standard deviation over five runs.

Data Efficiency and Stability (RQ2) To evaluate IMAPR's data efficiency, we refined prompts using subsets of the training data ranging from 5% to 100%. Table 2 summarizes the performance of these prompts (mean ± SD over five runs for each subset).

The results indicate two key findings. First, for train-refined prompts, the framework consistently learns a policy that prioritizes high recall, with mean $\text{Rec}_{\text{Rel}} \geq 0.94$ across all data subsets. This high-recall policy is especially evident at low data percentages, where the model learns a lenient, "safe" prompt. As more data is introduced, we observe a slight controlled decrease in mean recall, from 0.987 (5%) to 0.962 (100%), while Macro-F_1 increases (from 0.511 to 0.582). This reflects the system learning a more sophisticated trade-off between precision and recall: it proposes stricter prompts that improve precision at a small cost to recall, yielding better overall Macro-F_1. The process is most unstable at the 40% subset (SD ±0.079), suggesting that calibration of this trade-off is most sensitive to training-set composition at mid label budgets before stabilizing again on the full dataset. For completeness, test-refined prompts can have slightly lower recall (e.g., 0.939 at 100%) because label-free updates optimize a proxy objective rather than ground-truth recall.

Analysis of IMAPR Framework Properties To better understand the contributions of IMAPR's architectural components and its behavior in a simulated deployment setting, we conducted two targeted analyses.

The Role of Knowledge Alignment. To isolate the contribution of knowledge alignment, we conducted an ablation study that compared our entire framework against a variant with the knowledge alignment mechanism disabled. The study was performed on the full 100% training dataset, with results averaged over five runs.

As shown in Table 3, the impact on the overall Macro-F_1 score was minimal, with both configurations performing almost identically. However, the analysis reveals a difference in the recall on the relevant class. The full IMAPR framework maintains a stable and high recall of 0.962 (±0.01), while the ablated version's recall degrades substantially to a mean of 0.861 and exhibits high variance (±0.22). This finding indicates that the knowledge alignment score functions as a safeguard, preventing the model from sacrificing recall for precision, which is essential for the reliability of the system in a high-stakes screening environment.

Performance of Test-Time Refinement. Second, we evaluated the performance of IMAPR's test-time refinement mechanism, where the system adapts on the test set using its trained correctness oracle instead of gold labels. This label-free process generally improves the Macro-F_1 score across all data subsets, a trend visually summarized in Figure 3. However, a closer look at the individual runs reveals that the magnitude of this improvement is not uniform, which highlights that the efficacy of this process is conditional on the quality of the oracle. Full results, including the recall scores and

Data (%)	Train-Refined		Test-Refined	
	F_1	Rec_{Rel}	F_1	Rec_{Rel}
5%	0.511 (±0.033)	0.987 (±0.005)	0.512 (±0.032)	0.987 (±0.005)
10%	0.512 (±0.038)	0.984 (±0.010)	0.520 (±0.036)	0.984 (±0.010)
20%	0.487 (±0.013)	0.991 (±0.005)	0.520 (±0.025)	0.984 (±0.006)
40%	0.556 (±0.043)	0.948 (±0.079)	0.576 (±0.022)	0.942 (±0.072)
100%	**0.582 (±0.018)**	**0.962 (±0.012)**	**0.596 (±0.023)**	0.939 (±0.035)

Table 2: Macro-F_1 and $\text{Recall}_{\text{Rel}}$ on the fixed test set for prompts refined on varying amounts of training data. Results are the mean and standard deviation over five runs. The 'Test-Refined' columns show the final performance after allowing the trained oracle-guided refinement on the test set.

Configuration (100% Data)	Macro-F_1	$\text{Recall}_{\text{Rel}}$
Full IMAPR	0.582 (±0.018)	0.962 (±0.012)
w/o Knowledge Alignment	0.583 (±0.018)	0.861 (±0.231)

Table 3: Ablation study on the full training set (n=2,865). Removing the knowledge alignment component has a minimal impact on the overall Macro-F_1 score but severely degrades recall on the relevant class. Results are averaged over five runs.

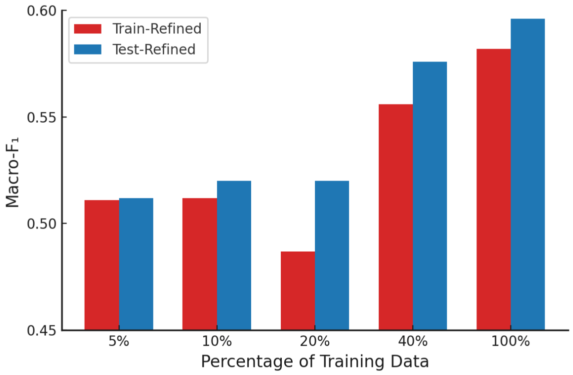

Figure 3: **Macro-F_1 vs. label budget.** Bars show mean performance over five runs for *train-refined* and *test-refined* prompts. Test-time refinement increases Macro-F_1 at every budget (largest at 20–40%). Exact values and recall appear in Table 2.

standard deviations, are available in Table 2.

To validate this, we analyzed the oracle's performance on its primary task: correctly identifying the main classifier's errors. Across our five runs, we observed a strong positive correlation between the oracle's recall on this error class and the performance gain from test-time refinement. (A detailed run-by-run breakdown is available in Appendix D).

Critically, even in runs where the oracle was less effective, the framework demonstrated high robustness. The system did not suffer severe performance degradation, indicating that the selective validation mechanism for new prompts effectively prevents the model from accepting harmful changes. Therefore, we conclude that IMAPR's test-time refinement is a robust mechanism whose success is directly coupled with the performance of its correctness oracle.

Generalizability Across Models (RQ4) To assess whether our framework is model-agnostic, we evaluated IMAPR against a static prompt using a diverse set of seven LLMs. The results, averaged over multiple runs, are presented in Table 4. For the majority of capable models (DeepSeek, and LLaMA models 70B and larger), IMAPR provides a clear and consistent improvement in Macro-F_1 over the static baseline while maintaining high recall. The final optimized prompts for the best-performing run of each model are provided in Appendix C to illustrate the concrete outputs of the refinement process. For instance, it improved the F_1 score of DeepSeek-V3 by +0.063. We also observe two interesting boundary cases. For the state-of-the-art GPT-4o, which already achieved a very high baseline, the gains were negligible, suggesting a ceiling effect. Conversely, the small LLaMA-3.2 3B model performed poorly in both conditions, indicating a floor of reasoning capability required for the framework to be effective.

4.4 Observed Prompt Edit Patterns

Beyond aggregate metrics, we analyse how prompts change during refinement. The edits do not merely rephrase instructions; they increase specificity and add verification logic that reduces ambiguity and unsupported inferences. Table ?? summarises recurring behaviours with examples; full before/after prompts are in Appendix C, and a step-by-step trace appears in Appendix B.

5 Discussion

The empirical results demonstrate that IMAPR's white-box, signal-driven approach to prompt re-

	Static prompt		Train-refined prompt (IMAPR)			
LLM	F$_1$	Rec$_{\text{Rel}}$	F$_1$	Rec$_{\text{Rel}}$	**Abs. Macro-F$_1$ Gain**	**% Macro-F$_1$ Gain**
LLaMA-3.2 3B	0.115 (±0.008)	1.000 (±0.000)	0.119 (±0.060)	1.000 (±0.000)	+0.004	+3.5%
LLaMA-3.1 8B*	0.439 (±0.000)	0.989 (±0.000)	0.582 (±0.018)	0.962 (±0.012)	**+0.143**	**+32.6%**
LLaMA-3.1 70B	0.559 (±0.008)	0.995 (±0.004)	0.568 (±0.010)	0.959 (±0.050)	+0.009	+1.6%
LLaMA-3.3 70B	0.571 (±0.006)	0.995 (±0.004)	0.598 (±0.026)	0.980 (±0.022)	+0.027	+4.7%
LLaMA-3.1 405B	0.561 (±0.034)	0.995 (±0.004)	0.627 (±0.014)	0.933 (±0.081)	+0.066	+11.8%
DeepSeek-V3	0.549 (±0.004)	0.991 (±0.004)	0.612 (±0.010)	0.959 (±0.020)	+0.063	+11.5%
GPT-4o	0.661 (±0.003)	0.951 (±0.035)	0.652 (±0.015)	0.980 (±0.018)	-0.009	-1.4%

Table 4: LLM-agnostic evaluation of the IMAPR framework. The table compares the Macro-F$_1$ and relevant-class recall of the static prompt against the IMAPR-refined prompt. All results are the mean and standard deviation over five runs. (*) The LLaMA-3.1 8B results are from local, deterministic runs, while its static baseline is from a single run. Refinement was performed on the full training set (n=2,865).

Emerging behaviour	**Example**
Specificity to limit ambiguity	"Does the abstract *explicitly* mention randomisation or allocation ratio?"
Domain heuristics / phrasing	"Check for terms like 'double-blind' or 'masked outcome assessor'."
Rules for missing information	"If 'placebo' is omitted, look for 'sham treatment' or 'vehicle control'; otherwise mark No."
Redundant emphasis on key checks	"Ensure migraine is the *primary* outcome" (repeated in checklist and final note).

Table 5: How prompts evolve through refinement. Each behaviour is illustrated with an excerpt from a refined prompt.

finement is more effective and reliable for systematic review screening than state-of-the-art black-box methods. Our framework achieved a superior Macro-F$_1$ score without compromising the high, stable recall that is non-negotiable in this high-stakes domain. This success is particularly noteworthy as our methodology used definitive Stage 2 gold-standard labels for training; a choice which avoided the detrimental optimizations that could arise from noisy intermediate data and instead challenged the framework to find the true signals of a study's final value from its abstract alone. This discussion interprets these findings, highlighting the mechanisms behind IMAPR's performance and situating its contribution within the broader research landscape.

The primary advantage of IMAPR stems from its ability to diagnose the reasoning process rather than only observing the final result. Black-box optimizers treat the language model as an opaque function, using a single, scalar performance score as the sole signal for improvement. This provides information that a prompt has failed, but not why. In contrast, IMAPR's trio of internal signals provides multifaceted diagnostic evidence. While model confidence helps flag uncertainty, we hypothesize the primary corrective power comes from the generated explanation and the knowledge alignment vector. Together, these signals provide rich, actionable insights into how the prompt's logic was flawed, enabling targeted revisions that directly address the reasoning failure. A detailed case study of this diagnostic and correction process, including a visual walkthrough of the framework in action, is provided in Appendix B.

The results of the ablation study for RQ3 clearly isolate the function of the knowledge alignment module as a critical "recall safeguard." Although removal had a negligible impact on the overall Macro-F$_1$ score, it caused the recall to degrade substantially and become unstable. This suggests that without an explicit mechanism to verify its reasoning against domain criteria, the optimizer is prone to sacrificing recall for precision. In applications like medical screening, where false negatives have severe consequences, such a safeguard is essential for building trustworthy and reliable systems.

Our generalizability analysis (RQ4) revealed the operational boundaries of the IMAPR framework. The lack of improvement on a 3B parameter model suggests a "floor" of reasoning ability is required for a model to successfully self-diagnose and refine. Conversely, the negligible gains on GPT-4o point to a potential "ceiling effect," where extremely large or highly-aligned models may benefit less from this type of self-correction. This positions IMAPR as a particularly valuable tool for enhancing the vast and growing ecosystem of powerful, moderately sized open-source models (e.g., in the 7B-70B class), which possess strong foundational capabilities but can still be significantly improved with targeted refinement.

Furthermore, our exploration of label-free test-time refinement provided insights into the limits of automated adaptation. The framework proved robust, as the selective validation mechanism prevented large performance drops even when oracle guidance was weak. Nevertheless, the effectiveness of this unsupervised loop is strongly dependent on oracle quality. This highlights a fundamental bottleneck for the field, because learning reliable proxies for ground-truth feedback remains difficult in high-stakes settings. A pragmatic path forward is a hybrid, human-in-the-loop workflow. This would involve using IMAPR's internal signals to drive active learning by flagging uncertain or low-alignment cases for review, periodically recalibrating the oracle, and falling back to the static prompt under high uncertainty. Such a system balances automation with expert oversight and supports continuous, auditable improvement.

Finally, this work contributes to the emerging discipline of Context Engineering, which moves beyond simple prompt design to the systematic optimization of an LLM's informational context. While other systems like Retrieval-Augmented Generation (RAG) engineer the external knowledge component of the context, IMAPR presents a novel, process-oriented system for engineering the LLM's internal instructional context. This white-box paradigm of self-diagnosis and correction represents a promising step towards building more robust, interpretable, and reliable LLM-based systems.

6 Limitations and Future Work

While our findings demonstrate the effectiveness of the IMAPR framework, this study has several limitations that present clear paths for future research.

First, the scope of our empirical evaluation was focused on a single, albeit high-impact, domain of biomedical abstract screening. While this provides a strong case study, future work should validate the generalizability of IMAPR to screening tasks in other knowledge-intensive domains where success depends on a clear set of inclusion criteria.

Second, the framework has several methodological dependencies. The knowledge alignment module, which proved crucial as a recall safeguard, requires a manually curated list of should-know from a domain expert for each new task. This semi-manual setup could be addressed in future work by exploring methods to automatically extract these key criteria from research protocols, moving towards a more fully autonomous system. Additionally, as discussed, the performance of the label-free, test-time refinement is entirely dependent on the quality of its correctness oracle. A promising direction for future research is to leverage the internal signals more directly in a human-in-the-loop system. For instance, predictions made with low confidence could be automatically flagged for human review, creating an efficient active learning workflow that optimally balances automation with expert oversight.

Our evaluation of the baselines was constrained to a single-model setup to ensure a fair comparison with IMAPR. The original publications for some of these methods (e.g., GPO, StraGo) utilize a more powerful model like GPT-4 as the prompt optimizer, and their performance might be higher under that configuration.

Finally, our core analysis was conducted on a capable 8B parameter model. While our generalizability study confirmed the framework's effectiveness across a range of model sizes, the dynamics of self-correction are likely dependent on model scale. As our results suggest, very small models may lack the requisite reasoning capacity for the diagnostic loop to be effective, while very large models may see diminishing returns from this type of refinement. A valuable direction for future research would be to systematically investigate how the quality of the diagnostic signals and the efficacy of the self-correction process scale with model size, which would help identify the optimal conditions for applying frameworks like IMAPR.

7 Conclusion

In this paper, we introduce IMAPR, a novel framework for iterative self-correcting prompt refinement designed to improve the reliability of LLM in the high-stakes task of systematic review screening. Unlike traditional black-box optimization methods that rely on external performance metrics, IMAPR operates as an interpretable, white-box system. Using a trio of internal signals, model confidence, a generated explanation, and a knowledge alignment score, the framework diagnoses and corrects flaws in its own reasoning process. Our experiments demonstrated that this signal-driven approach significantly outperforms strong baselines, achieving a superior balance between overall performance and the high, stable recall essential for the task. This work highlights the value of diagnostic, self-correcting mechanisms and represents a step towards building more transparent, robust, and effective LLM systems for critical, domain-specific applications.

References

Aymeric Blaizot, Sajesh Veettil, Pantakarn Saidoung, Carlos Moreno-García, Nirmalie Wiratunga, Magaly Aceves-Martins, Nai Ming Lai, and Nathorn Chaiyakunapruk. 2022. Using artificial intelligence methods for systematic review in health sciences: A systematic review. *Research Synthesis Methods*, 13.

Koen Bruynseels, Lotte Asveld, and Jeroen van den Hoven. 2025. “foundation models for research: A matter of trust?”. *Artificial Intelligence in the Life Sciences*, 7:100126.

Kevin Chai, Robin Lines, F. Gucciardi, and Leo Ng. 2021. Research screener: a machine learning tool to semi-automate abstract screening for systematic reviews. *Systematic Reviews*, 10.

Mengnan Du, Fengxiang He, Na Zou, Dacheng Tao, and Xia Hu. 2023. Shortcut learning of large language models in natural language understanding. *Preprint*, arXiv:2208.11857.

Gerald Gartlehner, L Kahwati, Rainer Hilscher, I Thomas, S Kugley, Karen Crotty, M Viswanathan, Barbara Nussbaumer-Streit, G Booth, N Erskine, Amanda Konet, and R Chew. 2023. Data extraction for evidence synthesis using a large language model: A proof-of-concept study.

Carina Kauf, Emmanuele Chersoni, Alessandro Lenci, Evelina Fedorenko, and Anna A. Ivanova. 2024. Log probabilities are a reliable estimate of semantic plausibility in base and instruction-tuned language models. *Preprint*, arXiv:2403.14859.

Qusai Khraisha, Sophie Put, Johanna Kappenberg, Azza Warraitch, and Kristin Hadfield. 2024. Can large language models replace humans in systematic reviews? evaluating <scp>gpt</scp>-4’s efficacy in screening and extracting data from peer-reviewed and grey literature in multiple languages. *Research Synthesis Methods*, 15(4):616–626.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2022. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *Preprint*, arXiv:2104.08786.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

Lingrui Mei, Jiayu Yao, Yuyao Ge, Yiwei Wang, Baolong Bi, Yujun Cai, Jiazhi Liu, Mingyu Li, Zhong-Zhi Li, Duzhen Zhang, Chenlin Zhou, Jiayi Mao, Tianze Xia, Jiafeng Guo, and Shenghua Liu. 2025. A survey of context engineering for large language models. *Preprint*, arXiv:2507.13334.

Hamideh Moosapour, Farzane Saeidifard, Maryam Aalaa, Akbar Soltani, and Bagher Larijani. 2021. The rationale behind systematic reviews in clinical medicine: a conceptual framework. *Journal of Diabetes Metabolic Disorders*, 20.

Kolby Nottingham, Bodhisattwa Prasad Majumder, Bhavana Dalvi Mishra, Sameer Singh, Peter Clark, and Roy Fox. 2024. Skill set optimization: reinforcing language model behavior via transferable skills. In *Proceedings of the 41st International Conference on Machine Learning*, ICML’24. JMLR.org.

Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. 2023. Automatic prompt optimization with "gradient descent" and beam search. *Preprint*, arXiv:2305.03495.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models’ sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. *Preprint*, arXiv:2310.11324.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

Alisa Smirnova, Jie Yang, and Philippe Cudre-Mauroux. 2024. Xcrowd: Combining explainability and crowdsourcing to diagnose models in relation extraction. pages 2097–2107.

Xinyu Tang, Xiaolei Wang, Wayne Xin Zhao, Siyuan Lu, Yaliang Li, and Ji-Rong Wen. 2025. Unleashing the potential of large language models as prompt optimizers: Analogical analysis with gradient-based model optimizers. *Preprint*, arXiv:2402.17564.

Amir Valizadeh, Mana Moassefi, Amin Nakhostin-Ansari, Seyed Hossein Hosseini Asl, Mehrnush Saghab Torbati, Reyhaneh Aghajani, Zahra Ghorbani, and Shahriar Faghani. 2022. Abstract screening using the automated tool rayyan: results of effectiveness in three diagnostic test accuracy systematic reviews. *BMC Medical Research Methodology*, 22.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Gary Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby

Kuznia, Krima Doshi, Maitreya Patel, and 21 others. 2022. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *Preprint*, arXiv:2204.07705.

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

Yurong Wu, Yan Gao, Bin Benjamin Zhu, Zineng Zhou, Xiaodi Sun, Sheng Yang, Jian-Guang Lou, Zhiming Ding, and Linjun Yang. 2024. Strago: Harnessing strategic guidance for prompt optimization. *Preprint*, arXiv:2410.08601.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Tony Z. Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. *Preprint*, arXiv:2102.09690.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2023. Large language models are human-level prompt engineers. *Preprint*, arXiv:2211.01910.

A IMAPR Framework Meta-Prompts

This appendix contains the full text of the initial (static) prompt and the key meta-prompts used by the IMAPR framework's modules.

A.1 Initial (Static) Prompt

This is the human-engineered prompt used as the starting point for refinement and as the "Static Prompt" baseline in our experiments.

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**

1. For each paper, evaluate the Title and
    Abstract according to the following five
    criteria:
  - Is it an **original study**? (Yes/No)
  - Is it **placebo-controlled**? (Yes/No)
  - Is it **double-blinded or
      triple-blinded**? (Yes/No)
  - Is it a **randomized clinical trial
      (RCT)**? (Yes/No)
  - Is the **main focus on migraine**? (Yes/No)

2. **Classification Rule:**
  - If **all five answers are "Yes"**,
      classify the paper as **"Relevant"**.
  - If **any answer is "No"**, classify the
      paper as **"Irrelevant"**.

3. **Return Format:** Return only one word:
    ‘"Relevant"‘ or ‘"Irrelevant"‘.
  Do not explain. Do not justify. Do not
      output anything else.
```

Listing 1: The initial system prompt for classification.

A.2 Explainer Meta-Prompt

The following template is used to construct the user message for the Explainer module. The placeholders in (*@<blue text>@*) are filled in by the system at runtime.

```
Using these criteria: <Full text of the system's
    classification prompt>

And this input; TITLE:
<Title of the paper being classified>

ABSTRACT:
<Abstract of the paper being classified>

You decided the paper is <Model's prediction
    (e.g., 'Relevant')>.
Please explain your decision using EXACT
    phrases and words from the text.
```

Listing 2: Template for the Explainer prompt.

A.3 Assessor (Knowledge Alignment) Meta-Prompt

The Assessor module uses the following system prompt to act as a "domain checker." It takes a list of ‘really-know‘ tokens from the user message and evaluates them against the should-know defined by the domain expert. In this case the should-know are related to the medical domain.

```
You are a domain checker.
Given a comma-separated list of evidence
    words/phrases (verbatim from the paper),
decide whether each of the five criteria is
    PRESENT or ABSENT in that evidence.

Criteria:
1. Original study     (PRESENT if evidence
    indicates the paper is a primary trial)
2. Placebo-controlled  (PRESENT if 'placebo' or
    synonyms appear)
3. Double / triple blinded (PRESENT if
    'double-blind', 'masked', etc.)
4. Randomized RCT      (PRESENT if 'randomized',
    'RCT', etc.)
5. About migraine treatment (PRESENT if
    'migraine' or synonym appears)

Return exactly this JSON schema:

{
  "original_study": true/false,
  "placebo_controlled": true/false,
  "blinded": true/false,
  "randomized": true/false,
  "migraine": true/false,
}

Return ONLY JSON schema
```

Listing 3: The system prompt for the knowledge alignment task.

A.4 Refiner Meta-Prompts

The Prompt Refiner module uses a two-step process. First, an ’Interpreter’ prompt synthesizes the raw diagnostic signals into a coherent paragraph. Second, a ’Refiner’ prompt uses this paragraph to generate the patched prompt.

A.4.1 Step 1: Interpreter System Prompt

This prompt instructs the LLM to summarize the raw signals from the ‘Assessor‘ into a natural language paragraph.

```
You are an assessment interpreter.

Write one concise paragraph (approx. 3-5
    sentences) that:

- States whether the last prediction was
    **Correct** or **Incorrect**, and
  whether the models confidence was **High**
      or **Low** (use those words).
- Mentions the alignment score (xx.xx) and
    whether you consider it High
  (>= 0.6) or Low (< 0.6).
- Lists every domain cue whose value is
    **false** in the domain_match dict,
  introduced with: Missing cues:   followed by
      the comma-separated list.
- Ends with a brief consequence, e.g. These
    missing cues likely explain the
  incorrect high-confidence prediction.

Return only that paragraphno bullet points, no
    JSON.
```

Listing 4: System prompt for the Assessment Interpreter.

A.4.2 Step 2: Refiner System and User Prompts

The main ’Refiner’ system prompt sets the rules for editing, while the user message provides the specific context for the edit, including the paragraph generated in Step 1.

```
You are a prompt refiner.

TASK
Take the ORIGINAL prompt shown below and
    produce a PATCHED version that
*still performs the same task* (binary Relevant
    vs Irrelevant screening of
migraine RCT abstracts) but corrects the
    specific weaknesses reported in the
assessment and explanation. Always modify the
    original prompt.

[+] Clarify or add one-line reminders about any
    cue missing in *domain_match*.
[-] Do **NOT** add new endpoints, output
    formats, or numerical calculations.
[-] Do **NOT** alter the one-word output
    requirement ("Relevant" or "Irrelevant").

OUTPUT FORMAT
Return **only** the final patched prompt textno
    commentary, no Markdown
fences and not the same prompt as the original.
```

Listing 5: System prompt for the Prompt Refiner.

```
ORIGINAL PROMPT:
<<<
<The original, failing prompt text>
>>>

ASSESSMENT PARAGRAPH:
<The assessment paragraph generated by the
    interpreter>

LLM Explanation (raw):
<The raw rationale from the Explainer module>

PATCH NOW:
```

Listing 6: Template for the Refiner user message.

B Case Study of a Single Refinement Cycle

This appendix provides a detailed, step-by-step example of a single IMAPR refinement cycle. Figure 4 offers a visual walkthrough of this process, illustrating how the core modules interact. Table ?? then breaks down the specific internal signals and reasoning at each stage of the same example, showing how the framework diagnoses a sophisticated error and applies a targeted correction.

C Final Optimized Prompts per LLM

This appendix shows the final optimized prompt for each LLM from our generalizability study (RQ4). As the optimal outcome requires balancing both performance and safety, we present the prompt from the run that achieved the highest Macro-F_1 score while maintaining a $\text{Recall}_{\text{Rel}}$ of at least 0.95.

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**
```

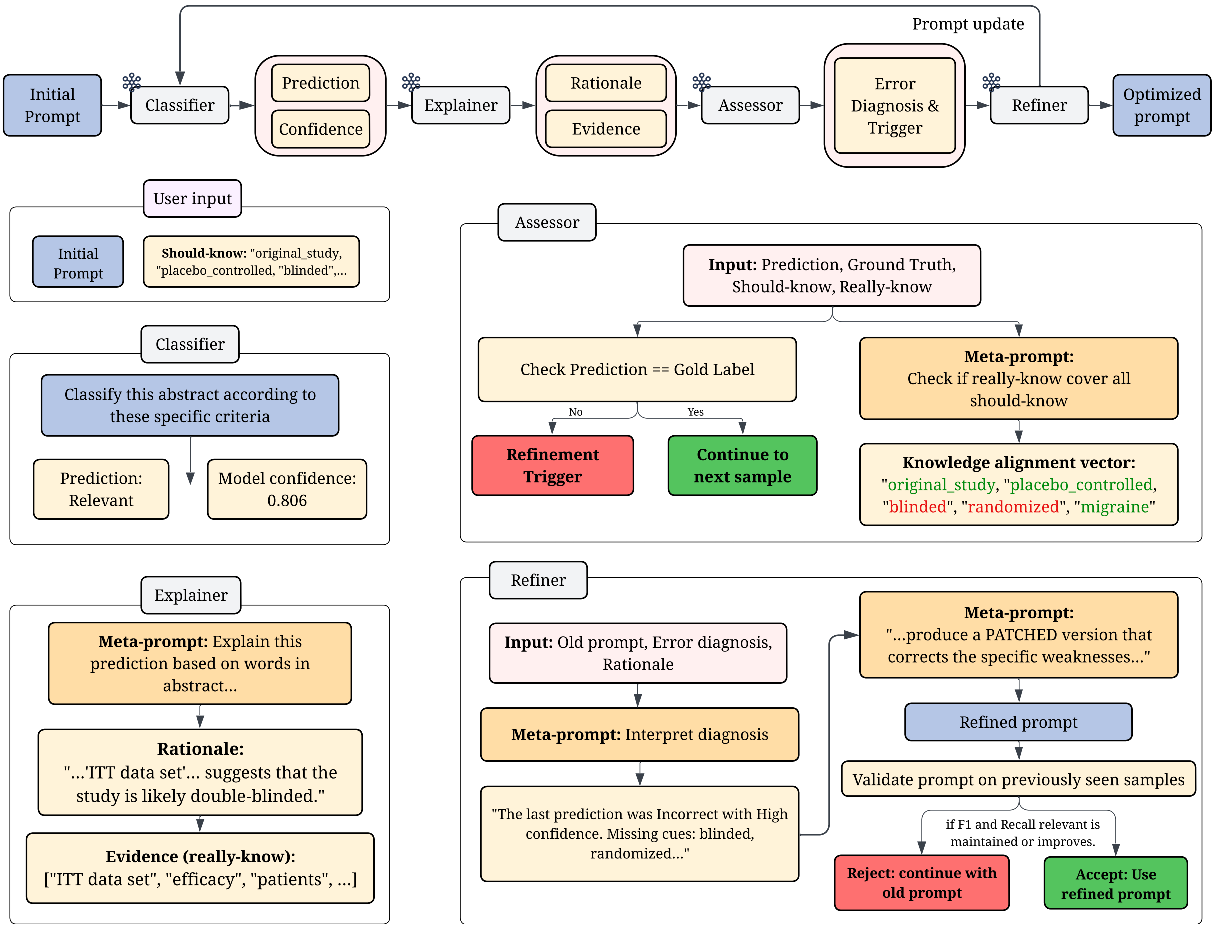
Prompt update
Initial Prompt
Classifier
Prediction
Confidence
Explainer
Rationale
Evidence
Assessor
Error Diagnosis & Trigger
Refiner
Optimized prompt
User input
Initial Prompt
Should-know: "original_study, "placebo_controlled, "blinded",...
Classifier
Classify this abstract according to these specific criteria
Prediction: Relevant
Model confidence: 0.806
Explainer
Meta-prompt: Explain this prediction based on words in abstract...
Rationale: "...'ITT data set'... suggests that the study is likely double-blinded."
Evidence (really-know): ["ITT data set", "efficacy", "patients", ...]
Assessor
Input: Prediction, Ground Truth, Should-know, Really-know
Check Prediction == Gold Label
No
Yes
Refinement Trigger
Continue to next sample
Meta-prompt: Check if really-know cover all should-know
Knowledge alignment vector: "original_study, "placebo_controlled, "blinded", "randomized", "migraine"
Refiner
Input: Old prompt, Error diagnosis, Rationale
Meta-prompt: Interpret diagnosis
"The last prediction was Incorrect with High confidence. Missing cues: blinded, randomized..."
Meta-prompt: "...produce a PATCHED version that corrects the specific weaknesses..."
Refined prompt
Validate prompt on previously seen samples
if F1 and Recall relevant is maintained or improves.
Reject: continue with old prompt
Accept: Use refined prompt

Figure 4: An overview of the IMAPR framework, illustrating its two-level structure. (Top) The high-level flowchart shows the main iterative loop connecting the four core modules: Classifier, Explainer, Assessor, and Refiner. (Bottom) The detailed panels walk through a concrete example of a single refinement cycle.

```
1. For each paper, evaluate the Title and
     Abstract according to the following five
     criteria:
   - Is it an **original study**? (Yes/No)
   - Is it **placebo-controlled**? (Yes/No)
   - Is it **double-blinded or
        triple-blinded**? (Yes/No)
   - Is it a **randomized clinical trial
        (RCT)**? (Yes/No)
   - Is the **main focus on migraine**? (Yes/No)

2. **Classification Rule:**
   - If **all five answers are "Yes"**,
        classify the paper as **"Relevant"**.
   - If **any answer is "No"**, classify the
        paper as **"Irrelevant"**.

3. **Return Format:**
   Return only one word: ‘"Relevant"‘ or
        ‘"Irrelevant"‘.
   Do not explain. Do not justify. Do not
        output anything else.

**Additional Clarification:** Please ensure
     that the title clearly states "blinded" to
     accurately assess the presence of this
     criterion.
**Clarification for "Randomized" Criterion:**
```

```
     Ensure that the title clearly states
     "randomized" to accurately assess the
     presence of this criterion.
**Clarification for "Placebo-Controlled"
     Criterion:** Please note that the term
     "placebo-controlled" implies the presence
     of this criterion.
**Clarification for "Double-Blinded"
     Criterion:** Note that "double-blinded" and
     "triple-blinded" are equivalent conditions.
>>>

**Domain Match Clarification:** Note that the
     term "blinded" implies both
     "double-blinded" and "triple-blinded"
     conditions.
>>>

**Classification Criteria Clarification:**
     Ensure that the title clearly states
     "randomized" to accurately assess the
     presence of this criterion.
>>>

**One-Line Reminder:** Ensure that the title
     clearly states "randomized" to accurately
     assess the presence of this criterion.
>>>
```

Stage	Details & Analysis
1. Initial Error	An abstract for a study that was not explicitly blinded or randomized was **incorrectly classified as ‘Relevant‘** with high confidence (0.81).
2. Flawed Rationale	The LLM generated a rationale where it **incorrectly inferred** the presence of the missing criteria, stating: *"Although the abstract does not explicitly mention blinding, it does mention ’ITT data set’... This suggests that the study is likely double-blinded."*
3. IMAPR's Diagnosis	The framework's internal signals correctly identified the reasoning failure: • **Alignment Vector:** The Assessor flagged the missing criteria, returning: `{"blinded": False, "randomized": False}`. • **Diagnosis Summary:** The system concluded: *"Missing cues: blinded, randomized. These missing cues likely explain the incorrect high-confidence prediction."*
4. The Refined Prompt	Based on the diagnosis, the Refiner generated a new prompt with a more constrained instruction, adding a crucial final note (in **bold**): *"...**Note:** When evaluating the criteria, please ensure that the abstract explicitly mentions the following: blinded, randomized..."*
5. The Outcome	The refined prompt correctly classified the original abstract as ‘Irrelevant‘. When evaluated on the rolling validation window, this new prompt was accepted.

Table 6: A case study of IMAPR diagnosing and correcting a reasoning failure. The initial prompt caused the LLM to incorrectly infer criteria that were not explicitly present in the abstract. IMAPR diagnosed this failure using the knowledge alignment signal and generated a more constrained prompt that resolved the error.

```
**One-Line Reminder for "Original Study"
    Criterion:** Ensure that the title clearly
    states "original" to accurately assess the
    presence of this criterion.
>>>

**One-Line Reminder for "Double-Blinded"
    Criterion:** Note that "double-blinded" and
    "triple-blinded" are equivalent conditions.
>>>

**Output Format:** Return only the final
    patched prompt textno commentary, no
    Markdown fences and not the same prompt as
    the original.
```

Listing 7: Best prompt for Llama 3.2 3B (Macro-F_1: 0.224, Recall$_{Rel}$: 1.0).

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**

1. For each paper, evaluate the Title and
    Abstract according to the following five
    criteria:
  - Is it an **original investigation**?
      (Yes/No) (Note: Assume new investigation
      unless explicitly stated otherwise, and
      explicitly look for phrases like "new
      investigation" or "investigation of", or
      phrases like "comparative effectiveness
      study" which implies a new
      investigation, and also verify the
      presence of a specific study design or
      methodology section in the Abstract,
      including explicit mention of study
      design, methodology, or research methods)
  - Is it **placebo-controlled**? (Yes/No)
      (Note: Look for phrases like
      "placebo-controlled study" or "placebo
      group" in the Title and Abstract, and
      specifically verify the presence of
      "placebo" in both the Title and
      Abstract, and also verify the presence
      of "placebo" in the Abstract to confirm
      it is placebo-controlled)
  - Is it **double-blinded or
      triple-blinded**? (Yes/No) (Note:
      Explicitly look for phrases like
      "double-blind" or "triple-blind" in the
      Title and Abstract)
  - Is it a **randomized clinical trial
      (RCT)**? (Yes/No) (Note: Look for
      phrases like "randomized" or "randomized
      controlled trial" in the Title and
      Abstract, and specifically verify the
      presence of "randomized" in both the
      Title and Abstract, or phrases like
      "phase 3 PREEMPT trials" which imply a
      randomized controlled trial, and also
      verify the presence of "randomized" in
      the Abstract to confirm it is a
      randomized clinical trial)
  - Is the **main focus on migraine**?
      (Yes/No) (Note: Look for phrases like
      "acute migraine therapy" or "migraine
      treatment" in the Title and Abstract,
      and specifically verify the presence of
```

```
       "migraine" in both the Title and
       Abstract)

2. **Classification Rule:**
   - If **all five answers are "Yes"**,
       classify the paper as **"Relevant"**.
   - If **any answer is "No"**, classify the
       paper as **"Irrelevant"**.

3. **Return Format:**
   Return only one word: ‘"Relevant"‘ or
       ‘"Irrelevant"‘.

4. **Additional Reminder:** Specifically verify
     the presence of the word "randomized" in
     both the Title and Abstract for the paper
     to be classified as a randomized clinical
     trial, and also verify the presence of
     "placebo" in both the Title and Abstract
     for the paper to be classified as
     placebo-controlled, and check if the study
     design or methodology section in the
     Abstract explicitly mentions "placebo" to
     confirm it is placebo-controlled.
```

Listing 8: Best prompt for Llama 3.1 8B (Macro-F_1: 0.612, Recall$_{Rel}$: 0.955).

```
Here is the patched prompt:

You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**

1. For each paper, evaluate the Title and
     Abstract according to the following five
     criteria:
   - Is it an **original study**, i.e., not a
       review, meta-analysis, or secondary
       analysis of existing data? Check for
       phrases like "randomized controlled
       trial", "clinical trial", "original
       research", or "primary study". Ensure
       that the study presents new data or
       findings. Verify that the study
       explicitly states its novelty. (Yes/No)
   - Is it **placebo-controlled**, meaning that
       it includes a group receiving a placebo
       treatment? Look for phrases like
       "placebo-controlled", "placebo group",
       "sham treatment", or "control group
       receiving placebo". (Yes/No)
   - Is it **double-blinded or
       triple-blinded**, meaning that both the
       researchers and participants are unaware
       of group assignments? Check for phrases
       like "double-blind", "triple-blind",
       "masked", "blinded", or
       "investigator-masked". Ensure that the
       study explicitly mentions that both
       researchers and participants are unaware
       of group assignments. Verify that the
       blinding is not only mentioned but also
       clearly described. Also, check if the
       study mentions any exceptions or
       limitations to the blinding. (Yes/No)
   - Does the study design explicitly mention
       **randomization** (e.g., "randomized",
       "randomly assigned", "random
       allocation", etc.)? Ensure that the
       randomization is clearly described and
       not just mentioned. Verify that the
       study explicitly states the method of
       randomization. **Remember to check for
       phrases like "randomized clinical trial"
       or "RCT" to confirm randomization**.
       (Yes/No)
   - Is the **main focus specifically on
       migraine**, rather than a broader
       category of headaches or other
       conditions? Check for phrases like
       "migraine", "migraine treatment",
       "migraine prevention", or "migraine
       management". Ensure that the study's
       primary objective is to investigate
       migraine and not just mention it as a
       secondary aspect. Also, verify that the
       study does not focus on a different
       condition that happens to have migraine
       as a symptom or comorbidity.
       Additionally, check if the study's
       population is specifically defined as
       having migraine. Verify that the study
       explicitly states its focus on migraine.
       (Yes/No)

2. **Classification Rule:**
   - If **all five answers are "Yes"**,
       classify the paper as **"Relevant"**.
   - If **any answer is "No"**, classify the
       paper as **"Irrelevant"**.

3. **Return Format:**
   Return only one word: ‘"Relevant"‘ or
       ‘"Irrelevant"‘.
   Do not explain. Do not justify. Do not
       output anything else.

**Reminder:** When evaluating the blinding
    criterion, ensure that the study explicitly
    mentions that both researchers and
    participants are unaware of group
    assignments and that the blinding is
    clearly described. Additionally, verify
    that the study's primary objective is to
    investigate migraine and not just mention
    it as a secondary aspect, and that the
    study does not focus on a different
    condition that happens to have migraine as
    a symptom or comorbidity. Also, ensure that
    the study explicitly states its novelty,
    method of randomization, and focus on
    migraine.
```

Listing 9: Best prompt for Llama 3.1 70B (Macro-F_1: 0.570, Recall$_{Rel}$: 0.966).

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine, remembering to check for explicit
    mentions of original study, indicated by
    phrases like "we conducted a trial" or
    "this study aimed to", and noting that
    implied or suggested characteristics may
    not suffice for a "Relevant"
    classification, also considering the
    following key aspects: is the study clearly
    an original study, as indicated by phrases
    like "original research" or "new study", is
    the study clearly placebo-controlled, with
    explicit comparison to a placebo, is the
    blinding method explicitly stated as
    double-blinded or triple-blinded, is the
    study clearly a randomized clinical trial
    (RCT), with explicit mention of
    randomization, and is the main focus
    explicitly on migraine, demonstrated by
    clear statements like "this study
    investigates migraine treatment" or
    "migraine patients were enrolled", and
    noting that the study must explicitly state
    its design as an original study, clearly
    mention placebo control, specify
    double-blinded or triple-blinded
    methodology, explicitly mention
    randomization, and have a primary focus
    explicitly stated as migraine, with
    reminders that original study means newly
    conducted research, placebo-controlled
    means compared to a placebo, blinded means
    double-blinded or triple-blinded,
    randomized means explicitly mentioning
    randomization, and migraine focus means
    explicitly stating migraine as the primary
    condition, and also remembering to verify
    that the text explicitly mentions the key
    terms "original study",
    "placebo-controlled", "blinded",
    "randomized", and "migraine" to ensure
    accurate classification.

**Classification Instructions**

1. For each paper, evaluate the Title and
    Abstract according to the following five
    criteria:
   - Is the study **explicitly stated as an
       original study**, with clear wording
       indicating it is not just an analysis or
       review of existing data, such as phrases
       like "we conducted a trial" or "this
       study aimed to", and does it clearly
       indicate that it is a new study, not a
       secondary analysis, with explicit
       phrases like "original research" or "new
       study", and is the study design
       explicitly mentioned as original, with
       the key term "original study" explicitly
       mentioned? (Yes/No)
   - Is the study **clearly
       placebo-controlled**, with explicit
       comparison to a placebo mentioned in the
       text, such as "patients received either
       the treatment or a placebo", and is the
       placebo control explicitly mentioned as
       part of the study design, using phrases
       like "placebo-controlled trial", and is
       the comparison to placebo clearly
       stated, with the key term
       "placebo-controlled" explicitly
       mentioned? (Yes/No)
   - Does the study **explicitly state its
       blinding method as double-blinded or
       triple-blinded**, with clear
       descriptions like "double-blind,
       placebo-controlled trial" or
       "triple-blinded randomized trial", and
       is the blinding method clearly described
       as double-blinded or triple-blinded,
       with specific mention of blinding, and
       is the blinding method explicitly
       stated, with the key term "blinded"
       explicitly mentioned? (Yes/No)
   - Is the study **clearly a randomized
       clinical trial (RCT)**, with explicit
       mention of randomization, such as
       "patients were randomized to treatment
       groups" or "random assignment to
       treatment arms", and does the text
       clearly state that the study is
       randomized, using phrases like
       "randomized trial" or "randomized
       study", and is the randomization
       explicitly mentioned, with the key term
       "randomized" explicitly mentioned?
       (Yes/No)
   - Is the **main focus explicitly on
       migraine**, with migraine clearly stated
       as the primary condition studied or
       treated, such as "this study
       investigates migraine treatment" or
       "migraine patients were enrolled", and
       is migraine explicitly mentioned as the
       primary focus of the study, with clear
       statements like "migraine research" or
       "migraine study", and is the focus on
       migraine clearly stated, with the key
       term "migraine" explicitly mentioned?
       (Yes/No)

2. **Classification Rule:**
   - If **all five answers are "Yes"**,
       classify the paper as **"Relevant"**.
   - If **any answer is "No"**, classify the
       paper as **"Irrelevant"**.

3. **Return Format:**
   Return only one word: ‘"Relevant"‘ or
       ‘"Irrelevant"‘.
```

Listing 10: Best prompt for Llama 3.3 70B (Macro-F_1: 0.631, Recall$_{Rel}$: 0.955).

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**

1. For each paper, evaluate the Title and
    Abstract according to the following five
    criteria:
   - Is it an **original study** (i.e., a new
       investigation, not a review or
       meta-analysis, and not a secondary
       analysis of existing data, with a clear
       statement of a research question or
       hypothesis, and not a study that only
       presents a new analysis or
       interpretation of existing data)?
       (Yes/No)
   - Is it **placebo-controlled** (i.e.,
       includes a placebo arm as a control
       group, and the placebo is not used as an
       active comparator, with explicit mention
       of placebo control, and the placebo
       control is used throughout the entire
       study)? (Yes/No)
   - Is it **double-blinded or triple-blinded**
       (i.e., both participants and
       investigators are blinded to treatment
       assignments throughout the entire study,
       with **explicit mention of blinding
       method**, such as ”double-blind”,
       ”triple-blind”, ”participant-blind”, or
       ”investigator-blind”, and the blinding
       method is clearly described)? **Implicit
       suggestions of blinding are
       insufficient; explicit mention is
       required.** (Yes/No)
   - Is it a **randomized clinical trial
       (RCT)** (i.e., participants are randomly
       assigned to treatment groups using a
       clear randomization method, such as a
       random number generator or a
       computer-generated randomization
       schedule, and the randomization is
       explicitly stated)? **Verify that the
       randomization method is explicitly
       described.** (Yes/No)
   - Is the **main focus on migraine** (i.e.,
       migraine is the primary condition being
       studied, and not just a secondary
       outcome or subgroup analysis, with clear
       mention of migraine as the primary
       endpoint, and the study is primarily
       designed to investigate migraine)?
       **Ensure migraine is the central focus,
       not a peripheral aspect, and that the
       study aims to investigate migraine
       specifically.** (Yes/No)

2. **Classification Rule:**
   - If **all five answers are ”Yes”**,
       classify the paper as **”Relevant”**.
   - If **any answer is ”No”**, classify the
       paper as **”Irrelevant”**.

3. **Return Format:**
   Return only one word: ‘”Relevant”‘ or
       ‘”Irrelevant”‘.
   Do not explain. Do not justify. Do not
       output anything else.
```

Listing 11: Best prompt for Llama 3.1 405B (Macro-F_1: 0.636, Recall$_{Rel}$: 0.978).

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**

1. For each paper, evaluate the Title and
    Abstract according to the following five
    criteria:
   - Is it an **original study**? (Must
       describe new prospective data collection
       with explicit methods section. Require
       terms like ”prospective,” ”clinical
       trial,” or ”study protocol” - must show
       evidence of original data collection,
       not pooled/combined analysis. Verify
       explicit description of data collection
       methods in abstract.)
   - Is it **placebo-controlled**? (Requires
       explicit ”placebo group” or
       ”placebo-controlled” terminology in
       abstract, not just ”control group” or
       active comparators. Must appear verbatim
       - no inferences. Cross-check for placebo
       mentioned in both methods and results
       sections of abstract.)
   - Is it **double-blinded or
       triple-blinded**? (Must specify
       ”double-blind” or ”triple-blind” in
       abstract, not just ”blinded” or implied
       by RCT status. No assumptions allowed -
       exact phrasing required. Verify blinding
       method is explicitly stated in both
       title and abstract.)
   - Is it a **randomized clinical trial
       (RCT)**? (Must describe randomization
       procedure in abstract, not just use
       ”randomized” without details. Requires
       explicit method description - check for
       randomization methodology in abstract.
       Ensure proper randomization technique
       mentioned verbatim.)
   - Is the **main focus on migraine**?
       (Primary outcome/study population must
       specifically mention ”migraine”
       diagnosis in title/abstract, not just
       headache disorders generally. Must
       appear verbatim - no broader headache
       terms accepted. Confirm migraine is
       primary focus by checking all sections
       of abstract.)

2. **Classification Rule:**
   - If **all five answers are ”Yes”** based on
       explicit evidence in the text, classify
       the paper as **”Relevant”**.
   - If **any answer is ”No” or cannot be
       confirmed**, classify the paper as
       **”Irrelevant”**.

3. **Return Format:**
   Return only one word: ‘”Relevant”‘ or
       ‘”Irrelevant”‘.
   Do not explain. Do not justify. Do not
       output anything else.
```

Listing 12: Best prompt for Deepseek V3 (Macro-F_1: 0.615, Recall$_{Rel}$: 0.978).

```
You are an expert research assistant evaluating
    medical papers about placebo-controlled,
    blinded, randomized clinical trials for
    migraine.

**Classification Instructions**

1. For each paper, evaluate the Title and
    Abstract according to the following five
    criteria:
   - Is it an **original study**? (Yes/No)
       Ensure the study presents new research
       findings and is not a secondary analysis
       or review.
   - Is it **placebo-controlled**? (Yes/No)
       Confirm the study includes a placebo
       group.
   - Is it **double-blinded or
       triple-blinded**? (Yes/No) Verify the
       study explicitly mentions blinding.
   - Is it a **randomized clinical trial
       (RCT)**? (Yes/No) Check for explicit
       mention of randomization.
   - Is the **main focus on migraine**?
       (Yes/No) Ensure the study primarily
       addresses migraine.

2. **Classification Rule:**
   - If **all five answers are "Yes"**,
       classify the paper as **"Relevant"**.
   - If **any answer is "No"**, classify the
       paper as **"Irrelevant"**.

3. **Return Format:** Return only one word:
    ‘"Relevant"‘ or ‘"Irrelevant"‘.
   Do not explain. Do not justify. Do not
       output anything else.
   Ensure that all criteria are strictly
       evaluated based on the information
       provided in the Title and Abstract only.
   Pay close attention to the explicit mention
       of each criterion to avoid
       misclassification.
   Be aware of implicit information that may
       not be explicitly stated but is crucial
       for accurate classification.
   Double-check each criterion to ensure no
       oversight in the evaluation process.
   Remember to consider the context and nuances
       in the language used in the Title and
       Abstract to accurately determine the
       presence of each criterion.
```

Listing 13: Best prompt for GPT-4o (Macro-F_1: 0.671, Recall$_{Rel}$: 0.989).

D Correctness Oracle Performance

Table 7 presents the run-by-run performance of the correctness oracle (trained on the full 100% dataset). This data provides the evidence for the claim made in the main text that the success of the test-time refinement mechanism is directly correlated with the oracle's performance in that specific run.

E Baseline Meta-Prompts

This appendix contains the meta-prompts used to run the external baselines in our experiments.

E.1 APE (Automatic Prompt Engineer)

The following template was used to generate candidate prompts for the APE baseline. The model is given a set of input-output examples and asked to infer the instruction that produced them.

Table 7: Run-by-run oracle performance and its impact on the 100% dataset. Oracle Rec$_{Err}$ is the oracle's recall on the error class (class 0). F_1 gain is the final gain in Macro-F_1 from test-time refinement.

Run	Oracle Rec$_{Err}$	Oracle Macro-F_1	F_1 gain
1	0.748	0.766	+0.066
2	0.733	0.775	+0.028
3	0.475	0.712	-0.021
4	0.000	0.456	+0.000
5	0.000	0.449	+0.000
Mean	**0.391**	**0.632**	**+0.015**

```
I gave a friend an instruction and six inputs.
    The friend read the instruction and wrote
    an output for every one of the inputs.
Here are the input-output pairs:

Input: <Title 1>
<Abstract 1>
Output: <Label 1>

Input: <Title 2>
<Abstract 2>
Output: <Label 2>

... (and so on for all 6 examples) ...

The detailed instruction was:
```

Listing 14: Meta-prompt for APE candidate generation.

E.2 GPO (Gradient-inspired Prompt Optimizer)

The GPO baseline uses the following meta-prompt to iteratively refine its prompt. It provides the optimizer LLM with the current best prompt and a list of other relevant prompts from its history to guide the generation of a new candidate.

```
Your task is to write a new, improved prompt.
    You are allowed to change up to <current
    edit cap> words from the current best
    prompt. You are NOT allowed to change the
    return format.

--- CURRENT BEST PROMPT ---
Prompt: <current best prompt text>
Score: <current best score>/100

--- RELEVANT PREVIOUS PROMPTS (higher scores
    are better) ---
<List of relevant previous prompts and their
    scores>
```

Listing 15: Meta-prompt for the GPO baseline.

E.3 StraGo (Strategic-Guided Optimization)

The StraGo baseline uses a multi-step refinement process involving experience analysis, strategy generation, and prompt optimization. The following prompts are used at each stage of the process.

```
As a logician, you excel at breaking down
    reasoning step by step.

<prompt>
<Current prompt text>
</prompt>

<examples>
<Correctly answered examples>
</examples>

TASK ---
Summarise the **<Number of
    reasons> most valuable reasons** this
    prompt produced the correct answers above.
Return a JSON list of length <Number of
    reasons>; each element must contain:
  - "reason": one concise sentence.
```

Listing 16: StraGo prompt for generating positive experiences from success cases.

```
As a logician, you excel at breaking down
    reasoning step by step.

<prompt>
<Current prompt text>
</prompt>

<examples>
<Incorrectly answered examples>
</examples>

TASK ---
Identify the **<Number of
    flaws> primary flaws** in the prompt that
    caused the wrong answers.
Return a JSON list of length <Number of
    flaws>; each element must contain:
  - "error": one concise sentence.
```

Listing 17: StraGo prompt for generating negative experiences from failure cases.

```
You are an expert prompt engineer. Craft a
    step-by-step strategy that
fixes the issue while preserving successful
    behaviour.

# Demos (few-shot)
<Few-shot examples of experience-to-strategy
    generation>

<prompt>
<Current prompt text>
</prompt>

<example>
<Specific example case>
</example>

<experience>
<A single positive or negative experience>
</experience>

OUTPUT ---
Return a **numbered list** (3-6 steps). Each
    step must start with an imperative verb.
Return only the list.
```

Listing 18: StraGo prompt for generating a corrective strategy.

```
You are the Optimizer.

Current instruction:
<prompt>
<Current prompt text>
</prompt>

Task data (for context, may be empty):
<example>
<Specific example case>
</example>

Guidance:
<experience>
<The experience being addressed>
</experience>

Strategy to apply:
<strategy>
<The strategy generated in the previous step>
</strategy>

Rewrite the instruction **once** so that it:
- Implements every step of the strategy.
- Retains original intent and style.
- Fits within 400 tokens.

Return only the revised instruction --- no
    explanations.
```

Listing 19: StraGo prompt for rewriting the main prompt based on the generated strategy.

F Full Run-by-Run Results

This appendix provides the detailed, run-by-run data that supports the aggregated results presented in the main paper.

F.1 Train vs Test refined results

Table 8 contains the complete performance scores for the data efficiency experiments (RQ2), showing the Macro-F_1 and Recall$_{Rel}$ for both the ‘Train-Refined‘ and ‘Test-Refined‘ configurations for every run and data subset with Llama 3.1 8B.

F.2 Model Generalizability Results (RQ4)

Table 9 provides the detailed, run-by-run scores for the model generalizability experiments. For each LLM, it compares the performance of the static prompt against the IMAPR-refined prompt. The aggregated mean and standard deviation for each model correspond to the values presented in Table 4 in the main paper.

Table 8: Full run-by-run results for data efficiency experiments, showing Macro-F_1 and Recall$_{Rel}$ scores.

	Train-Refined		**Test-Refined**	
Run ID	**Macro-F_1**	**Recall$_{Rel}$**	**Macro-F_1**	**Recall$_{Rel}$**
		5% Training Data		
Run 1	0.526	0.989	0.531	0.989
Run 2	0.558	0.989	0.558	0.989
Run 3	0.484	0.989	0.484	0.989
Run 4	0.476	0.989	0.476	0.989
Run 5	0.510	0.978	0.510	0.978
Mean (±SD)	**0.511 (±0.033)**	**0.987 (±0.005)**	**0.512 (±0.032)**	**0.987 (±0.005)**
		10% Training Data		
Run 1	0.568	0.966	0.573	0.966
Run 2	0.535	0.989	0.535	0.989
Run 3	0.484	0.989	0.519	0.989
Run 4	0.478	0.989	0.478	0.989
Run 5	0.495	0.989	0.495	0.989
Mean (±SD)	**0.512 (±0.038)**	**0.984 (±0.010)**	**0.520 (±0.036)**	**0.984 (±0.010)**
		20% Training Data		
Run 1	0.486	0.989	0.486	0.989
Run 2	0.484	0.989	0.543	0.989
Run 3	0.507	0.989	0.543	0.989
Run 4	0.472	0.989	0.531	0.978
Run 5	0.485	1.000	0.500	0.978
Mean (±SD)	**0.487 (±0.013)**	**0.991 (±0.005)**	**0.520 (±0.025)**	**0.984 (±0.006)**
		40% Training Data		
Run 1	0.566	0.966	0.583	0.944
Run 2	0.526	0.978	0.550	0.989
Run 3	0.574	0.989	0.574	0.989
Run 4	0.612	0.809	0.612	0.809
Run 5	0.500	1.000	0.563	0.978
Mean (±SD)	**0.556 (±0.043)**	**0.948 (±0.079)**	**0.576 (±0.022)**	**0.942 (±0.072)**
		100% Training Data		
Run 1	0.569	0.978	0.635	0.876
Run 2	0.570	0.966	0.598	0.978
Run 3	0.612	0.955	0.591	0.933
Run 4	0.586	0.944	0.586	0.944
Run 5	0.571	0.966	0.571	0.966
Mean (±SD)	**0.582 (±0.018)**	**0.962 (±0.012)**	**0.596 (±0.023)**	**0.939 (±0.035)**

Table 9: Full run-by-run results for the model generalizability study (RQ4).

		Static Prompt		**IMAPR (Train-Refined)**	
LLM	**Run ID**	**Macro-F_1**	**Recall$_{Rel}$**	**Macro-F_1**	**Recall$_{Rel}$**
GPT-4o					
	Run 1	0.660	0.978	0.642	0.966
	Run 2	0.661	0.989	0.661	1.000
	Run 3	0.655	0.978	0.671	0.989
	Run 4	0.664	0.900	0.631	0.989
	Run 5	0.664	0.911	0.656	0.955
	Mean (±SD)	**0.661 (±0.003)**	**0.951 (±0.035)**	**0.652 (±0.015)**	**0.980 (±0.018)**
DeepSeek-V3					
	Run 1	0.552	0.989	0.631	0.944
	Run 2	0.552	0.989	0.600	0.966
	Run 3	0.552	0.989	0.615	0.978
	Run 4	0.542	0.989	0.613	0.944
	Run 5	0.549	1.000	0.604	0.966
	Mean (±SD)	**0.549 (±0.004)**	**0.991 (±0.004)**	**0.612 (±0.010)**	**0.959 (±0.020)**
Llama3.1-405B					
	Run 1	0.590	1.000	0.636	0.978
	Run 2	0.590	1.000	0.612	1.000
	Run 3	0.590	1.000	0.628	0.787
	Run 4	0.516	0.989	0.649	0.921
	Run 5	0.518	0.989	0.612	0.978
	Mean (±SD)	**0.561 (±0.034)**	**0.995 (±0.004)**	**0.627 (±0.014)**	**0.933 (±0.081)**
Llama3.3-70B					
	Run 1	0.565	1.000	0.575	1.000
	Run 2	0.564	1.000	0.631	0.955
	Run 3	0.567	1.000	0.568	1.000
	Run 4	0.579	0.989	0.600	0.989
	Run 5	0.579	0.989	0.617	0.955
	Mean (±SD)	**0.571 (±0.006)**	**0.995 (±0.004)**	**0.598 (±0.026)**	**0.980 (±0.022)**
Llama3.1-70B					
	Run 1	0.552	1.000	0.576	0.876
	Run 2	0.552	1.000	0.570	0.966
	Run 3	0.552	1.000	0.560	0.966
	Run 4	0.569	0.989	0.568	0.989
	Run 5	0.569	0.989	0.567	1.000
	Mean (±SD)	**0.559 (±0.008)**	**0.995 (±0.004)**	**0.568 (±0.010)**	**0.959 (±0.050)**
Llama3.2-3B					
	Run 1	0.122	1.000	0.224	1.000
	Run 2	0.122	1.000	0.096	1.000
	Run 3	0.122	1.000	0.074	1.000
	Run 4	0.105	1.000	0.121	1.000
	Run 5	0.105	1.000	0.079	1.000
	Mean (±SD)	**0.115 (±0.008)**	**1.000 (±0.000)**	**0.119 (±0.060)**	**1.000 (±0.000)**