Trustworthy LLM-Based Clinical Trial Matching with Satisfiability Modulo Theories

Anonymous ACL submission

Abstract

Clinical trial matching is a high-stakes language understanding task where accuracy alone is not enough: clinicians must also understand why a patient is matched or rejected, what evidence supports the decision, and what assumptions remain under missing information. We study patient–trial eligibility matching and show that strong LLM-based matchers can produce plausible rationales that are difficult to audit and not always faithful to their own decisions. We propose SMTMATCH, a hybrid LLM+SMT framework that separates reading from deciding: LLMs extract structured trial requirements and patient facts, while an SMT/MaxSAT solver derives verdicts, criterion-level statuses, residual assumptions, and rejection reasons from executable constraints. Across accuracy evaluation, a clinician study of rationale usefulness, and counterfactual self-faithfulness tests, SMTMATCH matches or improves on strong LLM baselines while producing more interpretable decisions and useful rationales. Our results suggest that clinical trial matching benefits from separating language understanding from decision execution, especially when explanations must be inspectable and faithful to the underlying verdict.

1 Introduction

Applying AI in high-stakes domains, such as healthcare, requires more than predictive accuracy—it demands systems that are trustworthy, transparent, and useful to clinicians and patients. In clinical contexts, decisions are rarely clear-cut; ambiguity is common due to missing data, uncertain institutional policies, or ill-defined patient preferences. As a result, clinicians need more than a final answer: they must understand the system's reasoning, the assumptions it makes, and how its decision might change as evidence or policies evolve.

Clinical trial matching serves as an example of these challenges. While automating this process could help overcome recruitment bottlenecks that slow medical progress, errors are costly: excluding a suitable patient can deny them timely and potentially life-saving experimental therapies, whereas admitting ineligible candidates wastes resources or endangers patient safety [5, 18, 12]. The complexity of eligibility criteria—often spanning diagnoses, treatments, time frames, lab values, and exceptions—alongside incomplete patient records, makes reliable automation challenging. Effective matching, therefore, requires not just strong language understanding, but careful reasoning about missing information, clinical conventions, and the preferences of both clinicians and patients.

In this setting, providing clear and faithful rationales is essential. Knowing why a patient is currently ineligible empowers clinicians to take appropriate next steps—for instance, ordering missing lab tests or monitoring a condition such as cancer that could eventually lead to eligibility. Thus, the quality and faithfulness of rationales are as crucial as predictive outcomes.

To meet these requirements, we propose evaluating clinical trial matchers using two complementary approaches:

First, **rationale usefulness** is a property evaluated by human experts, encompassing both interpretability and faithful reasoning. An **interpretable** rationale should address all trial eligibility criteria (*criteria completeness*), clearly cite supporting evidence or assumptions from the patient record (*traceability*), and provide actionable insight that identifies modifiable barriers to eligibility (*actionability*). **Faithful reasoning** requires that the rationale exhibits *logical consistency*—avoiding contradictions or self-conflicting statements—and offers *support*, that is, sufficient justification for each eligibility verdict. Human evaluation is indispensable for ensuring these qualities, as only clinicians can judge the clinical relevance and reliability of such explanations.

Second, we introduce **counterfactual self-faithfulness** as a quantitative, automated metric. This approach systematically modifies or removes the blockers identified in the rationale—the reasons for ineligibility—while keeping all other patient information unchanged. A matcher that reasons faithfully should then update its eligibility verdict accordingly. By measuring accuracy on these counterfactual cases, we directly test whether the system's decisions truly depend on the cited factors, rather than on spurious correlations or memorized patterns. Unlike conventional benchmarks that may suffer from leakage or overfitting, this method provides targeted, novel scenarios that rigorously probe both the faithfulness of the rationale and the reasoning capabilities of the matcher itself.

Matcher Design. A common approach to clinical trial matching is to use LLM-based matchers [9, 6, 20], where a large language model is prompted to determine if a patient is eligible for a trial and to articulate its reasoning step by step. To this end, we developed LLMMATCH, a carefully constructed agentic pipeline that outperforms previous methods (see Appendix C for details).

Alternatively, SATIR introduced a formal approach that uses model checking to improve recall in retrieving relevant and eligible trials [24]. In SatIR, LLMs independently translate trial criteria and patient records into Satisfiability Modulo Theories (SMT) formulas [1] grounded in the SNOMED medical ontology [16]. Efficient and accurate retrieval is achieved by projecting the SMT representation to relational algebra. However, SATIR does not address SMT-based trial matching.

Our first contribution of this paper is **to formulate the LLM trial matching task using SMT.** Unlike the SATIR approach, our SMTMATCH method derives the LLM constraints by considering both the trial and patient records together. We use a SAT solver [3] to determine if a set of constraints are SATISFIED, VIOLATED, or DEFERRED. It also generates a formal *trace* deriving the solution and any *residual constraints* that need to be satisfied (due to the presence of unknown variables) in the case of DEFERRED. In the case of VIOLATED, we use a MaxSAT solver [8] to determine the *flip set*, the minimum set of changes that can reverse the decision. The results can be verbalized to provide the user with the rationale.

By construction, **if the semantic parser correctly translates the problem to SMT**, SMTMATCH provides a verifiably faithful and *useful rationale*, and is necessarily *self-faithful*.

Secondly, this paper shows that **SMTMATCH and LLMMATCH have similar accuracy on our TRIALELIG benchmark.** Our TRIALELIG consists of the 552 patient–trial pairs in the SIGIR trial-matching dataset [10], with LLM-ensemble labels and clinician review of a stratified subsample. Across LLM backbones, our formal layer improves F1 over the strongest LLM baseline.

Third, we show that **SMTMATCH is more useful and trustworthy**: clinicians strongly prefer its rationales. Fourth, we show that **LLM-based matchers achieve at most 65% self-faithfulness**, whereas SMTMATCH is counterfactually self-faithful by construction, assuming accurate natural-language-to-SMT parsing.

2 Related Work

LLM-based patient–trial matching. Recent systems use LLMs to match patients to trials by interpreting patient records and eligibility criteria in natural language. TrialGPT [9] decomposes criteria, predicts criterion-level relevance and eligibility, and aggregates these judgments for ranking. Zero-shot LLM matching [20] and PRISM [6] similarly demonstrate the flexibility of LLMs for clinical text interpretation. These systems primarily rely on generated rationales or criterion-level judgments to support the final decision. Our work targets the post-retrieval decision stage: given a patient–trial pair, we move the final eligibility decision from free-text generation to symbolic evaluation, producing verdicts and rationales from logical traces.

Executable eligibility criteria and formal trial retrieval. A long line of work has translated eligibility criteria into structured or executable forms for cohort identification. Systems such as Criteria2Query and CriteriaMapper [22, 11] map free-text criteria to database queries, clinical concepts, or computable phenotypes. SATIR [24] introduced an LLM-based semantic parser that translates patient records and trial criteria into symbolic constraints for scalable trial retrieval. We build on this substrate but study a different problem: after a candidate trial has been retrieved, how should a system make and explain the final patient–trial eligibility decision? More broadly, our work relates to NL2FOL and NL2SMT semantic parsing [21, 7], but uses logical forms as decision artifacts rather than treating semantic parsing as the endpoint.

Trustworthy clinical LLM reasoning. Clinical LLMs must be evaluated not only for accuracy, but also for safety, auditability, and explanation quality. Med-PaLM and MultiMedQA [15] helped establish human evaluation dimensions such as factuality, reasoning, bias, and potential harm. MedAlign [4] shows that standard medical QA benchmarks miss realistic clinician information needs by introducing clinician-written EHR instructions. Work on LLM explanations for clinical decision support [17] highlights the difficulty of producing explanations that are both useful and reliable. Our work addresses these concerns in trial matching by deriving rationales from the same theorem-prover traces that produce the final decision.

3 How is SMT-Based Matcher Different?

LLM-based and SMT-based matchers may achieve similar headline accuracy, but they decide in different ways. An LLM matcher produces a natural-language rationale, but that rationale need not reflect the process that produced the verdict. It may mention the right criterion while missing key chart evidence, or cite a plausible reason that was not decisive. SMTMATCH, by contrast, first extracts structured facts from the chart and criteria as logic constraints, then checks them with an external solver. Its explanation is therefore derived from the actual formal decision process. We illustrate this difference using representative GPT-4.1 cases.

Example 1 (LLM ignoring evidence). A trauma-resuscitation trial excludes patients who arrive more than six hours after injury; the patient fell off his bike a week earlier. The LLM incorrectly marks the patient ELIGIBLE, saying the timing is unspecified and the exclusion is not triggered. Although it lists the right exclusion criterion, it misses the evidence in the chart. SMTMATCH instead extracts `hours_since_injury` ≈ 168, checks it against the six-hour threshold, and rejects the patient.

Note that the trial also excludes the use of anticoagulant and has additional constraints on the body weight and receipt of blood products, but none of this information is mentioned in the patient's record. SMTMATCH explicitly states the patient is considered eligible assuming these residual constraints are satisfied. On the other hand, the LLM rationale simply collapses these details into "no exclusion documented," so the reader cannot tell which criteria were checked, which facts were missing, and how missing evidence was handled.

Example 2 (LLM's unfaithful rationale). Consider a trial for PCOS-related sleep apnea. The trial requires the patient to have PCOS and excludes patients with prolactin excess or galactorrhea. The patient does not have PCOS and also has both prolactin excess and galactorrhea. The LLM correctly rejects the patient, but its rationale cites only prolactin excess and galactorrhea, omitting the missing PCOS diagnosis. As a result, if we create a counterfactual chart that removes only the cited blockers, the patient still remains ineligible because the uncited PCOS requirement is still unmet. The verdict is right, but the rationale is incomplete: it does not identify all the facts that support the rejection.

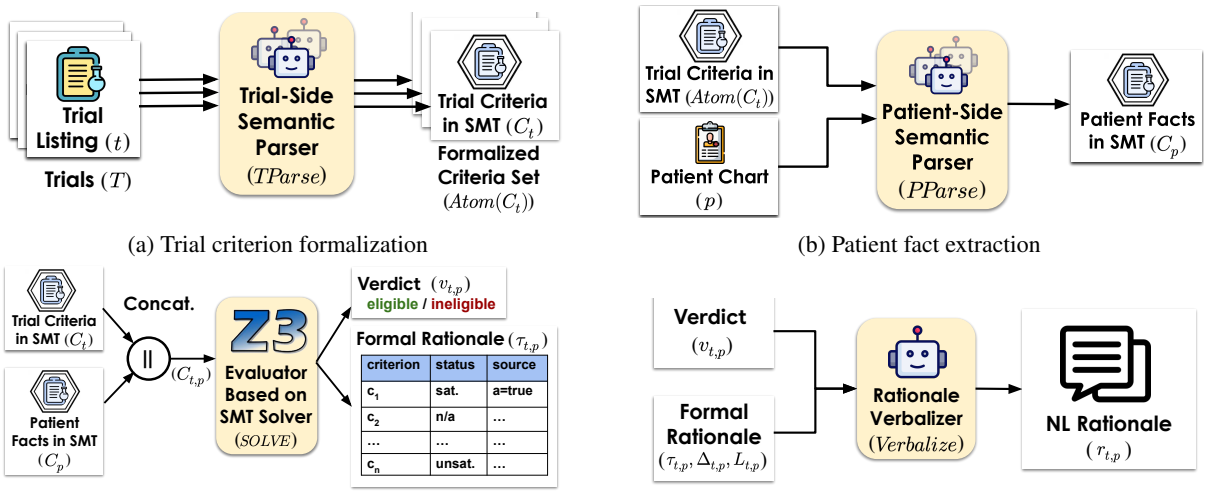
SMTMATCH instead computes the full flip set, identifying all three reasons that must change for the patient to become eligible: the patient would need to have PCOS, not have prolactin excess, and not have galactorrhea. A counterfactual that changes all three conditions would flip the formal verdict to eligible. Thus, SMTMATCH makes the decision boundary explicit: every condition that contributes to rejection appears in the trace, rather than being hidden behind a plausible but incomplete natural-language rationale.

Example 3 (SMTMATCH failure due to strictness). Consider a Kawasaki-disease trial that requires fever for more than five days, and a two-year-old patient with five days of high fever plus classic Kawasaki features. The LLM applies clinical judgment and correctly marks the patient ELIGIBLE. SMTMATCH instead extracts `fever_days = 5`, evaluates the trial condition as `fever_days > 5`, and rejects the patient on a one-day technicality.

While this is a real SMT error, it is localized. The trial requirement was formalized as a strict inequality, even though the relevant clinical standard treats five days as sufficient. Because the predicate appears explicitly in the formal trace, a clinician can immediately identify the off-by-one error, change > 5 to ≥ 5, and re-run the decision. In contrast, an opaque LLM rationale offers no equally precise place to inspect or repair the mistake.

4 SMT-Based Clinical Trial Matching

SMTMATCH has a hybrid design: LLMs read clinical text, map language to structured requirements, and propose relevant patient facts, while formal reasoning applies an explicit matching policy to produce a rerunnable verdict. In this framework, rationales are renderings of an executable decision



(a) Trial criterion formalization

(b) Patient fact extraction

(c) SMT eligibility evaluation

(d) Solver-grounded rationale generation

Figure 1: **Overview of the SMTMATCH clinical-trial matcher architecture.** The system formalizes trial criteria and patient charts as SMT, uses a solver to decide eligibility, and verbalizes the solver trace as a rationale.

object rather than post-hoc stories. Equivalent criteria can compile to equivalent constraints, verdicts follow from formulas, missing evidence and defaults are explicit, and users can modify policies or residual constraints and rerun the decision.

4.1 Task Formalization

Given a trial $t \in T$ and a patient $p \in P$, the clinical trial matching task

$$\text{MATCH}(t, p) = \{v, r\},$$

where $v \in \{\text{ELIGIBLE}, \text{INELIGIBLE}\}$ is the verdict, and r is a rationale identifying the patient evidence and trial requirements that support the verdict.

4.2 Clinical Trial Matching Constraints

Let $\text{TPARSE}(t)$ be the semantic parse of a trial t in natural language to an SMT formula;

$\text{ATOM}(f)$ be the atoms of SMT formula f;

$\text{PPARSE}(A, p)$ be the SMT semantic parse of a patient record p in natural language with respect to A, the set of atoms in the formal constraints of a trial being matched.

Note that PPARSE is dependent on the trial of interest because it needs to include only and all the information of significance to that trial.

Definition 1. Given a trial $t \in T$ and a patient $p \in P$, the constraints of their match

$$C_{t,p} = C_t \wedge C_p, \text{ where}$$

$C_t = \text{TPARSE}(t), C_p = \text{PPARSE}(\text{ATOM}(C_t), p)$. □

The semantic parse for trials, TPARSE, is similar to that of Zhou et al. [24]. All the terms are canonicalized using the SNOMED ontology. The atoms in the constraints are defined by a typed signature and a self-contained, human-readable definition.

Unlike Zhou et al. [24] however, which extracts every possible clinical fact from a patient record, PPARSE extracts only information related to each atom predicate of a given trial (Figure 1b). An LLM is prompted with the typed signature and self-contained, human-readable definition to emit the assertions whose truth values are given by the patient records. Predicates not addressed by the record are left unbound. For numerical atoms, PPARSE first uses explicit values in the patient record when available; otherwise, under the selected missing-evidence policy, it may use contextual evidence to decide whether the corresponding threshold is likely satisfied. Predicates that cannot be supported by the record are left unbound.

4.3 The SMTMATCH Method

Satisfiability Modulo Theories (SMT) checks whether logical formulas are satisfiable under background theories such as arithmetic, equality, and datatypes [1]. The MaxSAT solver determines the maximum number of clauses, of a given Boolean formula in conjunctive normal form, that can be made true by an assignment of truth values to the variables of the formula. It is a generalization of the Boolean satisfiability problem, which asks whether there exists a truth assignment that makes all clauses true.

Definition 2. Given constraints of a clinical trial and a patient $C_{t,p}$, the matcher

$$\text{SMTMATCH} = \text{VERBALIZE}(\text{SOLVE}(C_{t,p})),$$

where the verdict and formal rationale generated by the SOLVE SMT/MaxSAT solver are fed to VERBALIZE, an LLM agent that converts the result to text. □

Definition 3. Given constraints of a clinical trial and patient $C_{t,p} = C_t \wedge C_p$, the SMT/MaxSAT matcher

$$\text{SOLVE}(C_{t,p}) = \{v_{t,p}, \tau_{t,p}, \rho_{t,p}, \Delta_{t,p}, L_{t,p}\},$$

where

$v_{t,p} \in \{\text{ELIGIBLE}, \text{INELIGIBLE}\}$ is the verdict,

$\tau_{t,p}$ is a trace that identifies the assertions that determine the verdict,

$\rho_{t,p}$ are the residual constraints or assumptions made in support of eligibility,

$\Delta_{t,p}$ is the flip set, the minimum set of patient facts that block the patient from eligibility, and

$L_{t,p}$ labels the verdict of each criterion in the constraints.□

SOLVE uses Z3 [3] to decide the satisfiability on $C_{t,p}$, as shown in Figure 1c. In the presence of unknown values, Z3 considers the constraints to be *satisfiable* as long as there exists an assignment to the unknowns that satisfies the constraints. If $C_{t,p}$ is satisfiable, it returns the ELIGIBLE verdict, $\Delta_{t,p} = \emptyset$, and $\rho_{t,p}$ is the set of residual constraints, involving the unknowns, that remain to be satisfied. In other words, $\rho_{t,p}$ makes explicit the assumptions made in the judgment. If $C_{t,p}$ is not satisfiable, it returns an INELIGIBLE verdict and computes, using MaxSAT [2], the flip set with weighted self constraints:

$$\Delta_{t,p} = \arg\min_{\Delta} \text{cost}(\Delta) \quad \text{s.t.} \quad C_t \wedge \text{INVERT}(C_p, \Delta) \text{ is satisfiable.}$$

where $\text{INVERT}(f_1, f_2)$ inverts f_2 and uses them to replace the contradictory clauses in f_1.

In addition, Z3 returns a trace $\tau_{t,p}$ used to assign one of $\{\text{SATISFIED}, \text{VIOLATED}, \text{DEFERRED}\}$ for each eligibility criterion in C_t. The criteria under the DEFERRED category correspond to the residual constraints.

Definition 4. The LLM-based

$$\text{VERBALIZE}(\{v, \tau, \rho, \Delta, L\}) = \{v, r\}$$

returns both the overall verdict v, and a detailed rationale r in natural language so a clinician can understand the full decision procedure. If the verdict v is ELIGIBLE, the rationale includes why and the residual constraints that must be satisfied if any. Otherwise, the verbalizer describes the reasons why the patient is ineligible by articulating the flip set. It also describes the status of each eligibility constraint in the trial for clarity. □

Implementation details of SMTMATCH are provided in Appendix A.

5 Experiments

We now present our evaluation on (1) precision and recall, (2) usefulness of the rationale, and (3) counterfactual self-faithfulness.

5.1 Matchers Evaluated

We compare five matchers on the same patient charts, eligibility criteria, and large language model backbones. For each matcher with multiple variants, we report the best-performing variant on a held-out development split.

SMTMATCH (ours). SMTMATCH, described in §4, uses SMT constraints to make the final eligibility decision. When a load-bearing criterion depends on a predicate the patient-side parser leaves unbound, SMTMATCH records it as an explicit assumption or residual constraint. Unlike the LLM-based matchers, SMTMATCH produces an SMT-backed verdict, criterion-level statuses, explicit assumptions, and residual constraints.

LLMMATCH. A single LLM prompt emits a verdict and free-text rationale from the patient record and trial. We experimented with four prompting variants — *think-then-aggregate*, *single-step*, *aggregation-first*, and *blocker-then-verdict* — and found *think-then-aggregate* to be the strongest. Here, the model first reasons through the patient's clinical state before committing to a verdict.

CoT LLM. Same as LLMMATCH, but the model enumerates each criterion with a per-criterion judgment before aggregating into a verdict. We use the best-performing variant from our prompt search (Appendix C).

TrialGPT-Matching [9]. We use TrialGPT's matching component, which classifies each criterion as *included*, *not included*, *not applicable*, *excluded*, or *not excluded*, but does not output a

patient-level verdict. We aggregate its outputs with the most favorable rule: eligible iff no inclusion criterion is labeled as *not included* and no exclusion criterion is labeled as *excluded*.

ZSPM (Zero-Shot Patient Matching) [20]. ZSPM produces, for each eligibility criterion, either a *met* or *unmet* judgment, and outputs an overall ternary patient verdict: {*not eligible*, *might be eligible*, *eligible*}. For matching metrics, we map *might be eligible* and *eligible* to ELIGIBLE, which is the recall-favorable binary mapping, and map *not eligible* to INELIGIBLE. For counterfactual analysis (§5.5), the binary *met*/*unmet* flag is insufficient, so we use the internal confidence classification: {*deterministic met*, *deterministic unmet*, *defer*}.

Implementation details for SMTMATCH are provided in Appendix A; details for the remaining matchers are provided in Appendix B.

5.2 Dataset and Reference Evaluation

We evaluate on patient–trial pairs derived from the SIGIR patient–trial matching benchmark [10], which contains 3,621 ClinicalTrials.gov trials and 59 synthetic patient cases. Because SIGIR provides retrieval relevance judgments rather than patient-level eligibility labels, we use it only to source challenging candidate pairs; Appendix J compares SIGIR relevance with our reference eligibility labels. For each patient, we use TrialGPT-Retrieval [9] only to retrieve the top-10 candidate trials, yielding 590 patient–trial pairs.

Reference evaluation panel. Because eligibility labels depend on how ambiguity and missing evidence are handled, we derive reference labels using a five-judge GPT-5 panel. Each judge uses a different framing—pragmatic screening, protocol adherence, schema-based review, checklist-driven scoring, or explanation-focused assessment—and independently labels each anonymized patient–trial pair as eligible, ineligible, or inconclusive after seeing standardized matcher rationales in randomized order. We exclude pairs marked inconclusive by any judge, yielding 552 decisive pairs, and use the majority verdict as the reference label. Appendices H and F describe the label derivation process and show that the exclusions are not concentrated at particular retrieval ranks.

Why a panel is not a single GPT-5 matcher. The reference is a five-framing consensus rather than one GPT-5 eligibility call. Among the 552 decisive pairs, only 445 (80.6%) receive a unanimous 5–0 verdict; the remaining 107 (19.4%) split 4–1 or 3–2, so individual GPT-5 framings often differ from the majority. Consistently, the strongest GPT-5 LLMMATCH matcher reaches $F_1 = 0.91$ against the panel, not 1.0. The reference is therefore a consensus signal, not a single-call GPT-5 label.

Why GPT-5 is omitted from the main accuracy comparison. We omit GPT-5 systems to avoid self-preference: a GPT-5 matcher shares the reference panel's backbone and may match its verdict style for reasons unrelated to matching quality. Thus, the main comparison uses GPT-4.1 and smaller backbones; GPT-5 results appear only in the appendix with this caveat.

Clinician validation. Because panel labels may be noisy and clinician review is costly, we validated the reference labels on a targeted sample enriched for difficult cases. We sampled two 16-pair sets, comparing SMTMATCH against LLMMATCH and ZSPM, with equal coverage of cases where both systems, only SMTMATCH, only the comparator, or neither system matched the panel.

A clinician reviewed 30 completed patient–trial pairs, adjudicating disagreements from the chart, criteria, panel label, judge explanations, and system rationales. Only one case remained a clear panel–clinician disagreement; ten were clinically ambiguous. Ambiguous cases counted as half agreement yield Cohen's $\kappa = 0.60$ on the reviewed sample and 0.67 after corpus reweighting; counted as consistent with the panel's prescreening interpretation, they yield $\kappa = 0.93$ and 0.98. Thus, most disagreements reflect borderline eligibility rather than clear panel errors. Full details are in Appendix I.

5.3 Precision and Recall

We first evaluate each matcher on the 552 clinician-validated patient–trial pairs from Section 5.2, reporting accuracy, precision, recall, and F1.

Table 1 compares SMTMATCH with LLM-based matching baselines across four backbones. Overall, SMTMATCH performs strongly and consistently across model families. With GPT-4.1, **SMTMATCH achieves the best F1 score** of 0.900, improving over the strongest LLM-only baseline (LLMMATCH, 0.884) while maintaining both high precision (0.927) and high recall (0.874). This suggests that the formal matching layer can improve decision quality even when the underlying backbone is already strong. The gain is more

Table 1: Accuracy by GPT backbone against the 5-judge reference. Best F1 per backbone in **bold**. CIs and clinician-adjusted F1: App. K, Table 14.

LLM	System	F1	P	R	Acc
4.1	SMTMATCH	**0.900**	0.927	0.874	0.902
	LLMMATCH	0.884	0.850	0.921	0.879
	CoT LLM	0.721	0.964	0.576	0.775
	TrialGPT	0.359	0.863	0.227	0.592
	ZSPM	0.835	0.806	0.867	0.828
4o	SMTMATCH	**0.836**	0.797	0.878	0.826
	LLMMATCH	0.769	0.938	0.651	0.803
	CoT LLM	0.773	0.929	0.662	0.804
	TrialGPT	0.759	0.817	0.709	0.774
	ZSPM	0.806	0.779	0.835	0.797
4o-mini	SMTMATCH	**0.754**	0.802	0.712	0.766
	LLMMATCH	0.476	0.957	0.317	0.649
	CoT LLM	0.599	0.912	0.446	0.699
	TrialGPT	0.588	0.861	0.446	0.685
	ZSPM	0.691	0.543	0.950	0.572

pronounced for GPT-4o and GPT-4o-mini: SMTMATCH improves F1 from 0.806 to 0.836 over the strongest GPT-4o baseline and from 0.691 to 0.754 over the strongest GPT-4o-mini baseline. These results indicate that explicit constraint reasoning is especially helpful for weaker or less stable backbones, where end-to-end LLM matchers often trade off precision and recall unevenly.

5.4 Rationale Preference

Our second experiment evaluates rationale usefulness for clinical prescreening. We compare SMTMATCH with the strongest non-SMTMATCH matchers, LLMMATCH and ZSPM, using GPT-4.1, on the same clinician-audited pairs from Section 5.2: 16 pairs per head-to-head comparison.

For each pair, the clinician views the chart, trial criteria, and two anonymous randomized rationales, then selects the more useful one or marks a tie. We report head-to-head counts and tie-adjusted win rate. The clinician also rates each rationale on five 1–5 axes: *actionability*, *criterion completeness*, *chart traceability*, *logical consistency*, and *overall support*. These axes assess whether the rationale tells clinicians what to check at the prescreening visit, covers the trial criteria decisive to the verdict, grounds cited facts in verifiable chart evidence, aggregates its bullet-level conclusions consistently to the stated verdict, and credibly supports that verdict overall. Appendix I and D.4 describe sampling, reweighting, and confidence intervals. Concrete examples of each system's user-facing rationales can be found in Appendix E.

Table 2: Pairwise clinician preference over rationales, comparing SMTMATCH against each baseline. Columns 2–4 give SMTMATCH wins, ties, and baseline wins; the last column reports SMTMATCH's tie-adjusted win rate.

Baseline	SMTMATCH	Tie	Base.	Win rate
ZSPM	13	3	0	**90.6**
LLMMATCH	11	2	3	75.0

Table 3: Per-axis rationale review on 16 audited pairs per comparator (1–5 Likert). App. K, Table 24 for details.

Axis	SMTMATCH	Comp.	Win rate
SMTMATCH *vs.* ZSPM			
Actionability	5.00	3.12	100%
Criterion completeness	5.00	4.69	62%
Chart traceability	5.00	4.25	84%
Logical consistency	5.00	3.62	88%
Support	4.94	3.81	97%
SMTMATCH *vs.* LLMMATCH			
Actionability	4.69	3.12	94%
Criterion completeness	4.81	3.75	84%
Chart traceability	4.81	3.88	81%
Logical consistency	4.69	4.31	62%
Support	4.62	4.00	75%

Clinicians prefer SMTMATCH's rationales. Tables 2 and 3 show that the clinician strongly prefers SMTMATCH's rationales. Against ZSPM, SMTMATCH wins 13 of 16 comparisons and ties 3, yielding a 90.6% win rate with no ZSPM wins. Against LLMMATCH, SMTMATCH wins 11 of 16 comparisons and ties 2, for a 75.0% win rate.

The axis ratings suggest that these preferences are driven by reviewability rather than verdict agreement alone. SMTMATCH receives higher mean scores on every axis against both comparators, with the largest gains in actionability, chart traceability, criterion completeness, and support. Thus, clinicians find SMTMATCH's solver-backed rationales more actionable, better grounded in the chart, and easier to review.

5.5 Counterfactual Self-Faithfulness

Counterfactual construction. For each rejected patient–trial pair, we construct a counterfactual chart by extracting the matcher's cited rejection reasons (i.e., the flip set), minimally editing the patient record to address those reasons while preserving other eligibility-relevant facts, validating the edit with a separate LLM, and re-running the same matcher on the modified chart.

LLMMATCH-FlipExplicit uses the same LLMMATCH matcher, but asks the model to explicitly

Table 4: Counterfactual self-faithfulness (GPT-5 modifier & validator). *Raw flip rate* = fraction of validator-accepted CFs the matcher re-judges as eligible (higher is better). Backbone = matcher's judge and rejudge LLM. More results in Appendix K Table 15 and Table 25; clinician audit results in Table 18.

System	Backbone	Raw flip rate
LLMMATCH	GPT-4.1	48.9%
CoT LLM	GPT-4.1	65.0%
CoT LLM	GPT-5	59.5%
LLMMATCH-FlipExplicit	GPT-4.1	57.7%
TrialGPT	GPT-4.1	61.1%
ZSPM	GPT-4.1	26.6%

list the unmet requirements whose change would make the patient eligible; these listed flip targets are used for counterfactual construction.

A final validation step checks that the edit is clinically coherent and does not introduce unrelated eligibility changes or new blockers.

The raw **self-faithfulness rate** is the fraction of validator-accepted counterfactuals on which the matcher changes its verdict to ELIGIBLE:

$$\mathrm{SF} = \frac{|\{p : V(p) = 1 \wedge \hat{y}'(p) = \text{ELIGIBLE}\}|}{|\{p : V(p) = 1\}|},$$

where $V(p) = 1$ indicates that the validator accepts the modification and $\hat{y}'(p)$ is the matcher's verdict on the modified chart. Higher values indicate that the cited rejection reasons are more actionable and better aligned with the matcher's own decision boundary. The metric tests whether the cited blockers are sufficient as an actionable explanation for rejection; it does not require each cited blocker to be individually necessary.

We report multiple modifier–validator configurations over validator-accepted counterfactuals. Because LLM validators can err, a clinician also audits a stratified subsample. We estimate validity separately for flipped and non-flipped cases and report a clinician-adjusted self-faithfulness rate; details are in Appendix D. The audit finds that roughly 80% of counterfactual modifications are valid (Table 17), which is reasonable given the task's difficulty.

Unlike the main accuracy comparison, this experiment can include GPT-5-backed matchers because it tests whether each matcher flips under counterfactual edits to its own cited blockers, not agreement with a GPT-5 reference panel. Thus, the GPT-5 CoT result in Table 4 measures self-consistency rather than agreement with a shared GPT-5 judge.

SMTMATCH is self-faithful by construction. In SMTMATCH, rejection reasons are given by the flip set: the minimal patient facts or assumptions whose change makes the constraints satisfiable. Thus, a counterfactual constructed from this flip set should be judged eligible by construction, except when errors occur in counterfactual rewriting or semantic parsing. SMTMATCH achieves an 84.3% flip rate; the remaining 15.7% therefore reflects pipeline errors rather than a mismatch between the rationale and the formal decision boundary.

LLM rationales often fail counterfactual self-faithfulness. Table 4 shows that LLM rationales often identify blockers that are not sufficient to change the model's own verdict. The strongest baseline reaches only 65.0%, while LLMMATCH reaches 48.9%; even prompting LLMMATCH to list unmet requirements improves the flip rate only to 57.7%. The GPT-5 CoT matcher also remains below GPT-4.1 CoT, at 59.5%, showing that this failure is not removed by a stronger backbone. These results suggest that natural-language rationales can sound plausible while still failing to identify the reasons that actually drive the model's rejection.

Detailed results on the three experiments described above can be found in Appendix K.

6 Conclusion

Clinical trial matching systems should be judged not only by accuracy, but by whether clinicians can understand, verify, and correct their decisions. Our experiments show that strong LLM matchers can be accurate while still producing rationales that miss key rejection reasons, hide missing evidence, or fail to identify what would make a patient eligible.

SMTMATCH narrows this gap by separating language understanding from eligibility reasoning. Rather than asking an LLM for verdict and explanation, SMTMATCH grounds each decision in explicit constraints whose satisfied criteria, violations, unresolved assumptions, and rejection reasons can be inspected. This structure makes failures local: clinicians can trace them to specific predicates, assumptions, or criteria. Text-reading errors remain possible, but each decision's basis becomes visible.

The broader lesson is that high-stakes clinical matching should not rest on a single natural-language judgment. LLMs are powerful readers of clinical text, but their decisions become trustworthy only when tied to structures clinicians can examine, challenge, and correct.

01 Limitations

This paper has several limitations. First, although our system makes the final matching decision explicit and rerunnable, it still relies on LLMs to read the trial criteria and patient notes. If the LLM misses a fact, extracts the wrong evidence, or represents a requirement incorrectly, the final decision can still be wrong. Our formal reasoning step makes the decision process easier to inspect and control, but it does not remove all errors from clinical text understanding.

Second, our counterfactual study uses LLMs to interpret the residual constraints and rewrite patient notes, targeting the smallest set of patient facts or assumptions whose change would make the patient satisfy the trial formula. To assess these rewrites, we ask clinicians whether each edited note addresses the targeted facts or assumptions, avoids adding new reasons for rejection, and remains clinically realistic and coherent. However, broader clinician review at larger scale would provide stronger evidence.

Third, we use the counterfactual experiment to probe both trustworthiness and one aspect of steerability: whether a system can expose changes that would turn an initially ineligible case into an eligible one. This does not cover the reverse direction, since making an eligible patient ineligible is rarely useful in trial screening. It also does not fully test other forms of steering, such as changing how the system treats missing information, disease hierarchies, time windows, or conflicting evidence.

Fourth, our benchmark is curated and limited in scale. Existing retrieval benchmarks such as SIGIR [10] and TREC [14] say which trials are relevant, but not whether the patient truly satisfies each trial's eligibility rules. The n2c2 cohort-selection dataset is closer to our task, but covers only one trial and is no longer easy to access. We therefore evaluate on curated patient–trial pairs from the top-ranked SIGIR candidates. These are meaningful cases for fine-grained matching, but they do not cover the full trial corpus.

Finally, the patient notes in our benchmark are synthetic. This avoids privacy barriers and allows controlled evaluation, but may not capture the full complexity of real clinical notes, which can be longer, noisier, incomplete, and spread across many visits. Testing on real clinical notes under proper privacy safeguards remains important future work.

02 Ethical considerations.

Clinical trial matching is a high-stakes task: incorrect recommendations may affect patient access to therapy, clinician workload, and trial recruitment. Our system is intended as a decision-support tool for surfacing candidate matches, assumptions, and residual eligibility constraints, not as an autonomous enrollment or exclusion system. Final eligibility decisions should remain under qualified clinical and trial-site review.

Our framework reduces some risks of end-to-end LLM matching by separating clinical text interpretation from the final eligibility decision. The solver makes the encoded decision explicit, rerunnable, and interpretable. However, the system still relies on LLMs to parse trial criteria and patient records; missed evidence, incorrect extraction, or faulty formalization can still lead to wrong verdicts. Formal reasoning improves interpretability of the encoded policy, but does not guarantee clinical correctness.

Our evaluation uses synthetic patient notes from the SIGIR benchmark [10], which reduces privacy risks because no real patient records are used, but may not capture the full complexity, incompleteness, and documentation bias of real electronic health records. Deployment on real patient data would require institutional review, privacy-preserving data handling, prospective validation, and monitoring for subgroup-specific errors, especially for patients whose records are incomplete or whose conditions are documented using nonstandard terminology.

There are also risks of overreliance. Structured rationales and solver traces may appear authoritative even when upstream extraction errors are subtle. Deployed systems should expose supporting evidence, uncertainty flags, and deferred criteria, and should not be used to deny patients clinician review without human verification or appeal.

References

[1] Clark Barrett and Cesare Tinelli. 2018. Satisfiability modulo theories. In *Handbook of model checking*, pages 305–343. Springer.

[2] Nikolaj Bjørner, Anh-Dung Phan, and Lars Fleckenstein. 2015. νz-an optimizing smt solver. In *International conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 194–199. Springer.

[3] Leonardo De Moura and Nikolaj Bjørner. 2008. Z3: An efficient smt solver. In *International conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 337–340. Springer.

[4] Scott L Fleming, Alejandro Lozano, William J Haberkorn, Jenelle A Jindal, Eduardo Reis, Rahul Thapa, Louis Blankemeier, Julian Z Genkins, Ethan Steinberg, Ashwin Nayak, and 1 others. 2024. Medalign: A clinician-generated dataset for instruction following with electronic medical records. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22021–22030.

[5] Megan E Gregory, Elise Russo, and Hardeep Singh. 2017. Electronic health record alert-related workload as a predictor of burnout in primary care providers. *Applied clinical informatics*, 8(03):686–697.

[6] Shashi Gupta, Aditya Basu, Mauro Nievas, Jerrin Thomas, Nathan Wolfrath, Adhitya Ramamurthi, Bradley Taylor, Anai N Kothari, Regina Schwind, Therica M Miller, and 1 others. 2024. Prism: Patient records interpretation for semantic clinical trial matching system using large language models. *NPJ digital medicine*, 7(1):305.

[7] Simeng Han, Hailey Schoelkopf, Yilun Zhao, Zhenting Qi, Martin Riddell, Wenfei Zhou, James Coady, David Peng, Yujie Qiao, Luke Benson, and 1 others. 2024. Folio: Natural language reasoning with first-order logic. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22017–22031.

[8] Alexey Ignatiev, António Morgado, and João Marques-Silva. 2019. Rc2: an efficient maxsat solver. *Journal on Satisfiability, Boolean Modelling and Computation*, 11(1):53–64.

[9] Qiao Jin, Zifeng Wang, Charalampos S Floudas, Fangyuan Chen, Changlin Gong, Dara Bracken-Clarke, Elisabetta Xue, Yifan Yang, Jimeng Sun, and Zhiyong Lu. 2024. Matching patients to clinical trials with large language models. *Nature communications*, 15(1):9074.

[10] Bevan Koopman and Guido Zuccon. 2016. A test collection for matching patients to clinical trials. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 669–672.

[11] K Lee, Y Mai, Z Liu, K Raja, T Jun, M Ma, T Wang, L Ai, E Calay, W Oh, and 1 others. 2024. Criteriamapper: establishing the automatic identification of clinical trial cohorts from electronic health records by matching normalized eligibility criteria and patient clinical characteristics. *Scientific Reports*, 14(1):25387.

[12] Mingyu Lee, Kyuri Kim, Yoojin Shin, Yoonji Lee, and Tae-Jung Kim. 2025. Advancements in electronic medical records for clinical trials: Enhancing data management and research efficiency. *Cancers*, 17(9):1552.

[13] Arjun Panickssery, Samuel R Bowman, and Shi Feng. Llm evaluators recognize and favor their own generations, 2024. *URL https://arxiv.org/abs/2404.13076*.

[14] Kirk Roberts, Dina Demner-Fushman, Ellen M Voorhees, Steven Bedrick, and William R Hersh. 2022. Overview of the trec 2022 clinical trials track. In *TREC*.

[15] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, and 1 others. 2023. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180.

[16] SNOMED International. 2026. SNOMED CT. https://www.snomed.org/. Accessed: 2026-05-24.

[17] Dmitriy Umerenkov, Galina Zubkova, and Aleksandr Nesterov. 2023. Deciphering diagnoses: How large language models explanations influence clinical decision making. *arXiv preprint arXiv:2310.01708*.

[18] David Wendler. 2021. The inevitability and ethics of inaccurate screening in clinical trials: A call for research and guidance. *Ethics & human research*, 43(4):37–44.

[19] Edwin B Wilson. 1927. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212.

[20] Michael Wornow, Alejandro Lozano, Dev Dash, Jenelle Jindal, Kenneth W Mahaffey, and Nigam H Shah. 2025. Zero-shot clinical trial patient matching with llms. *NEJM AI*, 2(1):AIcs2400360.

[21] Xi Ye, Qiaochu Chen, Isil Dillig, and Greg Durrett. 2023. Satlm: Satisfiability-aided language models using declarative prompting. *Advances in Neural Information Processing Systems*, 36:45548–45580.

[22] Chi Yuan, Patrick B Ryan, Casey Ta, Yixuan Guo, Ziran Li, Jill Hardin, Rupa Makadia, Peng Jin, Ning Shang, Tian Kang, and 1 others. 2019. Criteria2query: a natural language interface to clinical databases for cohort definition. *Journal of the American Medical Informatics Association*, 26(4):294–305.

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

[24] Cyrus Zhou, Yufei Jin, Yilin Xu, Yu-Chiang Wang, Chieh-Ju Chao, and Monica S Lam. 2026. Scalable high-recall constraint-satisfaction-based information retrieval for clinical trials matching. *arXiv preprint arXiv:2604.08849*.

Appendix Roadmap

Appendix	Contents
Part I — System and methodology	
A	SMTMATCH pipeline: parser, value miner, arbiter, verbalizer, and Z3 program.
B	Baseline implementations and aggregation rules.
C	Prompting-strategy search for LLMMATCH and CoT LLM.
D	Counterfactual modifier, validator, and clinician audit.
E	Native rationale form per system on two example pairs.
Part II — Data and reference labels	
F	Dataset statistics: patients, trials, retrieval-derived pairs.
G	Descriptive statistics: per-system verdict balance, strata, audit composition.
H	Reference-label derivation from the five-judge LLM panel.
I	Clinician validation of reference labels and agreement analysis.
J	Comparison between our eligibility labels and SIGIR referral qrels.
Part III — Detailed results	
K	Cross-backbone accuracy, clinician-adjusted F1, and pairwise pop-recon.
Part IV — Responsible NLP and reproducibility	
L	API-call budget, wall-clock, infrastructure.
M	Pinned package versions, sampling parameters, seeds.
N	Human-subjects status of the clinician audit.
O	Clinician recruitment, payment, and burden.
P	Data consent and corpus licensing.
Q	Use of artifacts, licenses, and intended use.
R	Use of AI assistants in drafting the paper.

Roadmap to the appendix.

A System Modules and Prompts

This appendix documents the modules of SMTMATCH and reproduces the prompts used to produce the results in the paper. We follow the notation of §4: the trial parser TPARSE, the patient parser PPARSE, the SMT/MaxSAT solver SOLVE, and the LLM verbalizer VERBALIZE. LLM-driven stages use `gpt-4.1` as the backbone unless otherwise noted; the solver itself is deterministic. Each prompt uses placeholder tokens, such as `{{PATIENT_NOTES}}` and `{{VARIABLE_LIST}}`, that are substituted at call time.

A.1 Pipeline and Prompt Inventory

For each patient–trial pair, SMTMATCH first compiles the trial criteria into formal constraints, then extracts patient facts only for the atoms required by those constraints, evaluates the combined formula with an SMT/MaxSAT solver, and verbalizes the solver output into a clinician-readable rationale. Table 5 lists the pipeline stages and the prompts they use. Stages that are deterministic use no prompt.

Stage	Module	Prompt
1	Leaf collection for trial criteria	—
2	Trial scope mining in TPARSE	§A.2.1
3	Trial projection rewriting in TPARSE	§A.2.2
4	Patient value mining in PPARSE	§A.3.1, A.3.2
5	Alias remapping	—
6	SMT/MaxSAT evaluation with SOLVE	—
–	Extraction-resolution arbiter	§A.4.1
–	Natural-language verbalizer VERBALIZE	§A.5.1

Table 5: Pipeline stages and prompts. Stages 1, 5, and 6 are deterministic. The arbiter is invoked only when extracted patient facts conflict with auxiliary definitional constraints rather than with eligibility requirements.

A.2 Trial Parser TPARSE

The trial parser TPARSE compiles the natural-language eligibility criteria of a trial t into the trial constraint formula $C_t = \text{TPARSE}(t)$. We build on the SATIR trial-side semantic parser of Zhou et al. [24], which performs the underlying criterion-to-SMT compilation and canonicalizes clinical terms using the SNOMED ontology. In SMTMATCH, we adapt this parser to the matching setting by adding two patient-independent stages that specialize the trial atoms for downstream patient-fact extraction. Because these stages depend only on the trial, their outputs are cached per trial and reused across all patients matched against that trial.

Scope miner. The scope miner classifies each atomic predicate in $\text{ATOM}(C_t)$ as either *in scope*, meaning that it describes a checkable current or historical clinical state of the patient, or *projected away*, meaning that its main content belongs to a dimension we do not ask the patient parser to extract directly, such as evidence strength or the criterion of determination. This implements the state-projection restriction used by SMTMATCH: PPARSE is only asked about predicates that can be checked against the patient record.

A.2.1 Scope-miner prompt

```
# === ROLE ===
You are an expert clinician who excels in logic.

# === TASK ===
For each variable below, decide whether it is:
- "in_scope": it constrains the patient's current or past clinical
↪  state in a meaningful way
- "projected_away": its primary content belongs to a non-state
↪  dimension that we do not preserve for this task

We only care about the projection of the eligibility criteria onto the
↪  patient's current and past clinical state. Note that we also do not
↪  care about the patient's race, ethnicity, or geographical location.
↪  Also, don't project away criteria that are correlated with the
↪  patient's clinical state as well - for example, it is unlikely that
↪  a 43-year individual is a high school student, so don't project the
↪  requirement of being a high school student away.

A variable can be clinically important in practice yet still be
↪  "projected_away" if its primary content is documentation,
↪  confirmation pathway, logistics, scheduling, care-process
↪  structure, willingness, site, geography, or another non-state
↪  dimension rather than substantive patient state.

Conversely, do not automatically project away a variable merely because
↪  it contains words like "planned", "scheduled", or similar
↪  future-oriented language. If the variable's meaning reflects a
↪  substantive current clinical or treatment-status constraint on the
↪  patient, it can still be "in_scope".

Variables are usually "projected_away" if their primary content is
↪  about:
- documentation status
- confirmation strength
- source procedure
- discovery pathway
- assessment pathway
- logistics
- willingness
- scheduling
- geography
- site
- clinic membership
- care setting
- hospitalization
- existing clinical-trial enrollment
- race and ethnicity
- geographical location
- hospital admission intent

Important:
- Apply this to both stem variables and qualifier variables.
- A stem variable may be "in_scope" while one of its qualifiers is
↪  "projected_away".
- Use the provided meaning rather than guessing from the variable name
↪  string.
- If unsure, prefer "in_scope".
- Base your scope decision on the variable meaning, not the variable
↪  name string.
- Use the variable name only as a fallback if the meaning is missing or
↪  clearly unusable.
- If the variable name and the provided meaning suggest different
↪  interpretations, follow the provided meaning.
- Do not project away a variable merely because it includes some
↪  auxiliary non-state framing, as long as its primary content still
↪  constrains substantive patient state.

# === GUIDELINES ===
Project away a variable only when its primary content is completely
↪  about one or more non-state dimensions listed above.

Do not project away a variable merely because it includes some
↪  auxiliary non-state framing. If the variable still directly
↪  constrains the patient's substantive current or past clinical
↪  state, it should remain "in_scope".

In other words:
- Keep variables whose core meaning answers: "What current or past
↪  clinical state does the patient have, or what substantive
↪  treatment/intervention state is the patient in?"
- Project away variables whose core meaning answers: "How was this
↪  documented, confirmed, discovered, assessed, arranged, located,
↪  intended, or operationalized?"

A current or past intervention can be "in_scope" when it is part of the
↪  patient's actual current or past clinical state.

An actively planned or scheduled intervention can also be "in_scope"
↪  when it reflects a substantive current treatment-status constraint
↪  on the patient rather than mere administrative scheduling.

Project away planned/scheduled variables only when their primary
↪  content is merely logistical or administrative scheduling rather
↪  than a meaningful clinical or treatment-status constraint on the
↪  patient.

For qualifiers, judge the qualifier's own dimension:
- If the qualifier adds a substantive patient-state constraint, mark it
↪  "in_scope".
- If the qualifier mainly adds documentation, confirmation, procedural,
↪  logistical, site, scheduling, willingness, or similar non-state
↪  structure, mark it "projected_away".

Elective-status qualifiers are often "projected_away" when they
↪  primarily reflect urgency, care-process structure, or logistics
↪  rather than substantive patient state.

# === OUTPUT FORMAT ===
Return ONLY a JSON object wrapped in <variable_scope_values> ...
↪  </variable_scope_values>.

Example:
<variable_scope_values>
{
  "patient_has_diagnosis_of_copd_now": {
    "scope": "in_scope",
    "reason": "Substantive current clinical state."
  },
  "patient_has_diagnosis_of_copd_now@@documented": {
    "scope": "projected_away",
    "reason": "Documentation dimension rather than patient state."
  }
}
</variable_scope_values>

Additional example:
<variable_scope_values>
{
  "patient_has_spinal_arthrodesis_scheduled": {
    "scope": "in_scope",
    "reason": "Being scheduled for spinal arthrodesis reflects a
   ↪  substantive current clinical/treatment state rather than mere
   ↪  administrative scheduling."
  },
  "patient_has_spinal_arthrodesis_scheduled@@elective": {
    "scope": "projected_away",
    "reason": "Elective status primarily reflects urgency/logistics
   ↪  rather than substantive patient state."
  },
  "patient_is_undergoing_spinal_arthrodesis_now": {
    "scope": "in_scope",
    "reason": "A procedure the patient is currently undergoing is part
   ↪  of the patient's present clinical state."
  },
  "patient_is_undergoing_spinal_arthrodesis_now@@elective": {
    "scope": "projected_away",
    "reason": "Elective status primarily reflects urgency/logistics
   ↪  rather than substantive patient state."
  }
}
</variable_scope_values>

# === INPUTS ===
<variable_list>
{{VARIABLE_LIST}}
</variable_list>
```

Projection rewriter. The projection rewriter turns each in-scope predicate into the version consumed by PPARSE: a typed signature together with a self-contained, human-readable definition of what should be checked in the patient chart. This stage ensures that every predicate passed to the patient parser has an unambiguous chart-checkable meaning.

A.2.2 Projection-rewriter prompt

```
# === ROLE ===
You are an expert clinician and semantic parser helping rewrite SMT
↪  variable metadata for patient-note value extraction.

# === TASK ===
You are given SMT variable metadata.
Conservatively rewrite each variable into the projection of the
↪  eligibility criteria onto the patient's current or past clinical
↪  state.

For each variable:
- minimally rewrite the variable name so it reflects only the surviving
↪  patient-state proposition
- rewrite the meaning accordingly
- project away wording that is not part of the substantive
↪  patient-state proposition
- keep the rewrite conservative and as close as possible to the
↪  original variable identity

This is a projection rewrite, not a broad reinterpretation.

# === CORE PRINCIPLE ===
Keep only the substantive proposition about the patient's current or
↪  past clinical state.
Project away wording that is not part of that substantive patient-state
↪  proposition.

This mainly applies when a variable is only partially burdened by
↪  non-substantive framing.
Do not treat a variable as removable just because it contains some
↪  diagnostic, methodological, documentation, or operational wording.
When a clinically meaningful patient-state proposition remains,
↪  preserve that proposition and remove only the extra framing.

Project away the following when they are not themselves the substantive
↪  patient-state content:
- diagnosis wording
- documentation wording
- confirmation strength
- named determination methods
- named guideline / criteria / scoring / classification / staging
↪  framework wording when it is only the method of assigning a
↪  clinically meaningful state
- source procedure
- discovery pathway
- operational / administrative framing
- healthcare-record structure
- timing defined only relative to study operations rather than
↪  patient-state time

# === IMPORTANT SPECIAL RULE FOR DISEASE-STATE VARIABLES ===
- Wording such as “has diagnosis of X” should usually be interpreted as
↪  the patient clinically having X.
- Do not preserve diagnostic-label wording as an extraction
↪  requirement.
- Do not let explicit-documentation burden survive in the rewritten
↪  meaning.

# === IMPORTANT RULE FOR CLINICALLY SUBSTANTIVE CATEGORIES ===
```

```
- If a variable includes a clinically substantive class, stage, grade,
↪  severity level, risk tier, threshold category, or other
↪  categorization that changes the actual patient-state proposition,
↪  preserve that content.
- Project away the determination framework or explicit documentation
↪  burden, but do NOT project away the substantive category itself.
- Do not reduce a specifically categorized variable to a more generic
↪  variable if the category changes the meaning.

# === IMPORTANT CONSTRAINTS ===
- Be conservative.
- Do NOT broaden the variable beyond its substantive patient-state
↪  core.
- Do NOT weaken clinically meaningful polarity, temporality, anatomy,
↪  count/threshold, measurement, severity, family-history content, or
↪  true outcome/state content.
- Do NOT invent a patient-state proposition if the variable is actually
↪  purely operational or purely documentation-based.
- Threshold helpers should be rewritten as direct yes/no comparison
↪  propositions.
- If the original variable is already close to the correct projected
↪  patient-state form, make only minimal changes.

# === IMPORTANT STEM / QUALIFIER CONSISTENCY RULE ===
Variables are grouped by stem.
- Rewrite the stem conservatively.
- For qualifier variables, keep them semantically consistent with the
↪  rewritten stem.
- Do not independently invent a different stem meaning for a qualifier.
- If a qualifier variable has original form stem@@qualifier, its
↪  rewritten variable name should still be a qualifier form that
↪  corresponds to the rewritten stem.

# === VARIABLE-NAME RULES ===
- The rewritten variable name must stay in SMT-style snake_case.
- Use lowercase and underscores only.
- Do NOT output readable English phrases with spaces.
- Change the variable name minimally.
- Only rewrite the name when the current name contains content that
↪  should be projected away or is no longer representative.
- Preserve clinically meaningful polarity, temporality, anatomy,
↪  thresholds, counts, measurements, severity, family-history scope,
↪  and event/state distinctions whenever they survive projection.
- Prefer minimal edits such as removing
↪  diagnosis/documentation/method/source-procedure fragments rather
↪  than renaming from scratch.

Examples:
- patient_has_diagnosis_of_copd_now -> patient_has_copd_now
- patient_has_pneumonia_determined_by_admitting_physician_now ->
↪  patient_has_pneumonia_now
- patient_has_copd_determined_by_gold_criteria_now ->
↪  patient_has_copd_now
- patient_has_gold_stage_3_copd_now ->
↪  patient_has_gold_stage_3_copd_now
- patient_has_nyha_class_ii_heart_failure_determined_by_nyha_criteria
↪  _now -> patient_has_nyha_class_ii_heart_failure_now
- patient_has_at_least_5_rectal_polyps_each_ge_2_mm_now -> keep if
↪  already representative
- patient_was_identified_during_screening_colonoscopy_as_having_recta
↪  l_polyps -> patient_has_rectal_polyps

# === MEANING RULES ===
- Rewrite the meaning to match the rewritten patient-state proposition.
- The meaning should be explicit, clinically faithful, and
↪  extraction-oriented.
- It should preserve clinically meaningful time, anatomy, threshold,
↪  severity, family-history, event, and category content when
↪  relevant.
- It should not preserve projected-away diagnosis-label, documentation,
↪  determination-method, source-procedure, or operational framing
↪  unless those are themselves clinically substantive.
- Focus on removing only the extra framing, not the substantive
↪  patient-state content.


# === IMPORTANT RULE FOR TRIAL-RELATIVE CLINICAL-IMPACT QUALIFIERS ===
Do NOT automatically project away wording such as:
- may interfere with study drug evaluation
- may confound efficacy evaluation
- may affect safety assessment
- may interfere with trial participation
- may compromise endpoint interpretation
- may interfere with adherence or study procedures

When such wording is attached to a patient disease, condition, symptom,
↪  medication, allergy, or other clinical state, treat it as a
↪  potentially substantive qualifier rather than mere operational
↪  framing.

Projection rule:
- If the phrase only names an administrative workflow, documentation
↪  source, or study-time event, it may be projected away.
- But if the phrase expresses a clinically meaningful effect of the
↪  patient's state on eligibility-relevant trial assessment, safety,
↪  interpretability, participation, or treatment evaluation, it must
↪  be retained in the rewritten variable and meaning.
- In such cases, preserve the qualifier semantically, even if you
↪  minimally normalize its wording.

Do NOT collapse:
- patient_has_X@@may_interfere_with_study_drug_evaluation
into:
- patient_has_X

Do NOT collapse:
- patient_has_X@@other_than_insomnia_and_may_interfere_with_study_dru
↪  g_evaluation
into:
- patient_has_X@@other_than_insomnia

Instead, retain the full clinically load-bearing qualifier content.


# === OUTPUT FORMAT ===
Return ONLY one JSON object of the form:

<variable_projection_rewrites>
{
  "<original_var_name>": {
    "rewritten_variable_name": "<minimally rewritten SMT-style name>",
    "meaning": "<projected patient-state meaning>",
    "projection_summary": "<brief note explaining what was projected
    ↪  away, or what was retained>",
    "projection_status": "<unchanged or partially_projected>"
  },
  ...
}
</variable_projection_rewrites>

# === TRIAL CONTEXT ===
{{TRIAL_CONTEXT}}

# === VARIABLE GROUPS JSON ===
{{VARIABLE_GROUPS_JSON}}
```

A.3 Patient Parser PPARSE

Given a patient record p and the atoms of the trial formula, the patient parser emits patient-side constraints $C_p = \text{PPARSE}(\text{ATOM}(C_t), p)$. Unlike systems that extract all possible clinical facts from a record, PPARSE extracts only the facts relevant to the trial being matched. Each predicate is presented to the LLM with its typed signature and the self-contained definition produced by the projection rewriter. Predicates not supported by the record are left unbound.

Patient value mining is run separately for inclusion-side and exclusion-side predicates, so that each pass can apply side-appropriate handling of missing evidence. For numerical atoms, the parser considers both the underlying value and the trial-specific comparison. If the raw value is reported, it is extracted directly; if the value is absent but the chart contains threshold-level evidence, the parser may assess the comparison directly under the selected missing-evidence policy.

A.3.1 Inclusion-side value-miner prompt

```
# === ROLE ===
You are an expert clinician at PRESCREEN evaluating whether a patient
↪  meets the **INCLUSION** criteria of a clinical trial. The atomic
↪  variables you populate feed a deterministic Z3 solver that decides
↪  eligibility — assign each variable's value as a competent clinician
↪  would, using the same reasoning quality you would apply to a
↪  holistic eligibility judgment.

You are encoding ONLY the inclusion-side variables in this pass.

# === STAGE 1 — Read the chart and the trial holistically ===

Before populating any atom, internally:
1. Identify the trial's substantive intent (what kind of patient is
↪  this trial seeking?).
2. Extract every chart fact relevant to that intent and to the listed
↪  atomic variables. Use chart text directly; use clinical inference
↪  where defensible (e.g., "20 days post-mastectomy" implies cancer
↪  history; "post-stroke on antithrombotics" implies recent stroke
↪  event).
3. Identify which inclusion criteria the chart explicitly addresses
↪  (positively or negatively) and which it is silent on.

Do NOT skip Stage 1. It anchors every atom value to a coherent reading
↪  of the chart.

# === STAGE 2 — Populate atom values ===

For each atomic variable, decide its value following these rules:

## Decision rule for each atom (apply IN ORDER)

**Rules A and B always take precedence over the silence defaults and
↪  atom-class defaults below.** If the chart speaks to the atom, do
↪  not default — extract.

A. **Chart explicitly supports the atom's proposition** (verbatim text
↪  OR strong clinical inference):
   → set to TRUE (or the documented numeric value).
   This overrides the "leave NULL" categories below — even refinement
   ↪  qualifiers, even "remediable-by-action" atoms. If the chart
   ↪  provides positive evidence, assign TRUE.
   Example: atom is `patient_has_diagnosis_of_pulmonary_embolism_now@
   ↪  @provoked_by_reversible_major_risk_factor`. Chart says "patient
   ↪  gave birth 3 weeks ago" → recent postpartum is a reversible risk
   ↪  factor → assign **TRUE**, not NULL.
```

```
B. **Chart explicitly contradicts the atom's proposition**:
   → set to FALSE (or the documented contradicting numeric value).
   Same precedence: B overrides any "default NULL" rule.

C. **Chart is silent or genuinely ambiguous on the atom's
↪  proposition**:
   → only now apply the silence default + atom-class defaults below.

**Internal consistency**: your `assessment` field must agree with your
↪  `value` field. If your assessment says "assign TRUE", the value
↪  must be `true`. Never write a TRUE-justifying assessment alongside
↪  a NULL value.

## Inclusion-side silence default

**Default: silence → NULL.** Z3 treats NULL as compatible with TRUE or
↪  FALSE; the patient passes inclusion as long as the OTHER inclusion
↪  atoms are satisfied. This is the safe prescreen behavior —
↪  coordinator review at the visit resolves ambiguity.

**EXCEPTION (silence → FALSE) — when the atom is the trial's
↪  target-population-defining condition AND the chart context
↪  positively establishes the patient is OUTSIDE that population. Two
↪  patterns:**

**Pattern 1 — clearly different clinical domain.**
  (a) the atom is the trial's **target-population-defining condition**
  ↪  (e.g., `patient_has_breast_cancer` for a breast-cancer trial;
  ↪  `patient_has_history_of_mi` for a post-MI trial), AND
  (b) the chart describes a clinical scenario in a **clearly different
  ↪  clinical domain**, judged by clinical context (specialty /
  ↪  anatomic site / disease class) — NOT by lexical overlap of words.
  ↪  Examples:
    - Trial wants breast cancer; chart is "32yo for routine pap,
    ↪  otherwise healthy" → `patient_has_breast_cancer = FALSE`.
    - Trial wants 65yo with elbow injury; chart is "55yo with shortness
    ↪  of breath" → `patient_has_elbow_injury = FALSE`.
    - **Lexical-overlap trap**: trial wants children with **cutaneous**
    ↪  large congenital melanocytic nevi; chart describes 36yo with
    ↪  uterine *hydatidiform mole* (also called "vesicular mole").
    ↪  Both contain the word "mole" but they are clinically unrelated
    ↪  (dermatologic vs gynecologic). Skin-mole atoms
    ↪  (`has_finding_of_congenital_pigmented_melanocytic_nevus`) →
    ↪  **FALSE** because the chart's clinical domain is gynecology,
    ↪  not dermatology. Always classify by clinical context, not by
    ↪  token overlap.

**Pattern 2 — `_inthehistory` atom in an initial-presentation chart.**
  (a) the atom asks about a PRIOR finding / diagnosis / treatment
  ↪  (`_inthehistory` suffix, or `prior_X`, `previous_X`,
  ↪  `recurrent_X`, `failed_prior_treatment`), AND
  (b) the chart describes a NEW / INITIAL presentation of this clinical
  ↪  entity, with no prior workup / treatment / diagnosis documented
  ↪  (e.g., trial wants "recurrent GTT after prior mole evacuation";
  ↪  chart is "36yo with first-presentation suspicious mole, no prior
  ↪  evacuation"). In that case the chart positively establishes "this
  ↪  patient has no such history" → silence on the `_inthehistory`
  ↪  atom is informative → FALSE.

Do NOT fire FALSE under Pattern 1 when the chart shows symptoms/workup
↪  in the same family as the trial's CURRENT-state population (e.g.,
↪  trial wants active MI; chart is "chest pain + EKG changes, MI not
↪  yet diagnosed" → NULL or TRUE via inference, never FALSE — the
↪  chart is mid-workup of the same condition).

When in doubt between Patterns 1/2 and the conservative NULL default,
↪  leave NULL.

# === Atoms you must NOT assign — leave NULL ===

## Program-defined / aggregate atoms

The SMT program already computes these from constituent atoms via
↪  assertions like `(assert (= X (... constituents ...)))`. If you
↪  assign a value and any constituent contradicts it, the program
↪  becomes UNSAT — even if the constituents are correct. **Always
↪  leave defined atoms NULL** and let the program compute them.

How to recognize a defined / aggregate atom:
- Name suggests an aggregate: `_count`, `_total`, `_sum`, `_score`,
↪  `_criteria_met`, `_criteria_satisfied`, `_present`, `_passed`,
↪  `_completed`.
- Name describes a CRITERION SUMMARY rather than an atomic clinical
↪  fact — e.g., `complete_kawasaki_criteria_met`,
↪  `inclusion_criteria_satisfied`, `meets_acr_1987_criteria`,
↪  `coronary_abnormality_present` (aggregates several anatomical
↪  findings).
- Examples to leave NULL: `main_clinical_signs_count`,
↪  `complete_kawasaki_criteria_met`,
↪  `patient_meets_radiographic_criterion`, `symptoms_signs_count`,
↪  `criteria_passed`, `eligibility_met`.

## Atoms remediable by simple action at the visit

If the atom describes a fact that can trivially be obtained, satisfied,
↪  or established by a routine action at the in-person visit (signing
↪  a form, providing a sample, scheduling a procedure, attending a
↪  session, granting access), silence → NULL. Coordinator will
↪  resolve. Never FALSE on silence.

Examples:
- Patient willingness, consent, capability for future trial actions.
- Operational / demographic-context atoms: geographic eligibility,
↪  residence, citizenship/nationality
↪  (`patient_is_<country>_citizen`), occupation, employment status,
↪  primary language, care setting, communication tech (mobile phone,
↪  internet access), insurance, study-defined timing windows, ability
↪  to attend visits. **Never default these to TRUE or FALSE on chart
↪  silence — always NULL.** A typical patient vignette describes the
↪  chief complaint and clinical picture, not
↪  citizenship/occupation/residence; silence is non-informative.
- Patient-action / sample-provision / questionnaire-completion atoms
↪  (e.g., `patient_has_signed_consent`,
↪  `patient_has_provided_blood_sample`,
↪  `patient_completed_intake_questionnaire`).
- Procedures the patient is **scheduled to undergo as part of the trial
↪  protocol** (e.g., `patient_has_undergone_endoscopy_now` when the
↪  trial is about endoscopy findings). These are trial-driven, not
↪  patient history — distinguish from genuine prior-history atoms.

# === Other atom-class clarifications ===

- **Active-medication atoms** (`patient_is_exposed_to_<drug>_now`):
↪  silence → NULL (do not reject on missing mention; coordinator
↪  confirms at visit).
- **History-of-X atoms** (`patient_has_diagnosis_of_X_inthehistory`,
↪  `patient_has_taken_X_inthehistory`): silence → NULL.
- **History-of-X with active treatment / sequelae** → set
↪  `has_active_X_now` or `has_diagnosis_of_X_now` to TRUE via
↪  clinical inference. State continuity is implied by treatment
↪  continuity (e.g., "h/o stroke + on antithrombotics" →
↪  `has_acute_stroke_state = TRUE`; "h/o hypothyroidism + on
↪  levothyroxine" → `has_hypothyroidism_now = TRUE`).
- **Specific diagnostic-entity atoms** (`has_diagnosis_of_SLE_now`):
↪  require entity-specific chart wording. Generic compatible symptoms
↪  do not establish the named diagnosis. Default NULL.
- **Methodology / refinement qualifier atoms**
↪  (`@@histologically_confirmed`, `@@stage_iiib`,
↪  `@@therapeutic_dose`, `@@provoked_by_<X>`, `ECOG_0_or_1`): default
↪  NULL **only** when the chart is silent on the qualifier. If the
↪  chart supports the qualifier — verbatim or by strong clinical
↪  inference (e.g., "recent postpartum" →
↪  `@@provoked_by_reversible_risk_factor` for a thromboembolism) —
↪  set TRUE per Rule A. Refinement-qualifier-NULL is a SILENCE
↪  default, not a blanket override of chart evidence.
- **Threshold helper variables** (`__THRESH__::*`): silence → NULL
↪  unless the threshold concerns a value that is part of standard
↪  intake AND the chart documents that value.
- **Numeric atoms**: extract the documented value. If only a range is
↪  documented, set value=null and provide range. If silent, set
↪  value=null.

# === Output Format ===

Produce a JSON object whose keys are the variable names. For each
↪  variable, output:

{
  "<var_name>": {
    "assessment": "<one-sentence reasoning that ties this atom's value
    ↪  to chart text or to an explicit silence-default rule>",
    "value": true | false | null | <number>,
    "evidence": "<verbatim chart quote, or 'silent: defer to visit' for
    ↪  NULL on silence>",
    "range": {"min": <num>|null, "max": <num>|null}   (numeric
    ↪  variables only, when range is the best you can do)
  }
}

# === INPUTS ===

You are encoding variables for the **inclusion** side of this trial.

<trial_title>
{{TRIAL_TITLE}}
</trial_title>

<inclusion_criteria>
{{INCLUSION_CRITERIA}}
</inclusion_criteria>

<exclusion_criteria>
{{EXCLUSION_CRITERIA}}
</exclusion_criteria>

<variable_list>
{{VARIABLE_LIST}}
</variable_list>

<patient_notes>
{{PATIENT_NOTES}}
</patient_notes>
```

A.3.2 Exclusion-side value-miner prompt

```
# === ROLE ===
You are an expert clinician at PRESCREEN evaluating whether a patient
↪  triggers any **EXCLUSION** criterion of a clinical trial. The
↪  atomic variables you populate feed a deterministic Z3 solver that
↪  decides eligibility — assign each variable's value as a competent
↪  clinician would, using the same reasoning quality you would apply
↪  to a holistic eligibility judgment.

You are encoding ONLY the exclusion-side variables in this pass.

# === STAGE 1 — Read the chart and the trial holistically ===

Before populating any atom, internally:
1. Identify the trial's substantive intent (what kind of patient is
↪  this trial seeking, and what is being excluded?).
2. Extract every chart fact relevant to that intent and to the listed
↪  exclusion variables. Use chart text directly; use clinical
↪  inference where defensible (e.g., "20 days post-mastectomy" implies
↪  cancer history; "post-stroke on antithrombotics" implies recent
↪  stroke event).
3. Identify which exclusion criteria the chart explicitly addresses
↪  (positively or negatively) and which it is silent on.

Do NOT skip Stage 1. It anchors every atom value to a coherent reading
↪  of the chart.

# === STAGE 2 — Populate atom values ===

For each atomic variable, decide its value following these rules:

## Decision rule for each atom (apply IN ORDER)

**Rules A and B always take precedence over the silence defaults and
↪  atom-class defaults below.** If the chart speaks to the atom, do
↪  not default — extract.

A. **Chart explicitly supports the exclusion fact** (verbatim text OR
↪  strong clinical inference):
   → set to TRUE (this exclusion fires).
   This overrides the "leave NULL" categories below — even refinement
   ↪  qualifiers, even "remediable-by-action" atoms. If the chart
   ↪  provides positive evidence, assign TRUE.
```

```
   Example: atom is `patient_has_diagnosis_of_pulmonary_embolism_now@provoked_by_reversible_major_risk_factor`. Chart says "patient gave birth 3 weeks ago" → recent postpartum is a reversible risk factor → assign **TRUE**, not NULL.

B. **Chart explicitly contradicts the exclusion fact** (e.g., chart says "no history of malignancy" and atom is `patient_has_diagnosis_of_cancer_inthehistory`):
   → set to FALSE (this exclusion does not fire).

C. **Chart is silent or genuinely ambiguous on the exclusion fact**:
   → **set to NULL** (defer to in-person visit). NEVER assign FALSE on silence.

**Internal consistency**: your `assessment` field must agree with your `value` field. If your assessment says "assign TRUE", the value must be `true`. Never write a TRUE-justifying assessment alongside a NULL value.

## Why exclusion silence is always NULL

The exclusion's job is to fire only when the chart explicitly evidences the exclusion fact. A short prescreen vignette cannot be assumed to enumerate every possible exclusion — silence on an exclusion fact is non-informative, not evidence of absence. Setting silence → FALSE on the exclusion side spuriously asserts "the patient does NOT have X" and lets ineligible patients through. Coordinator review at the visit safely resolves NULL.

# === Atoms you must NOT assign — leave NULL ===

## Program-defined / aggregate atoms

The SMT program already computes these from constituent atoms via assertions like `(assert (= X (... constituents ...)))`. If you assign a value and any constituent contradicts it, the program becomes UNSAT — even if the constituents are correct. **Always leave defined atoms NULL** and let the program compute them.

How to recognize a defined / aggregate atom:
- Name suggests an aggregate: `_count`, `_total`, `_sum`, `_score`, `_criteria_met`, `_criteria_satisfied`, `_present`, `_passed`, `_completed`.
- Name describes a CRITERION SUMMARY rather than an atomic clinical fact — e.g., `meets_exclusion_criteria`, `any_active_malignancy`, `coronary_abnormality_present`.

## Atoms remediable by simple action at the visit

If the exclusion atom describes a fact that can trivially be obtained, satisfied, or resolved by a routine action at the in-person visit (signing a form, providing a sample, scheduling a procedure, attending a session, granting access), silence → NULL. Coordinator will resolve. Never FALSE on silence; never TRUE without explicit chart support.

Examples:
- Patient willingness, consent, capability — these never constitute exclusions on silence.
- Operational / demographic-context atoms: geographic eligibility, residence, citizenship/nationality (`patient_is_<country>_citizen`), occupation, employment status, primary language, care setting, communication tech, insurance, study-defined timing windows. **Never default these to TRUE or FALSE on chart silence — always NULL.** Silence on these is non-informative; coordinator confirms at visit.
- Procedures the patient is **scheduled to undergo as part of the trial protocol** (e.g., `patient_has_undergone_endoscopy_now` when the trial uses endoscopy as its measurement modality; `patient_has_undergone_ct_angiography_inthehistory` when the trial uses CTA). These are trial-driven actions, not patient history — distinguish from genuine prior-history atoms used as exclusions (e.g., `prior_chemotherapy_inthehistory` excluding chemo-naive trials).

# === Other atom-class clarifications ===

- **Active-medication atoms** (`patient_is_exposed_to_<drug>_now`): silence → NULL.
- **History-of-X atoms** (`patient_has_diagnosis_of_X_inthehistory`, `patient_has_taken_X_inthehistory`): silence → NULL. Fire TRUE only when chart explicitly affirms.
- **History-of-X with active treatment / sequelae** → set `has_active_X_now` or `has_diagnosis_of_X_now` to TRUE via clinical inference. State continuity is implied by treatment continuity (e.g., "h/o stroke + on antithrombotics" → `has_acute_stroke_state = TRUE`; "h/o hypothyroidism + on levothyroxine" → `has_hypothyroidism_now = TRUE`).
- **Specific diagnostic-entity atoms** (`has_diagnosis_of_SLE_now`): require entity-specific chart wording. Generic compatible symptoms do not establish the named diagnosis. Default NULL.
- **Methodology / refinement qualifier atoms** (`@@histologically_confirmed`, `@@stage_iiib`, `@@therapeutic_dose`, `@@provoked_by_<X>`, `ECOG_0_or_1`): default NULL **only** when the chart is silent. If the chart supports the qualifier verbatim or via strong clinical inference (e.g., "recent postpartum" → `@@provoked_by_reversible_risk_factor`), set TRUE per Rule A. Refinement-qualifier-NULL is a SILENCE default, not a blanket override of chart evidence.
- **Threshold helper variables** (`__THRESH__::*`): silence → NULL unless the threshold concerns a value that is part of standard intake AND the chart documents that value.
- **Numeric atoms**: extract the documented value. If only a range is documented, set value=null and provide range. If silent, set value=null.

# === Output Format ===

Produce a JSON object whose keys are the variable names. For each variable, output:

{
  "<var_name>": {
    "assessment": "<one-sentence reasoning that ties this atom's value to chart text or to an explicit silence-default rule>",
    "value": true | false | null | <number>,
    "evidence": "<verbatim chart quote, or 'silent: defer to visit' for NULL on silence>",
    "range": {"min": <num>|null, "max": <num>|null}  (numeric variables only, when range is the best you can do)
  }
}

# === INPUTS ===

You are encoding variables for the **exclusion** side of this trial.

<trial_title>
{{TRIAL_TITLE}}
</trial_title>

<inclusion_criteria>
{{INCLUSION_CRITERIA}}
</inclusion_criteria>

<exclusion_criteria>
{{EXCLUSION_CRITERIA}}
</exclusion_criteria>

<variable_list>
{{VARIABLE_LIST}}
</variable_list>

<patient_notes>
{{PATIENT_NOTES}}
</patient_notes>
```

A.4 Solver and Resolution Module SOLVE

The solver evaluates the combined patient–trial constraints $C_{t,p} = C_t \wedge C_p$ after a deterministic alias-remapping pass. It uses Z3 [3] to determine satisfiability and the weighted optimization interface of Z3/MaxSAT [2] to compute the minimum-cost flip set $\Delta_{t,p}$ when the patient is ineligible. The solver returns

$$\text{SOLVE}(C_{t,p}) = \{v_{t,p}, \tau_{t,p}, \rho_{t,p}, \Delta_{t,p}, L_{t,p}\},$$

where $v_{t,p}$ is the verdict, $\tau_{t,p}$ is the trace supporting the verdict, $\rho_{t,p}$ is the set of residual constraints or assumptions, $\Delta_{t,p}$ is the minimum set of patient facts that would need to change to make the patient eligible, and $L_{t,p}$ labels each criterion as SATISFIED, VIOLATED, or DEFERRED.

Resolution module. The SMT program also contains auxiliary definitional constraints linking related predicates, such as continuity between current and historical states, qualifier predicates implying their stems, and aggregate counts defined by their constituents. When value-miner outputs violate one of these auxiliary constraints, the resulting inconsistency reflects an extraction conflict rather than a genuine eligibility conflict. In such cases, the arbiter re-reads the chart against the conflicting atoms and relevant auxiliary constraints, then outputs the minimal value changes needed to restore consistency while remaining faithful to the chart. If the arbiter cannot determine a faithful correction, it returns no override and the solver result is kept.

A.4.1 Arbiter prompt

```
You are an expert clinician auditing an LLM's atomic extraction of patient facts.

The SMT solver reports the following atoms form a CONTRADICTION (UNSAT core).
The contradiction is internal to the LLM's extractions — the program has auxiliary/definitional constraints that link these atoms (e.g., state continuity
between `X_now` and `X_inthehistory`; qualifier-implies-stem; aggregate counts
defined by constituents).

Your job: review each LLM-assigned atom in the core against the chart, and
output the minimal set of value changes that resolves the contradiction while
faithfully reflecting the chart.

PATIENT CHART:
{chart}

CONTRADICTING ATOMS (LLM-assigned values + evidence):
{atoms_listing}

RELEVANT PROGRAM CONSTRAINTS (auxiliary / definitional assertions in the unsat core):
{program_assertions}

Output STRICT JSON, no commentary:
{{
  "overrides": [
    {{"atom": "<atom_name>", "new_value": true | false | null,
      "rationale": "<one sentence: why the chart supports this value>"}},
    ...
  ]
}}

Rules:
- Prefer NULL when the chart truly doesn't speak to the atom (defer to visit).
- Prefer flipping a value (true↔false) only when the chart provides clear
  evidence. State continuity: if a finding is `X_now=true` and the chart
  describes a recent/acute presentation, `X_inthehistory` is likely also true
  (recent onset is part of history). Qualifier-stem: a positive stem with a
  silent qualifier should leave the qualifier NULL, not FALSE.
- Output the MINIMAL set of overrides needed. Do not list atoms whose LLM value
  is correct.
- If you cannot determine the right resolution from the chart, output
  `"overrides": []` and the solver will return UNSAT (patient ineligible).
```

A.5 Verbalizer VERBALIZE

The verbalizer renders the structured solver output as the natural-language rationale shown to clinicians:

$$\text{VERBALIZE}(\{v, \tau, \rho, \Delta, L\}) = \{v, r\}.$$

It does not make a new matching decision. Instead, it describes the solver-produced verdict, supporting patient facts, criterion statuses, residual constraints or assumptions for ELIGIBLE verdicts, and the flip set for INELIGIBLE verdicts. Thus, the solver output remains the source of truth; the rationale is only a readable rendering.

A.5.1 Verbalizer prompt

```
# === ROLE ===
You are a careful clinical-trial eligibility explainer.

# === GOAL ===
You are given:
- a patient note
- a trial's eligibility text
- decision artifacts produced by AEGIS, a symbolic eligibility-matching system
- extracted patient-fact values used by AEGIS (with chart evidence)
- (optionally) ARBITER overrides — atom values that the symbolic solver flagged
  as internally contradictory and a downstream arbiter re-resolved against the chart

Your task: explain, in plain English, why AEGIS reached the provided eligibility
label. Cite chart evidence directly when relevant.

Length: 6-12 sentences. Concision is good but is NOT a virtue if it hides
the per-criterion bookkeeping — see the explicitness rule below.
This is ONLY for evaluation. Do NOT suggest fixes. Do NOT propose code changes.
Do NOT reveal or speculate about the system's implementation beyond what's
described here.

# === SYSTEM PIPELINE (FOR YOUR REFERENCE) ===

AEGIS extracts atomic eligibility facts from the chart, encodes them and the
trial criteria into an SMT (satisfiability-modulo-theories) program, and asks
Z3 whether the patient's fact pattern is consistent with the trial's
inclusion-side requirements (and likewise the exclusion side).

When an LLM extraction step produces atom values that internally contradict
each other (e.g., "current acute coronary syndrome = TRUE" but
"history of acute coronary syndrome = FALSE" — incompatible because recent
onset is part of history), the SMT solver flags this as UNSAT. An ARBITER
stage then re-reads the chart and corrects the offending extraction. Any such
correction is shown to you in the artifacts.

# === IMPORTANT ===
- Use ONLY evidence available in the provided inputs.
- Do NOT assume missing facts.
- The artifacts may include `inclusion_unsat_core` / `exclusion_unsat_core` —
  atoms (after any arbiter resolution) whose values made one side inconsistent. If a side's status is "UNSAT", these atoms identify the source
  of ineligibility.
- If `arbiter_overrides` is non-empty, mention briefly that the arbiter resolved
  an internal contradiction and what the resolved value is. Don't speculate
  about the arbiter beyond what's recorded.

# === EXPLICITNESS RULE (READ CAREFULLY) ===

The rationale MUST enumerate every distinct trial criterion in plain
clinical language. Do not summarise with generic phrases like "the
patient meets all inclusion criteria" or "no exclusions fired" — a
clinician reviewing the rationale should be able to see, for every
criterion, exactly what was verified vs. what was assumed pending the
in-person visit.

For each criterion, write a short clinical-prose clause. Use natural
language; AVOID jargon like "prescreen-default", "soft-met",
"SAT", or atom names. Instead:

- **Criterion confirmed by chart:** *"Age 18 or older — the chart
  documents a 47-year-old."*  *"Documented coronary catheterization
  via the right femoral artery satisfies the procedure requirement."*

- **Criterion the chart does not address:** *"Antibody panel is not
  in this note; this would be checked at the in-person visit."*
  *"Pregnancy status is not documented; we'd verify at the visit."*
  *"No information on TSH; deferred to screening."*

- **Exclusion the chart does not address:** *"There is no mention of
  prior chemotherapy in the chart."*  *"No documented contraindication
  to study medication."*  *"No history of the listed cardiac
  conditions in this note."*

- **Inclusion criterion that fails (UNSAT case):** name the
  criterion, cite the chart fact that contradicts it. *"Trial
  requires acute pulmonary embolism; the chart describes only chronic
  shortness of breath without imaging-confirmed PE."*

- **Exclusion criterion that fires (UNSAT case):** name the
  exclusion, cite the chart fact. *"The chart documents a pulsatile
  groin mass with bruit, consistent with a femoral pseudoaneurysm —
  this triggers the exclusion for vascular access-site complications."*

For other (non-blocking) criteria, group them in a closing clause:
*"No documented contraindications among the other listed
exclusions."*

NEVER write "all inclusion criteria are satisfied based on available
chart evidence" as a standalone sentence — that conflates verified
and deferred criteria. Always distinguish chart-confirmed from
not-in-chart. Never use the phrase "prescreen-default" in the
output; render it in clinical prose as "deferred to the visit",
"would be checked at screening", or simply "not in this note".

The downstream reader is a clinician deciding whether to forward the
patient, not an SMT engineer.

# === FORMAT ===

Output STRICT JSON, no commentary outside the JSON:
{
  "label": "eligible" | "ineligible",
  "rationale": "<plain-English explanation, 4-8 sentences, cite chart evidence>"
}

# === INPUTS ===

PATIENT NOTE:
{{NOTE}}

TRIAL ELIGIBILITY TEXT:
{{TRIAL}}

AEGIS DECISION ARTIFACTS:
{{ARTIFACTS}}

INCLUSION-SIDE PATIENT FACT VALUES (atom = value, with chart evidence):
{{INC_PV}}

EXCLUSION-SIDE PATIENT FACT VALUES (atom = value, with chart evidence):
{{EXC_PV}}

ARBITER OVERRIDES (atom-value resolutions made after detecting internal contradiction):
{{ARBITER_OVERRIDES}}
```

B Baseline implementations

This appendix specifies the four LLM-based matching baselines used in §5.1: LLMMATCH, CoT LLM, TrialGPT-Matching, and ZSPM. For each baseline, we describe the structured output it produces, how we aggregate that output into a binary patient-level verdict, and the prompt used to produce the reported results. All baselines use the same patient records, trial criteria, and backbone LLMs as SMTMATCH.

For baselines with multiple prompting or aggregation variants, we evaluate all variants on a held-out development split and report only the best-performing variant in the main results.

B.1 LLMMATCH

Pipeline. LLMMATCH is a single-prompt LLM matcher. The model receives the patient chart and the full trial listing, including both inclusion and exclusion criteria with their original headers preserved. It is instructed to reason through the patient's clinical state and the trial criteria, then return a binary patient-level verdict and a short free-text rationale as a JSON object.

Variants tried. We evaluated four prompting variants on the development split: *think-then-aggregate*, *single-step*, *aggregation-first*, and *blocker-then-verdict*. The *think-then-aggregate* variant performed best and is the one reported in the main results. In this variant, the model first reasons through the patient's clinical state and trial requirements before committing to a verdict.

B.1.1 LlmMatch prompt

```
You are an expert clinician at PRESCREEN. Reason in two stages.

STAGE 1 — Extract relevant chart facts.
List the patient facts most relevant to this trial's criteria. Use only
↪  what
is documented.

STAGE 2 — Judge eligibility.
Apply the trial's criteria to the extracted facts. Default to FORWARD on
chart silence (prescreen-doctrine).

CLINICAL TRIAL:
{trial}

PATIENT VIGNETTE:
{note}

After your two-stage reasoning, output the final JSON:
{"eligibility": "eligible"|"ineligible", "explanation": "<see rules
↪  below>"}

# === EXPLANATION REQUIREMENTS (CRITICAL) ===

The `explanation` field MUST surface per-criterion bookkeeping. Do not
summarise with phrases like "the patient meets all inclusion criteria"
without grounding each part. Specifically:

- **For criteria with direct chart evidence:** name the criterion and
↪  cite
  the chart fact (e.g., "Age >= 18: met -- chart documents age 47").
- **For criteria where the chart is silent:** explicitly say so and
↪  label
  the criterion as "unknown / not addressed in chart". Do not gloss over
  unknowns.
- **For ineligible verdicts:** name the specific blocker criterion(s)
↪  and
  cite the chart fact that triggers it.
- **For eligible verdicts:** state which inclusion criteria are
↪  confirmed
  by chart evidence, which are unknown / silent (prescreen-defaulted),
  and that no exclusion criterion fires.
Length: 6--12 sentences. Concision is good but is NOT a virtue if it
↪  hides
the per-criterion bookkeeping above.
```

B.2 CoT LLM

Pipeline. CoT LLM uses the same inputs as LLMMATCH but requires an explicit criterion-by-criterion analysis before producing the final verdict. For each inclusion criterion, the model emits a status from {*included*, *not included*, *unknown*} and a short reason. For each exclusion criterion, it emits a status from {*excluded*, *not excluded*, *unknown*} and a short reason. The model then aggregates these per-criterion judgments into a binary patient-level verdict and an aggregation rationale.

Variants tried. We evaluated three CoT variants on the development split: (i) `V5_COT`, which uses the shared label set {*met*, *not met*, *unknown*} for both inclusion and exclusion criteria; (ii) `V5_VERBOSE_AGG`, which adds an explicit aggregation step but retains the shared *met*/*not met* labels; and (iii) `V5_VERBOSE_AGG_V2`, which uses side-specific labels: *included*/*not included* for inclusion criteria and *excluded*/*not excluded* for exclusion criteria. `V5_VERBOSE_AGG_V2` performed best and is the variant reported in the main results.

B.2.1 CoT LLM prompt

```
# === ROLE ===
You are an expert clinician determining patient eligibility for a
↪  clinical trial via explicit per-criterion analysis followed by
↪  aggregation. You will use side-specific, unambiguous labels — never
↪  the word "met."

# === INPUTS ===

PATIENT CHART:
{{CHART}}

TRIAL ELIGIBILITY:
{{TRIAL}}

# === TASK ===

You will work in TWO STAGES.

**Stage 1 — Per-criterion judgment.** For EACH criterion, output:
- `text`: a verbatim or near-verbatim restatement of the criterion
- `status`: chosen from a SIDE-SPECIFIC enum:
  - For inclusion criteria: `included` (patient satisfies the
  ↪  inclusion) | `not_included` (patient does NOT satisfy the
  ↪  inclusion) | `unknown` (chart silent)
  - For exclusion criteria: `excluded` (the exclusion APPLIES to the
  ↪  patient — patient should be EXCLUDED from the trial) |
  ↪  `not_excluded` (the exclusion does NOT apply — patient is NOT
  ↪  excluded) | `unknown` (chart silent)
- `reasoning`: 1-2 sentences citing chart evidence

The labels are deliberately side-specific so that the polarity is
↪  unambiguous: `excluded` always means "this exclusion criterion
↪  fires and disqualifies the patient." Never use `met` / `not_met`.

**Stage 2 — Explicit aggregation.** Apply the standard logical rule:
- Patient is **eligible** iff EVERY inclusion criterion is `included`
↪  AND every exclusion criterion is `not_excluded`.
- Patient is **ineligible** iff ANY inclusion criterion is
↪  `not_included` OR any exclusion criterion is `excluded`.
- For `unknown` rows: you may apply the prescreen-doctrine — treat
↪  unknown inclusion as soft-included (defer to visit) ONLY IF the
↪  unknown is on a non-population-defining minor criterion; treat
↪  unknown exclusion as soft-not_excluded under the same condition.

Output your aggregation reasoning step-by-step, then the final verdict.

# === OUTPUT (STRICT JSON, no commentary outside the object) ===
{
  "inclusion_criteria": [
    {"text": "<verbatim criterion>", "status":
    ↪  "included"|"not_included"|"unknown", "reasoning": "<1-2
    ↪  sentences>"},
    ...
  ],
```

```
  "exclusion_criteria": [
    {"text": "<verbatim criterion>", "status":
   ↪  "excluded"|"not_excluded"|"unknown", "reasoning": "<1-2
   ↪  sentences>"},
    ...
  ],
  "aggregation_reasoning": "<step-by-step explanation of how
 ↪  per-criterion statuses combine: e.g., 'Inclusion 1: included.
 ↪  Inclusion 2: included. Inclusion 3: unknown — applied
 ↪  prescreen-doctrine, treated as included. Exclusion 1:
 ↪  not_excluded. Exclusion 2: not_excluded. Therefore: ELIGIBLE.'>",
  "eligibility": "eligible" | "ineligible"
}

# === RULES ===
- The aggregation reasoning MUST be derivable from the per-criterion
↪  list — no introducing new facts.
- The eligibility verdict MUST be consistent with the aggregation
↪  reasoning.
- Be exhaustive on per-criterion judgments — list every distinct
↪  criterion the trial states.
- NEVER use the labels `met` or `not_met`. Use only the side-specific
↪  labels above.
```

B.3 TrialGPT-Matching

Pipeline. TrialGPT-Matching [9] classifies eligibility at the criterion level using separate prompts for inclusion and exclusion criteria. Each criterion is assigned one of five labels: {*included*, *not included*, *not applicable*, *excluded*, *not excluded*}, together with a short per-criterion rationale. The original TrialGPT matching component does not directly output a binary patient-level verdict.

Aggregation. To compare TrialGPT-Matching with the other matchers, we add a deterministic binary aggregation rule on top of its per-criterion outputs. We evaluated four aggregation rules on the development split: (i) all-inclusions-met, which marks a pair eligible iff every inclusion criterion is labeled *included*; (ii) majority-of-criteria; (iii) the published TrialGPT scoring heuristic with a learned cutoff; and (iv) a permissive rule that marks a pair eligible iff no inclusion criterion is labeled *not included* and no exclusion criterion is labeled *excluded*. Rule (iv) performed best and is the aggregation rule reported in the main results.

B.3.1 TrialGPT-Matching inclusion prompt

```
You are a helpful assistant for clinical trial recruitment. Your task
↪  is to compare a given patient note and the inclusion criteria of a
↪  clinical trial to determine the patient's eligibility at the
↪  criterion level.
The factors that allow someone to participate in a clinical study are
↪  called inclusion criteria.
You should check the inclusion criteria one-by-one, and output the
↪  following three elements for each criterion:
        Element 1: brief reasoning.
        Element 2: list of relevant sentence IDs (or empty list).
        Element 3: label from {"not applicable", "not enough
       ↪  information", "included", "not included"}.
Output ONLY a JSON dict: {criterion_idx: [reasoning, [sent_ids],
↪  label]}.
```

B.3.2 TrialGPT-Matching exclusion prompt

```
You are a helpful assistant for clinical trial recruitment. Your task
↪  is to compare a given patient note and the exclusion criteria of a
↪  clinical trial to determine the patient's eligibility at the
↪  criterion level.
The factors that disqualify someone from participating are called
↪  exclusion criteria.
You should check the exclusion criteria one-by-one, and output the
↪  following three elements for each criterion:
        Element 1: brief reasoning.
        Element 2: list of relevant sentence IDs (or empty list).
        Element 3: label from {"not applicable", "not enough
       ↪  information", "excluded", "not excluded"}.
Output ONLY a JSON dict: {criterion_idx: [reasoning, [sent_ids],
↪  label]}.
```

B.4 ZSPM

Pipeline. ZSPM [20] is the zero-shot patient matching baseline of Wornow et al. The model receives the patient chart and the trial's inclusion and exclusion criteria, parsed line by line. For each criterion, it emits an `is_met` judgment, a confidence label (*high*, *medium*, or *low*), and a short rationale. It then aggregates the criterion-level judgments into a patient-level verdict.

We use two prompt variants of ZSPM, one for the main matching evaluation and one for the counterfactual self-faithfulness analysis. The variants share the same chart and criterion inputs but differ in the verdict and evidence-status schema they ask the model to produce.

Matching-evaluation variant. For the main matching metrics in §5, we use the original ternary-verdict prompt from Wornow et al., which emits `global_decision` $\in \{0, 1, 2\}$: 0 = *not eligible*, 1 = *might be eligible*, and 2 = *eligible*. Because our reference evaluation is binary, we evaluated both possible mappings of the middle bucket on the development split: (i) map *might be eligible* to ELIGIBLE (`global_decision` ≥ 1), and (ii) map it to INELIGIBLE (`global_decision` $= 2$ only). Mapping the middle bucket to ELIGIBLE performed better and is the mapping reported in the main results.

B.4.1 ZSPM matching-evaluation prompt

```
# Task
Your job is to indicate which of the following inclusion and exclusion
↪  criteria are met by the patient.

For an **inclusion** criteria to be "met", the patient must have the
↪  condition described in the criteria.
For an **exclusion** criteria to be "met", the patient must NOT have
↪  the condition described in the criteria.

# Patient

Below is a clinical note describing the patient's current health
↪  status:

```
{note}
```

Inclusion Criteria

The inclusion criteria being assessed are listed below:
{inc_str}

The exclusion criteria being assessed are listed below:
{exc_str}

Assessment
```

```
For each of the criteria above, use the patient's clinical note to
↪  determine whether the patient meets it. Think step by step.

Format your response as a JSON object with:
- "assessments": a list of dicts, each with {"criterion": str,
↪  "rationale": str, "is_met": bool, "confidence": "low|medium|high"}
- "global_decision": int in {0, 1, 2} where 0=definitely ineligible,
↪  1=might be eligible, 2=likely eligible. Be LENIENT: if patient
↪  meets most criteria but some are uncertain, default to 2.

Provide ONLY the JSON object.
```

Counterfactual-analysis variant. For the counterfactual self-faithfulness analysis in §5.5, the ternary verdict and binary `is_met` flag are insufficient: we need to distinguish criteria contradicted by chart evidence from criteria that are unmet only because the chart is silent. We therefore rerun ZSPM with a modified prompt that forces a binary patient-level decision (`eligible: bool`) and assigns each criterion an `evidence_status` from {`deterministic_met`, `deterministic_not_met`, `defer`}. The first two labels mark criteria whose status is supported by explicit chart evidence; `defer` marks criteria for which the chart does not provide enough evidence to determine the criterion. This distinction is needed for the minimum-flip-set analysis, because deferred criteria should not be counted as evidence-grounded blockers. We rerun the matcher rather than post-hoc relabeling the matching-evaluation output, because the original prompt does not ask the model to expose this distinction.

B.4.2 ZSPM counterfactual-analysis prompt

```
# Task
Your job is to indicate which of the following inclusion and exclusion
↪  criteria are met by the patient, then make a binary eligibility
↪  decision.

For an **inclusion** criteria to be "met", the patient must have the
↪  condition described in the criteria.
For an **exclusion** criteria to be "met", the patient must NOT have
↪  the condition described in the criteria.

# Patient

Below is a clinical note describing the patient's current health
↪  status:

```
{note}
```

Inclusion Criteria

The inclusion criteria being assessed are listed below:
{inc_str}

The exclusion criteria being assessed are listed below:
{exc_str}

Assessment

For each of the criteria above, use the patient's clinical note to
↪ determine whether the patient meets it. Think step by step.

Format your response as a JSON object with:
- "assessments": a list of dicts, each with {"criterion": str,
↪ "rationale": str, "is_met": bool, "confidence": "low|medium|high",
↪ "evidence_status": "deterministic_met" | "deterministic_not_met" |
↪ "chart_silent_defaulted_not_met"}
- "eligible": bool — true if the patient is eligible for this trial,
↪ false if not. Make a definitive binary decision; do NOT use a
↪ "maybe" or "uncertain" middle option.

The `evidence_status` field distinguishes:
  - "deterministic_met": chart explicitly supports the criterion
  ↪ (inclusion: present; exclusion: absent)
  - "deterministic_not_met": chart explicitly contradicts the criterion
  ↪ (inclusion: explicitly absent; exclusion: explicitly present,
  ↪ exclusion fires)
  - "chart_silent_defaulted_not_met": chart does not address the
  ↪ criterion; matcher defaults to is_met=false but the criterion
  ↪ could be verified at the in-person visit

Provide ONLY the JSON object.
```

C Prompting Strategy Search for the Best-Prompted-LLM Baseline

The BEST PROMPTED LLM baseline reported in the main paper is the strongest configuration recovered by an explicit search over prompting strategies for a single-shot eligibility verdict. We did not pick a single "obvious" prompt and call it the baseline; instead, we wrote six variants that exercise three orthogonal design axes and selected the one with the highest dev-set macro-F1.

C.1 Search axes

The six prompts cover the following three axes, which together span the prompting choices most commonly described as "state of the art" in the clinical-trial matching literature:

- **Reasoning structure.** Single-pass verdict vs. think-then-aggregate (the model first reasons criterion-by-criterion, then emits a global verdict) vs. aggregate-first verbose chain-of-thought.
- **Vocabulary specificity.** Generic include/exclude labels vs. side-specific labels (INCLUDED/NOT-INCLUDED for inclusion criteria, EXCLUDED/NOT-EXCLUDED for exclusion criteria), and an explicit blocker channel.
- **Verbosity.** Minimal-rationale vs. verbose per-criterion rationales.

C.2 Selection protocol

Each variant was scored on the dev split using the same scoring harness as the main experiments (gpt-4.1 backbone, identical chart and trial text, identical post-processing). The search produces two separately reported baselines:

- BEST PROMPTED LLM — the strongest variant over the full six-variant grid. The winner is the think-then-aggregate prompt (§ C.3.1).
- COT LLM — the strongest variant restricted to chain-of-thought-shape prompts (single-pass CoT vs. aggregation-first verbose CoT, with generic vs. side-specific labels for the aggregation-first variant). The winner is the aggregation-first verbose CoT with side-specific labels (§ C.3.3); the generic-label version (§ C.3.4) and the single-pass CoT (§ C.3.5) both underperformed it on the dev split.

We report both baselines so that COT LLM reflects a fair chain-of-thought prompting effort rather than a single hand-written CoT prompt. Variants are listed below in roughly descending dev-set strength.

C.3 Variants searched

Two-pass prompt: the model first emits a structured per-criterion assessment (criterion span, side, met/unmet/uncertain, brief justification), then aggregates those into a single eligibility verdict. This is the prompt used as BEST PROMPTED LLM in all main-paper tables.

C.3.1 Think-then-aggregate (selected)

```
You are an expert clinician at PRESCREEN. Reason in two stages.

STAGE 1 — Extract relevant chart facts.
List the patient facts most relevant to this trial's criteria. Use only
↪  what
is documented.

STAGE 2 — Judge eligibility.
Apply the trial's criteria to the extracted facts. Default to FORWARD on
chart silence (prescreen-doctrine).

CLINICAL TRIAL:
{trial}

PATIENT VIGNETTE:
{note}

After your two-stage reasoning, output the final JSON:
{"eligibility": "eligible"|"ineligible", "explanation": "<see rules
↪  below>"}

# === EXPLANATION REQUIREMENTS (CRITICAL) ===

The `explanation` field MUST surface per-criterion bookkeeping. Do not
summarise with phrases like "the patient meets all inclusion criteria"
without grounding each part. Specifically:

- **For criteria with direct chart evidence:** name the criterion and
↪  cite
  the chart fact (e.g., "Age >= 18: met -- chart documents age 47").
- **For criteria where the chart is silent:** explicitly say so and
↪  label
  the criterion as "unknown / not addressed in chart". Do not gloss over
  unknowns.
- **For ineligible verdicts:** name the specific blocker criterion(s)
↪  and
  cite the chart fact that triggers it.
- **For eligible verdicts:** state which inclusion criteria are
↪  confirmed
  by chart evidence, which are unknown / silent (prescreen-defaulted),
  and that no exclusion criterion fires.

Length: 6--12 sentences. Concision is good but is NOT a virtue if it
↪  hides
the per-criterion bookkeeping above.
```

Variant of think-then-aggregate that adds an explicit *blocker* channel: the model is asked to first surface any criterion that, if unmet, would single-handedly make the patient ineligible, and to commit to a verdict consistent with that channel.

C.3.2 Blocker-then-verdict

```
# === ROLE ===
You are an expert clinician determining patient eligibility for a
↪  clinical trial.

# === INPUTS ===

PATIENT CHART:
{{CHART}}

TRIAL ELIGIBILITY:
{{TRIAL}}

# === TASK ===

Decide whether the patient is eligible. If ineligible, identify the
↪  SPECIFIC chart facts that block eligibility.

Output STRICT JSON (no commentary outside the JSON object):

{
  "eligibility": "eligible" | "ineligible",
  "explanation": "<1-3 sentence summary>",
  "blockers": [
    {
```

```
      "fact": "<verbatim chart fact OR clinically-inferred fact that is
     ↪  the reason this patient does NOT qualify>",
      "side": "inclusion" | "exclusion",
      "criterion": "<which trial criterion the fact is responding to>",
      "reasoning": "<why this fact blocks eligibility>"
    }
  ],
  "supports": [
    {
      "fact": "<chart fact that the patient currently SATISFIES toward
     ↪  this trial>",
      "side": "inclusion" | "exclusion-not-fired",
      "criterion": "<which trial criterion the fact is responding to>",
      "reasoning": "<why this fact supports the patient's case for the
     ↪  trial>"
    }
  ]
}

# === RULES ===
- For "eligible" verdicts, return `"blockers": []` and a non-empty
↪  `supports` listing every chart fact that establishes inclusion.
- For "ineligible" verdicts, list EVERY chart fact that directly causes
↪  ineligibility under `blockers` — do not list general/contextual
↪  facts that didn't drive your decision.
- For both eligible and ineligible verdicts, also populate `supports`
↪  with the inclusion-supporting facts the patient DOES meet (e.g.,
↪  "patient is 65 years old → meets age criterion"; "chart describes
↪  peripheral arterial disease → matches the trial's target
↪  population"). These are facts that should be PRESERVED in any chart
↪  rewrite — they're the patient's positive case.
- "fact" should be a concise statement (verbatim chart language is
↪  best); flipping or removing the fact in the chart should plausibly
↪  change the verdict (for blockers) or break inclusion (for
↪  supports).
- Distinguish inclusion-side blockers (chart fails to support an
↪  inclusion criterion) from exclusion-side blockers (chart triggers
↪  an exclusion criterion).
- Be exhaustive on both lists.
```

Aggregation-first variant: the model emits a verbose chain-of-thought over all criteria and then commits to a verdict, using side-specific labels (INCLUDED/NOT-INCLUDED vs. EXCLUDED/NOT-EXCLUDED) rather than generic include/exclude. This is also the variant used to source the side-by-side rationales for the CoT LLM system in the pairwise-preference table.

C.3.3 Aggregation-first verbose CoT (side-specific labels)

```
# === ROLE ===
You are an expert clinician determining patient eligibility for a
↪  clinical trial via explicit per-criterion analysis followed by
↪  aggregation. You will use side-specific, unambiguous labels — never
↪  the word "met."

# === INPUTS ===

PATIENT CHART:
{{CHART}}

TRIAL ELIGIBILITY:
{{TRIAL}}

# === TASK ===

You will work in TWO STAGES.

**Stage 1 — Per-criterion judgment.** For EACH criterion, output:
- `text`: a verbatim or near-verbatim restatement of the criterion
- `status`: chosen from a SIDE-SPECIFIC enum:
  - For inclusion criteria: `included` (patient satisfies the
  ↪  inclusion) | `not_included` (patient does NOT satisfy the
  ↪  inclusion) | `unknown` (chart silent)
  - For exclusion criteria: `excluded` (the exclusion APPLIES to the
  ↪  patient — patient should be EXCLUDED from the trial) |
  ↪  `not_excluded` (the exclusion does NOT apply — patient is NOT
  ↪  excluded) | `unknown` (chart silent)
- `reasoning`: 1-2 sentences citing chart evidence

The labels are deliberately side-specific so that the polarity is
↪  unambiguous: `excluded` always means "this exclusion criterion
↪  fires and disqualifies the patient." Never use `met` / `not_met`.

**Stage 2 — Explicit aggregation.** Apply the standard logical rule:
- Patient is **eligible** iff EVERY inclusion criterion is `included`
↪  AND every exclusion criterion is `not_excluded`.
- Patient is **ineligible** iff ANY inclusion criterion is
↪  `not_included` OR any exclusion criterion is `excluded`.
- For `unknown` rows: you may apply the prescreen-doctrine — treat
↪  unknown inclusion as soft-included (defer to visit) ONLY IF the
↪  unknown is on a non-population-defining minor criterion; treat
↪  unknown exclusion as soft-not_excluded under the same condition.

Output your aggregation reasoning step-by-step, then the final verdict.

# === OUTPUT (STRICT JSON, no commentary outside the object) ===
{
  "inclusion_criteria": [
    {"text": "<verbatim criterion>", "status":
    ↪  "included"|"not_included"|"unknown", "reasoning": "<1-2
    ↪  sentences>"},
    ...
  ],
  "exclusion_criteria": [
    {"text": "<verbatim criterion>", "status":
    ↪  "excluded"|"not_excluded"|"unknown", "reasoning": "<1-2
    ↪  sentences>"},
    ...
  ],
  "aggregation_reasoning": "<step-by-step explanation of how
  ↪  per-criterion statuses combine: e.g., 'Inclusion 1: included.
  ↪  Inclusion 2: included. Inclusion 3: unknown — applied
  ↪  prescreen-doctrine, treated as included. Exclusion 1:
  ↪  not_excluded. Exclusion 2: not_excluded. Therefore: ELIGIBLE.'>",
  "eligibility": "eligible" | "ineligible"
}

# === RULES ===
- The aggregation reasoning MUST be derivable from the per-criterion
↪  list — no introducing new facts.
- The eligibility verdict MUST be consistent with the aggregation
↪  reasoning.
- Be exhaustive on per-criterion judgments — list every distinct
↪  criterion the trial states.
- NEVER use the labels `met` or `not_met`. Use only the side-specific
↪  labels above.
```

Earlier version of the aggregation-first verbose CoT prompt using generic include/exclude vocabulary. Underperformed the side-specific variant on the dev split.

C.3.4 Aggregation-first verbose CoT (generic labels)

```
# === ROLE ===
You are an expert clinician determining patient eligibility for a
↪  clinical trial via explicit per-criterion analysis followed by
↪  aggregation.

# === INPUTS ===

PATIENT CHART:
{{CHART}}

TRIAL ELIGIBILITY:
{{TRIAL}}

# === TASK ===

You will work in TWO STAGES.

**Stage 1 — Per-criterion judgment.** For EACH inclusion and exclusion
↪  criterion the trial states (one row per criterion), output:
- `text`: a verbatim or near-verbatim restatement of the criterion
- `status`: one of:
  - `met` (inclusion satisfied / exclusion does NOT fire)
  - `not_met` (inclusion fails / exclusion fires)
  - `unknown` (chart silent on this criterion)
- `reasoning`: 1-2 sentences citing chart evidence

**Stage 2 — Explicit aggregation.** Apply the standard logical rule:
- Patient is **eligible** iff EVERY inclusion criterion is `met` AND
↪  every exclusion criterion is NOT `not_met`.
- Patient is **ineligible** iff ANY inclusion criterion is `not_met` OR
↪  any exclusion criterion is `not_met`.
- For `unknown` rows: you may apply the prescreen-doctrine — treat
↪  unknown inclusion as soft-met (defer to visit) ONLY IF the unknown
↪  is on a non-population-defining minor criterion.

Output your aggregation reasoning step-by-step, then the final verdict.

# === OUTPUT (STRICT JSON, no commentary outside the object) ===
{
  "inclusion_criteria": [
    {"text": "<verbatim criterion>", "status":
    ↪  "met"|"not_met"|"unknown", "reasoning": "<1-2 sentences>"},
    ...
  ],
  "exclusion_criteria": [
    {"text": "<verbatim criterion>", "status":
    ↪  "met"|"not_met"|"unknown", "reasoning": "<1-2 sentences>"},
    ...
  ],
  "aggregation_reasoning": "<step-by-step explanation of how
  ↪  per-criterion statuses combine: e.g., 'Inclusion 1: met.
  ↪  Inclusion 2: met. Inclusion 3: unknown — applied
  ↪  prescreen-doctrine, treated as met. Exclusion 1: not fired.
  ↪  Exclusion 2: not fired. Therefore: ELIGIBLE.'>",
  "eligibility": "eligible" | "ineligible"
}

# === RULES ===
- The aggregation reasoning MUST be derivable from the per-criterion
↪  list — no introducing new facts.
- The eligibility verdict MUST be consistent with the aggregation
↪  reasoning.
- Be exhaustive on per-criterion judgments — list every distinct
↪  criterion the trial states.
```

Single-pass CoT: the model is asked to think step-by-step about the chart and the criteria in a single message and emit a verdict at the end, without the explicit per-criterion table required by the think-then-aggregate variant.

C.3.5 Chain-of-thought, single-pass verdict

```
You are an expert clinician determining patient eligibility for a
↪  clinical trial.

# Patient chart
{chart}

# Trial criteria
{trial}

# Instructions

First, reason step by step. For each inclusion criterion, decide
↪  whether the chart establishes it. For each exclusion criterion,
↪  decide whether the chart triggers it. Pay special attention to:
- which chart facts are decisive blockers (would, if changed, flip the
↪  verdict)
- which chart facts the chart is silent on (deferred to in-person visit)
- which inclusion-supporting facts the patient currently meets (must be
↪  preserved)

Output strict JSON only:

{
  "reasoning": "<your step-by-step reasoning, 4-8 sentences>",
  "decisive_blockers": [
    {"side": "inclusion|exclusion", "criterion": "<criterion text>",
    ↪  "chart_fact": "<chart fact making this a blocker>",
    ↪  "would_flip_if_changed": true}
  ],
  "inclusion_supports": [
    {"criterion": "<criterion text>", "chart_fact": "<chart fact
    ↪  establishing this inclusion>"}
  ],
  "deferred_at_visit": [
    {"side": "inclusion|exclusion", "criterion": "<criterion text>"}
  ],
  "eligibility": "eligible" | "ineligible",
  "explanation": "<1-2 sentence summary of why>"
}
```

Minimal baseline: chart plus criteria, verdict only, no per-criterion reasoning and no enforced think-step. Included to confirm that the prompting-strategy gains we report are not driven entirely by the model's default behavior.

C.3.6 Plain single-step verdict

```
You are a clinical trial matcher. Given a patient chart and a clinical
trial's eligibility criteria, decide whether the patient is eligible.

CLINICAL TRIAL:
{trial}

PATIENT CHART:
{note}

Output a single JSON object:
{"eligibility": "eligible"|"ineligible", "explanation": "<one short
↪  paragraph explaining your decision>"}
```

C.4 Takeaway

Across the six variants, the think-then-aggregate prompt was the strongest on the dev split, and we report it as BEST PROMPTED LLM in the main paper; the strongest of the three chain-of-thought-shape prompts was the aggregation-first verbose CoT with side-specific labels, which we report as COT LLM. This framing matters because a casual "we wrote a single prompt and called it the LLM baseline" — or a single hand-written CoT prompt for COT LLM — would understate how much of the headroom between off-the-shelf LLM matching and our system can be closed by prompt engineering alone, and how much remains after that headroom is taken.

D Counterfactual Self-Faithfulness: Pipeline Details

This appendix specifies the counterfactual self-faithfulness pipeline of §5.5: how flip targets are extracted from each system, how counterfactual charts are constructed and validated, the three modifier–validator configurations we report, how confidence intervals are computed, and how clinician audit data is used to produce the clinician-adjusted variant of the metric. We embed all CF-side prompts verbatim in §D.6.

D.1 Per-pair workflow

For every patient–trial pair (p, t) on which a matcher emits the verdict INELIGIBLE, the pipeline runs the following five steps.

1. **Extract flip targets.** For each matcher, the rejection reasons are extracted in the form most natural to the matcher (§D.2). The result is a list of *flip targets* — either free-text criterion-and-fact pairs for LLM matchers, explicitly listed unmet requirements for LLMMATCH-FlipExplicit, or symbolic atoms with target values for the SMT-based matcher.

2. **Construct a counterfactual chart.** A modifier LLM is given the original chart, the trial, and the list of flip targets, and is instructed to make the minimal edits that address those targets while preserving everything else. The modifier is told explicitly not to introduce new conditions, treatments, or chart facts beyond what is needed to remove the cited rejection reasons.

3. **Validate the counterfactual.** A separate validator LLM checks that the modified chart (i) addresses the targeted blockers, (ii) does not alter unrelated eligibility-relevant facts, (iii) does not introduce new blockers, and (iv) remains clinically coherent. Pairs whose modification fails any of these checks are excluded from the denominator of $\mathrm{SF}_{\mathrm{raw}}$.

4. **Re-run the matcher.** The same matcher, with the same backbone and prompt, is run on the validated counterfactual chart. We record the new verdict $\hat{y}'(p)$.

5. **Score.** The pair contributes to $\mathrm{SF}_{\mathrm{raw}}$ as a positive if the new verdict is ELIGIBLE, and as a negative otherwise.

The modifier and validator are run *out of sample* with respect to the matcher under test: they never see the matcher's verdict on the original chart, so the modification is not biased toward "producing a flip."

D.2 Per-system flip-target extraction

The matchers in our evaluation expose their rejection reasons in different forms, so the flip-target extraction step is tailored to each.

Single-shot, CoT, and TrialGPT LLM matchers. These matchers emit a natural-language rationale that names the criteria that drove the rejection. We parse cited flip targets from the rationale text with a side-LLM call using the prompt in §D.6.1. The output is a list of `{criterion, chart_fact}` pairs, where `criterion` quotes the trial-side requirement the matcher cited and `chart_fact` quotes the chart evidence the matcher used. These are the targets the modifier is asked to address.

LLMMATCH-FlipExplicit. LLMMATCH-FlipExplicit uses the same LLMMATCH matcher, but asks the model to explicitly list the unmet requirements whose change would make the patient eligible. These listed flip targets are used directly for counterfactual construction, rather than being recovered from a post-hoc rationale parsing step.

ZSPM. ZSPM emits per-criterion `is_met/not_met` assessments rather than a free-text rationale. We extract a *minimum flip set* from these assessments with the prompt in §D.6.3: the smallest set of per-criterion verdicts that would need to flip for the global decision to change from INELIGIBLE to ELIGIBLE. This puts ZSPM's flip targets on the same footing as the other matchers for the modification step.

Ours, SMT-based matcher. Our matcher's rejection reasons are already symbolic: the MaxSAT minimum-cost flip set $\Delta_{p,t}$ (§4.3) names the atomic patient predicates whose values must change to make the program satisfiable. We extract these atoms with their target values using the prompt in §D.6.2 and pass them to the modifier as a list of `{atom, current_value, target_value}` tuples. Because $\Delta_{p,t}$ is minimal by MaxSAT, the modification target is unambiguous.

D.3 Modifier–validator configurations

We report results under three modifier–validator configurations that swap backbones to test the robustness of the metric:

Modifier	Validator	Purpose
gpt-4.1	gpt-4.1 strict	Baseline same-backbone setting
gpt-4.1	gpt-5 simclin	Tests sensitivity to the validator
gpt-5	gpt-5 simclin	Headline strongest modifier/validator setting

Table 6: Modifier–validator configurations for counterfactual self-faithfulness. The gpt-5 modifier / gpt-5 simclin validator configuration is the headline setting; the other two configurations test sensitivity to the modifier and validator choices.

The two validator prompts differ in framing. The gpt-4.1 *strict* validator (§D.6.4) is a checklist-style prompt that asks whether each modification leaves the chart clinically coherent. The gpt-5 *simclin* validator (§D.6.5) is a senior trial-coordinator persona that re-screens the modified chart from scratch. Comparing gpt-4.1 modifier / gpt-4.1 validator with gpt-4.1 modifier / gpt-5 validator isolates the effect of the validator; comparing gpt-4.1 modifier / gpt-5 validator with gpt-5 modifier / gpt-5 validator isolates the effect of the modifier.

D.4 Confidence intervals

Because $\mathrm{SF}_{\mathrm{raw}}$ is a binomial proportion over validator-accepted counterfactuals, we report Wilson 95% confidence intervals [19]:

$$\mathrm{CI}_{95} = \frac{\hat{p} + \frac{z^2}{2n}}{1 + \frac{z^2}{n}} \pm \frac{z}{1 + \frac{z^2}{n}} \sqrt{\frac{\hat{p}(1-\hat{p})}{n} + \frac{z^2}{4n^2}},$$

with $z = 1.96$. Wilson intervals are preferred over normal-approximation intervals at small n and near-boundary $\hat{p}$, both of which can occur for individual system/configuration pairs.

D.5 Clinician-adjusted self-faithfulness

The validator is itself an LLM and can make two kinds of error: it can accept counterfactuals that do not in fact address the cited flip targets, or it can reject counterfactuals that actually do. The first kind can inflate $\mathrm{SF}_{\mathrm{raw}}$ on systems whose cited reasons are easy to remove cosmetically; the second kind can deflate the denominator. To assess this effect, we run a stratified clinician audit on the gpt-5 modifier / gpt-5 validator configuration.

Audit design. For each system, we stratify the set of validator-accepted counterfactuals into two buckets: *flipped* (the matcher's re-run verdict became ELIGIBLE) and *not flipped* (the verdict remained INELIGIBLE). We sample $K = 7$ counterfactuals from each bucket and present each to a senior trial-coordinator clinician who marks the modification as clinically valid or not, blind to the matcher's identity and the bucket label. The audit produces, per system, two precision estimates: p_{flip} = clinician-valid rate on the flipped bucket, and p_{not} = clinician-valid rate on the not-flipped bucket.

Re-weighting. Let N_F and N_{NF} be the full-corpus counts of validator-accepted flipped and not-flipped counterfactuals. The clinician-adjusted SF re-weights the bucket-conditional clinician-valid rates back to the full corpus:

$$\mathrm{SF}_{\mathrm{adj}} = \frac{p_{\mathrm{flip}}\, N_F}{p_{\mathrm{flip}}\, N_F + p_{\mathrm{not}}\, N_{NF}}.$$

This is the corpus-level fraction of *clinician-valid* counterfactuals on which the matcher's verdict flipped. The 95% confidence interval is computed by stratified pair-level bootstrap (2,000 resamples) within each bucket, propagating uncertainty from both the bucket-conditional rates and the bucket sizes.

D.6 CF-pipeline prompts

D.6.1 Modifier prompt (LLM matchers)

```
# === ROLE ===
You are rewriting a clinical patient chart so that EVERY cited blocker
↪  is addressed. The system's rationale will be parsed atomically:
↪  each cited blocker is one fact you must flip OR one silence you
↪  must resolve.

# === TYPES OF BLOCKERS (you will encounter both) ===

**Type A — STATED FACT BLOCKER**: the rationale cites a positive
↪  finding that triggers an exclusion or fails an inclusion (e.g.,
↪  "the patient has acute limb ischemia"). To flip: REMOVE or NEGATE
↪  the supporting chart language, while keeping inclusion-supporting
↪  findings intact.

**Type B — SILENCE BLOCKER**: the rationale says "there is no
↪  information about X" or "the chart does not mention Y" — and that
↪  silence fails an inclusion (e.g., "no information about informed
↪  consent capability"). To flip: ADD a single, short, plausible
↪  positive statement to the chart asserting the required fact (e.g.,
↪  "Patient is alert and competent; provides written informed
↪  consent."). Use clinical-style prose; do NOT invent specific test
↪  values, just the qualitative fact.

# === TWO-STEP PROCESS (REQUIRED) ===

Step 1 — PLAN (inside `<plan>...</plan>` tags, kept private from final
↪  output):
  a) Enumerate the cited blockers from the rationale. For each,
  ↪  classify it as Type A (stated fact) or Type B (silence).
  b) For each Type A blocker, identify the EXACT chart sentence(s) that
  ↪  establish the fact. Plan minimal edits to flip ONLY that
  ↪  sentence.
  c) For each Type B blocker, draft a single short sentence to ADD to
  ↪  the chart that resolves the silence.
  d) List sentences from the original chart that you will reproduce
  ↪  VERBATIM (everything not in 1.b).
  e) Anti-overcorrection check: do any planned edits in 1.b also delete
  ↪  content that supports an inclusion criterion (per the SUPPORTS
  ↪  list)? If yes, rework.

Step 2 — WRITE (inside `<chart>...</chart>` tags):
- Reproduce all preserved sentences verbatim.
- Apply minimal edits planned for Type A blockers.
- Insert short positive sentences planned for Type B blockers.
- Output natural clinical prose; do not include any planning text.
```

```
# === HARD RULES ===
- Every cited blocker must be addressed (Type A flipped OR Type B
↪  resolved with an added sentence).
- Do NOT introduce unrelated diagnoses or findings.
- Do NOT remove sentences that support inclusion criteria (the SUPPORTS
↪  list).
- Length should resemble the original chart (±30%).

# === FEW-SHOT EXAMPLES ===

## EXAMPLE 1 — Mixed Type A + Type B

Trial: coronary angiography study; inclusion includes "able to provide
↪  informed consent" and "sheath size 5-8F".
Rationale blockers:
  - [met] Patient is 67yo with R femoral catheterization. [SUPPORTS —
  ↪  preserve]
  - [not met] No information about informed consent capability. [Type B
  ↪  silence]
  - [not met] No information about sheath size. [Type B silence]
Original chart: "67yo woman, post R femoral cath. Cool foot, pulsatile
↪  mass with bruit."

<plan>
a) Two blockers; both Type B (silence).
b) No Type A edits.
c) Add: "Patient is alert and competent; she provided written informed
↪  consent for the procedure." Add: "A 6F sheath was used."
d) Verbatim: "67yo woman, post R femoral cath. Cool foot, pulsatile
↪  mass with bruit."
e) No overcorrection risk; only additions.
</plan>
<chart>
67yo woman, post R femoral cath. Cool foot, pulsatile mass with bruit.
↪  Patient is alert and competent; she provided written informed
↪  consent for the procedure. A 6F sheath was used.
</chart>

## EXAMPLE 2 — Type A blocker (delete + keep adjacent)

Rationale blockers: [exclusion met] "patient has acute limb ischemia."
↪  Inclusion-supporting fact (preserve): chronic critical limb
↪  ischemia signs (cool foot, lost pulses).
Original chart: "Cool right foot, absent distal pulses, acute pain
↪  onset 4 hours ago."

<plan>
a) One blocker, Type A (acute onset).
b) Sentence: "acute pain onset 4 hours ago." Plan: change "acute pain
↪  onset 4 hours ago" → remove (acute timing is the blocker). KEEP
↪  "Cool right foot, absent distal pulses" (chronic ischemia signs —
↪  inclusion).
c) No Type B.
d) Verbatim: "Cool right foot, absent distal pulses,"
e) Anti-overcorrection: do NOT remove cool foot/lost pulses — those
↪  support inclusion.
</plan>
<chart>
Cool right foot, absent distal pulses. Symptoms have been progressive
↪  over months.
</chart>

# === INPUTS ===

ORIGINAL CHART:
{{CHART}}

TRIAL ELIGIBILITY TEXT:
{{TRIAL}}

SYSTEM'S RATIONALE (parse blockers atomically — classify each as Type A
↪  or Type B):
{{RATIONALE}}

SYSTEM'S SUPPORTS (chart facts that already meet inclusion — preserve
↪  verbatim):
{{SUPPORTS}}

# === OUTPUT ===
First `<plan>...</plan>`, then `<chart>...</chart>` (final rewrite).
```

D.6.2 Modifier prompt (Ours, atom targets)

```
# === ROLE ===
You are rewriting a clinical patient chart to flip specific atomic
↪  facts WITHOUT collateral damage. You are NOT allowed to introduce
↪  new content or delete content unrelated to the targets.

# === TWO-STEP PROCESS (REQUIRED) ===

Step 1 — PLAN (think before writing):
Inside `<plan>...</plan>` tags, do this analysis (kept private from the
↪  final output):
  a) For each TARGET atom, identify the EXACT sentence(s) in the
  ↪  original chart that establish its current value.
  b) For each PRESERVE atom, identify the EXACT sentence(s) in the
  ↪  original chart that establish it. List them verbatim.
  c) Plan the MINIMAL EDIT: which words/clauses in (a) you will modify;
  ↪  explicitly note which sentences in (b) you will leave untouched.
  d) Anti-overcorrection check: list any "adjacent vocabulary" overlap
  ↪  — words that appear in both a TARGET-establishing sentence and a
  ↪  PRESERVE-establishing sentence. State how you will modify the
  ↪  target without affecting the preserve.

Step 2 — WRITE:
Inside `<chart>...</chart>` tags, output the rewritten chart text.
- Preserve sentences NOT in step 1.a UNCHANGED, including their exact
↪  wording.
- Sentences in step 1.b MUST be reproduced verbatim.
- Apply only the minimal edits identified in step 1.c.
- Length and prose style should resemble the original.

# === HARD RULES ===
- MINIMAL EDIT: change only what flips a TARGET atom's polarity. Every
↪  other sentence stays.
- PRESERVE atoms: their establishing chart sentences MUST appear
↪  verbatim in the output. If a preserve sentence has the same
↪  vocabulary as a target, **rewrite the TARGET sentence to differ
↪  from the preserve sentence**, do not delete the preserve sentence.
- Do NOT introduce new diagnoses, new lab values, or new findings.
- Do NOT specify qualifiers for NULL atoms (see OTHER ATOMS list below).
- If a target atom is currently NULL (silent in chart) and needs to
↪  flip to TRUE, ADD a single short sentence; do not rephrase existing
↪  sentences.

# === FEW-SHOT EXAMPLES ===

## EXAMPLE 1 — GOOD (minimal edit, preserve intact)

ORIGINAL: "67yo woman, post-femoral catheterization, presents with cool
↪  right foot and pulsatile right groin mass with bruit. Distal pulses
↪  absent."

TARGETS: flip `patient_has_acute_limb_ischemia` to FALSE.
PRESERVE: `patient_has_chronic_limb_ischemia_signs` (chart sentence:
↪  "Distal pulses absent.")

<plan>
a) Target establishing sentences: "cool right foot" and "Distal pulses
↪  absent" both contribute to acute ischemia, but "Distal pulses
↪  absent" is ALSO the preserve evidence.
b) Preserve sentence (verbatim): "Distal pulses absent."
c) Minimal edit: remove "cool right foot" (target-only) and "presents
↪  with ... pulsatile groin mass with bruit" (target). KEEP "Distal
↪  pulses absent."
d) Adjacent vocabulary: "Distal pulses" overlaps. Plan: leave the
↪  preserve sentence completely intact; only edit the acute-ischemia
↪  sentence.
</plan>
<chart>
67yo woman, post-femoral catheterization. Distal pulses absent. No
↪  acute findings on exam.
</chart>

## EXAMPLE 2 — BAD (deleted too much, broke the preserve)

Same inputs. WRONG rewrite:
<chart>
67yo woman, post-femoral catheterization. No vascular abnormalities
↪  noted.
</chart>
WHY WRONG: Removed "Distal pulses absent." — the preserve sentence is
↪  gone. Now the patient no longer supports the chronic-ischemia
↪  inclusion.

# === INPUTS ===

ORIGINAL CHART:
{{CHART}}

TARGETS to flip (atom = desired_value):
{{TARGETS}}

PRESERVE (atomic facts that must stay supported, with chart evidence):
{{PRESERVE}}

OTHER ATOMS (must remain at current value when re-mined; silents stay
↪  silent):
{{OTHER_ATOMS}}

# === OUTPUT ===
First `<plan>...</plan>` (your private analysis), then
↪  `<chart>...</chart>` (final rewrite).
```

D.6.3 ZSPM min-flip extractor

```
You are auditing a clinical-trial matcher's per-criterion assessments.
↪  The matcher determined the patient is INELIGIBLE. You will identify
↪  the MINIMUM set of chart facts that, if changed, would flip the
↪  matcher's verdict to ELIGIBLE.

PATIENT CHART:
{chart}

TRIAL CRITERIA:
{trial}

MATCHER'S PER-CRITERION ASSESSMENTS:
{assessments}

For each criterion the matcher marked as not_met, classify it:
  - DETERMINISTIC: chart explicitly contradicts the criterion (or fires
  ↪  it on exclusion side)
  - DEFERRAL: chart is silent; matcher defaulted to not_met but a visit
  ↪  could resolve it

Identify the smallest set of DETERMINISTIC blockers whose chart facts,
↪  if changed, would result in the matcher re-running and outputting
↪  eligible=true. Include each blocker's current chart fact and a
↪  proposed target chart fact.

Output STRICT JSON:
{{
  "min_flip_blockers": [
    {{"criterion": "<criterion text>",
      "current_chart_fact": "<what the chart says now>",
      "target_chart_fact": "<what the chart would need to say to flip>",
      "side": "inclusion|exclusion"}}
  ],
  "deferred_at_visit": [
    {{"criterion": "<criterion text>", "reason": "<why deferred>"}}
  ]
}}
```

D.6.4 Strict validator (gpt-4.1)

```
# === ROLE ===
You are auditing a counterfactual chart rewrite. Your audit must be
↪  **explicit, one item at a time**. For each cited blocker, produce
↪  ONE structured row. For each preserve item, produce ONE structured
↪  row. Every row MUST cite verbatim chart evidence — short quoted
↪  spans from BOTH the original chart AND the rewritten chart. Do not
↪  summarize. Do not skip items. Do not collapse items.

# === HOW TO AUDIT EACH BLOCKER (one row per blocker) ===

For each cited blocker, you MUST produce a row with these fields:

  - `blocker_label`         : 4-10 word natural-language label of the
  ↪  blocker
  - `original_evidence`     : a short verbatim quote (≤25 words) from
  ↪  the ORIGINAL chart that established this blocker. If the blocker
  ↪  was a SILENCE in the original (no mention), write `"<absent in
  ↪  original — silence blocker>"`
  - `cf_evidence`           : a short verbatim quote (≤25 words) from
  ↪  the REWRITTEN chart that either negates / removes the blocker OR
  ↪  (for silence blockers) asserts the inclusion-supporting fact. If
  ↪  the rewrite is silent where it should not be, write `"<no
  ↪  relevant content in CF>"`
  - `flip_type`             : exactly one of `"negated"`, `"removed"`,
  ↪  `"silence_resolved_by_addition"`, `"unchanged"`
  - `flipped`               : true if `flip_type` ∈ −negated, remo⌋
      ved, silence_resolved_by_addition", elsefalse
  - `reasoning`             : one sentence explaining why the cited
  ↪  evidence demonstrates the flip (or why it doesn't)

A blocker counts as `flipped: true` ONLY when:
  (a) `cf_evidence` contains a verbatim phrase from the rewritten
  ↪  chart, AND
  (b) that phrase clearly expresses the opposite polarity (or asserts
  ↪  the silence-resolving fact).

Hand-waving, paraphrase, or "the chart no longer implies X" without a
↪  quote = `flipped: false`.

# === HOW TO AUDIT EACH PRESERVE ITEM (one row per preserve) ===

For each preserve item (chart fact that MUST remain supported), produce
↪  a row:

  - `preserve_label`       : natural-language label
  - `original_evidence`    : verbatim quote from ORIGINAL chart
  ↪  establishing it
  - `cf_evidence`          : verbatim quote from REWRITTEN chart still
  ↪  supporting it
  - `still_supported`      : true if `cf_evidence` is a verbatim quote
  ↪  substantively matching `original_evidence`'s clinical meaning,
  ↪  else false
  - `reasoning`            : one sentence

If no PRESERVE items are provided, return `preserve_rows: []` and
↪  `no_overcorrection: true` (you have nothing to violate).

# === SCAN FOR ADJACENT-FINDING DELETION ===

Even if no formal preserve list is given, scan for clinical content
↪  present in ORIGINAL but absent from CF that would support an
↪  inclusion criterion:
  - Symptoms (e.g., "cool foot", "chest pain", "fatigue")
  - Exam signs (e.g., "absent distal pulses", "bilateral crackles")
  - Lab values, vital signs
  - Demographics or procedure history
  - Diagnoses other than the cited blocker

For each such removed item, add an `adjacent_deletion` row:
  - `removed_content`     : what was removed
  - `original_evidence`   : verbatim quote from ORIGINAL chart
  - `severity`            : "high" if it would support a trial
  ↪  inclusion criterion, "low" otherwise

Only `high` severity adjacent deletions cause `no_overcorrection:
↪  false`.

# === FINAL VERDICT ===

`all_flipped` = (every blocker row has `flipped: true`)
`no_overcorrection` = (every preserve row has `still_supported: true`)
↪  AND (zero `adjacent_deletion` rows with severity == "high")
`valid` = `all_flipped` AND `no_overcorrection`

You MUST NOT mark `valid: true` if any individual blocker row is
↪  `flipped: false` or any preserve row is `still_supported: false` or
↪  any adjacent_deletion is `high` severity.

# === OUTPUT (STRICT JSON) ===
{
  "blocker_rows": [
    {"blocker_label": "...", "original_evidence": "...", "cf_evidence":
    ↪  "...",
     "flip_type": "...", "flipped": true|false, "reasoning": "..."},
     ...
  ],
  "preserve_rows": [
    {"preserve_label": "...", "original_evidence": "...",
    ↪  "cf_evidence": "...",
     "still_supported": true|false, "reasoning": "..."},
     ...
  ],
  "adjacent_deletions": [
    {"removed_content": "...", "original_evidence": "...", "severity":
    ↪  "high"|"low"},
     ...
  ],
  "n_blockers": <int>, "n_blockers_flipped": <int>,
  "n_preserve": <int>, "n_preserve_supported": <int>,
  "n_high_adjacent_deletions": <int>,
  "all_flipped": true|false,
  "no_overcorrection": true|false,
  "valid": true|false,
  "summary": "<one short sentence>"
}

# === INPUTS ===

ORIGINAL CHART:
{{ORIGINAL_CHART}}

REWRITTEN (COUNTERFACTUAL) CHART:
{{CF_CHART}}

CITED BLOCKERS (one row per item in your output):
{{CITED_FACTS}}

PRESERVE ITEMS (one row per item, if any):
{{PRESERVE_FACTS}}
```

D.6.5 Simclin validator (gpt-5)

```
You are a senior clinical-trial coordinator at the PRESCREEN stage.

ORIGINAL CHART:
{original_chart}

MODIFIED CHART (counterfactual):
{cf_chart}

INCLUSION CRITERIA:
{inclusion}

EXCLUSION CRITERIA:
{exclusion}

CITED BLOCKERS the modifier was asked to address:
{cited_block}

Return strict JSON:
{{
  "coherent":          true|false
  "flips_target_atom": true|false
  "keeps_other_facts": true|false
  "oracle_should_flip":true|false
  "explanation":       "<one sentence justification>"
}}
```

D.7 User Interface for Clinician Counterfactual Audit

Figure 2 shows a screenshot of the annotation interface presented to the clinician during the counterfactual audit described in Appendix D.5. The interface implements a task type: Counterfactual clinical-validity judgment.

Counterfactual clinical-validity judgment (Figure 2). For each sampled counterfactual, the clinician is shown the original patient chart, the trial eligibility criteria, the modified chart with edits highlighted, and the flip targets the modifier was instructed to address. The clinician then marks the modification as **CLINICALLY VALID** — the edits address the cited blockers, do not introduce new ineligibility grounds, and leave the chart clinically coherent — or **NOT CLINICALLY VALID** if one or more of these conditions fails, and provides a one-line written reason for the decision.

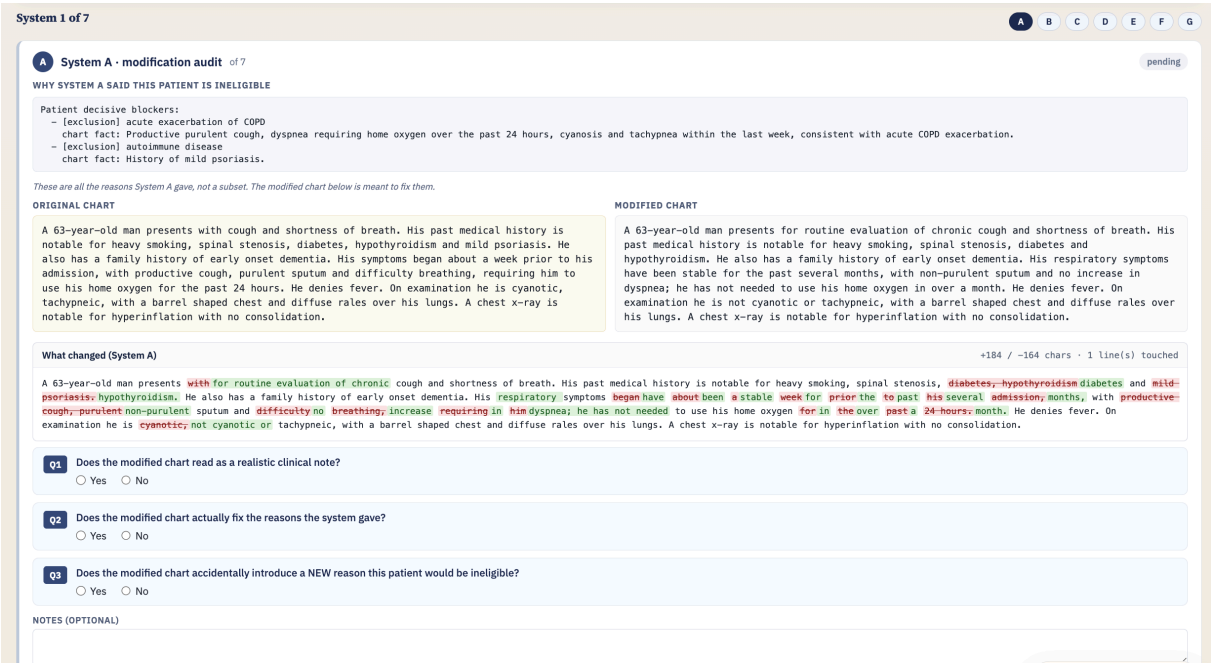


Figure 2: Annotation interface for the counterfactual clinical-validity audit.

E Example Rationales

This appendix illustrates what each system's rationale looks like on two patient–trial pairs, using each system's native output form — SMTMATCH's atom-grounded bullet structure, LLMMATCH's free-form aggregation prose, ZSPM's per-criterion met / not-met list, CoT LLM's strict-JSON per-criterion schema, and TrialGPT's two-pass criterion- index JSON. All systems run on `gpt-4.1` with `temperature` $= 0$. Verdicts are each system's own; the reference label is shown in each pair header.

E.1 Pair A

Pair: `sigir-20158__NCT00393913`. A 10-year-old boy with classic obstructive-sleep-apnea symptoms (nighttime snoring, witnessed apneas, excessive daytime sleepiness). *Reference label: eligible.*

SMTMATCH (verdict: eligible).

> A 10-year-old boy with prominent snoring, daytime sleepiness, and stable medical history satisfies the trial's core inclusion criteria for obstructive sleep apnea (OSA); no exclusion is triggered. Most exclusion and some inclusion criteria are not addressed in the chart and will require confirmation at the in-person visit.
>
> The clinical picture is consistent with pediatric OSA: the patient experiences snoring, excessive daytime sleepiness, and has a stable medical history without documented medication changes. The trial's inclusion criteria are directly met by these findings, and there is no evidence of exclusionary conditions or comorbidities in the chart.
>
> **Verdict:** Eligible
>
> **Inclusions met:**
>
> - Inclusion 1 — met: symptoms of OSA, including snoring and sleepiness (chart: "nighttime snoring," "excessive sleepiness during class").
> - Inclusion 2 — met: stable medical history with no change in medications affecting sleepiness.
>
> **No exclusions apply:**
>
> - Exclusion 1 — not applicable: no suspected diagnosis of a sleep disorder other than OSA.
> - Exclusion 5 — not applicable: not pregnant (chart: "10 yo boy").
>
> **To confirm at the in-person visit:**
>
> - **Medication history:** current medication use, change in medications affecting sleepiness, psychotropic medication use.
> - **Comorbidities:** medically unstable conditions, recreational drug use, alcohol abuse history, cognitive impairment.
> - **Communication and sensory:** inability to communicate verbally, visual or hearing impairment.

LLMMATCH (verdict: ineligible).

> The matcher determined that the patient meets the inclusion criterion of experiencing symptoms of obstructive sleep apnea, as evidenced by documented nighttime snoring, observed pauses in breathing, and reports of excessive sleepiness during class. However, the matcher found that the patient's stability of medical history and any recent changes in medications that could affect sleepiness were not addressed in the chart. On the exclusion side, the matcher noted that there was no information in the chart regarding suspected diagnoses of other sleep disorders, medically unstable health conditions, recent use of psychotropic medications, history of recreational drug or alcohol use, inability to communicate, or the presence of visual, hearing, or cognitive impairment. The criterion regarding pregnancy was explicitly addressed, with the matcher concluding that the patient, being a 10-year-old boy, is ineligible on this basis, as pregnancy is not applicable. Ultimately, the matcher judged the patient ineligible for the trial, citing the patient's age — since the chart documents he is 10 years old and the trial does not specify inclusion of pediatric patients — as the decisive factor for ineligibility.

CoT LLM (verdict: eligible).

{"inclusion_criteria": [{"text": "Experiences symptoms of OSA, including snoring and sleepiness", "status": "included", "reasoning": "The patient has nighttime snoring, pauses in breathing, and excessive sleepiness during class, which are symptoms of OSA."}, {"text": "Stable medical history with no change in medications that could affect sleepiness", "status": "unknown", "reasoning": "The chart does not mention any medical history or medication changes; this information is not provided."}], "exclusion_criteria": [{"text": "Suspected diagnosis of a sleep disorder other than OSA (i.e., periodic leg movements, narcolepsy, insomnia, central sleep apnea, sleep hypoventilation syndrome)", "status": "not_excluded", "reasoning": "There is no mention of other sleep disorders; symptoms described are consistent with OSA only."}, {"text": "Medically unstable health conditions (e.g., heart attack, congestive heart failure)", "status": "not_excluded", "reasoning": "No evidence of medically unstable health conditions is provided in the chart."}, {"text": "Use of psychotropic medications that cause sedation in the 3 months prior to study entry", "status": "unknown", "reasoning": "The chart does not mention any medication use, including psychotropic medications."}, {"text": "Recent or confirmed history of recreational drug use or alcohol abuse", "status": "not_excluded", "reasoning": "There is no mention of recreational drug use or alcohol abuse in the chart."}, {"text": "Pregnant", "status": "not_excluded", "reasoning": "The patient is a 10-year-old boy, so pregnancy does not apply."}, {"text": "Inability to communicate verbally, write, or read", "status": "not_excluded", "reasoning": "There is no evidence of communication impairment; the patient attends school and interacts with teachers."}, {"text": "Visual, hearing, or cognitive impairment", "status": "not_excluded", "rea-

soning": "No visual, hearing, or cognitive impairment is mentioned; the patient attends school and is described as having declining grades, not impairment."}], "aggregation_reasoning": "Inclusion 1: included. Inclusion 2: unknown — applied prescreen-doctrine, treated as included since it is not population-defining. Exclusion 1: not_excluded. Exclusion 2: not_excluded. Exclusion 3: unknown — applied prescreen-doctrine, treated as not_excluded since it is not population-defining. Exclusion 4: not_excluded. Exclusion 5: not_excluded. Exclusion 6: not_excluded. Exclusion 7: not_excluded. Therefore: ELIGIBLE.", "eligibility": "eligible"}

TrialGPT (no patient-level verdict; two passes).

Inclusion pass: {"0": ["The patient experiences snoring, pauses in breathing, restlessness, and excessive sleepiness during class, which are symptoms of OSA.", [0, 2], "included"], "1": ["There is no information about the patient's medical history stability or medication changes.", [], "not enough information"]}

Exclusion pass: {"0": ["There is no mention of a suspected diagnosis of a sleep disorder other than OSA; symptoms described are consistent with OSA.", [0], "not excluded"], "1": ["No evidence of medically unstable health conditions is provided.", [], "not enough information"], "2": ["No mention of psychotropic medication use.", [], "not enough information"], "3": ["No mention of recreational drug use or alcohol abuse.", [], "not enough information"], "4": ["The patient is a 10-year-old boy, so pregnancy is not applicable.", [], "not applicable"], "5": ["No mention of inability to communicate verbally, write, or read.", [], "not enough information"], "6": ["No mention of visual, hearing, or cognitive impairment.", [], "not enough information"]}

E.2 Pair B

Pair: `sigir-20153__NCT02596555`. A 65-year-old man with acute pulmonary embolism following hip-replacement surgery; the trial enrolls intermediate-risk PE patients. *Reference label: eligible.*

SMTMATCH (verdict: eligible).

A 65-year-old man meets the trial's minimum age requirement of 18 years, which is decisive; no exclusion is triggered. All other inclusion and exclusion criteria are not documented in the chart and will require confirmation at the in-person visit.

The chart picture is of a man presenting with acute onset shortness of breath and chest pain following recent orthopedic surgery, consistent with possible acute pulmonary embolism, but only age is confirmed. The trial's diagnostic, risk stratification, and safety criteria are not addressed in the chart and must be verified at screening.

Verdict: Eligible

Inclusions met:

- Inclusion 1 — met: age $\geq$ 18 years (chart documents "65 yo male").

No exclusions apply:

- Exclusion 1 — not applicable: not pregnant or lactating (chart: male sex).
- Exclusion 2 — not applicable: not a woman of childbearing potential (chart: male sex).

To confirm at the in-person visit: (12 items, condensed: imaging-based PE confirmation, hemodynamics, sPESI / NT-proBNP / troponin, procedural history, bleeding risk, comorbidities, hypersensitivity, anticoagulant washout, follow-up logistics).

ZSPM (verdict: ineligible).

Inclusion criteria (per-criterion verdict):

met	The patient is 65 years old, which is $\geq$ 18 years.
not met	There is no mention of an objectively confirmed diagnosis of acute PE by imaging. Symptoms are suggestive, but confirmation is not documented.
met	The patient presents with tachypnea and shortness of breath but there is no mention of hemodynamic collapse.
not met	sPESI score is not explicitly calculated… at least one point for age >65. However, the presence of additional required criteria is not documented.
not met	No imaging (CT or echo) findings of RV pressure overload/dysfunction are provided.
not met	No troponin levels or mention of myocardial injury are provided.
not met	No NT-proBNP levels or mention of RV failure are provided.
not met	No information is provided about the patient's ability to consent or understanding of the trial.

Exclusion criteria (per-criterion verdict):

met	The patient is a 65-year-old male, so pregnancy and lactation do not apply.
met	(… seven additional [met] entries for criteria not contradicted by the chart.)

F Dataset Statistics

Our evaluation set comprises 552 patient–trial pairs built from the SIGIR 2016 clinical-trial cohort: 59 patient vignettes, each paired with its top-10 TrialGPT-Retrieval trial candidates (590 pairs), of which 552 obtain complete five-judge gold coverage (Appendix H). Gold labels are near-balanced (278 ELIGIBLE / 274 INELIGIBLE). Table 7 reports the full statistics.

The 38 excluded pairs could in principle distort the benchmark if they were concentrated on particular patients or retrieval ranks. Figures 3 and 4 therefore examine attrition by patient and by TrialGPT-Retrieval rank. The exclusions are diffuse: most patients retain all 10 candidate pairs, no patient loses more than four, and the dropped pairs show no monotonic pattern across retrieval rank. This suggests that the decisive 552-pair set preserves the structure of the original 590-pair candidate pool.

Property	Value
Patient–trial pairs	552
ELIGIBLE	278
INELIGIBLE	274
Unique patients	59
Unique trials	479
Pairs per patient	6–10 (mean 9.4)
Patient vignette length (words)	70 (median), 38–204
Trial text length (words)	243 (median), 47–1,595

Table 7: Evaluation dataset statistics.

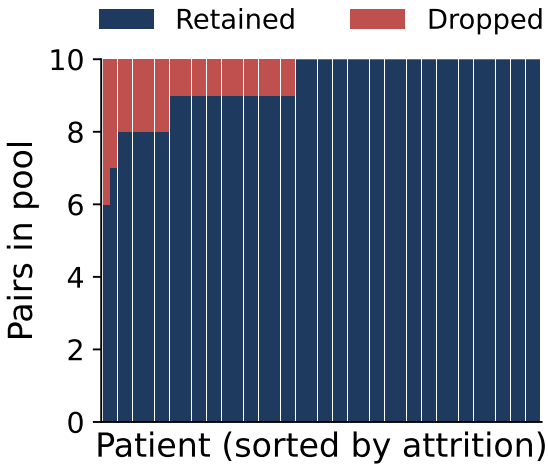


Figure 3: Coverage attrition is evenly distributed across patients. Each of the 59 patients contributes 10 candidate pairs to the 590-pair pool; 38 pairs are dropped for lack of complete five-judge coverage. No patient loses more than 4 pairs, and 33 of 59 patients lose none — the 552-pair evaluation set is not biased toward any patient subset.

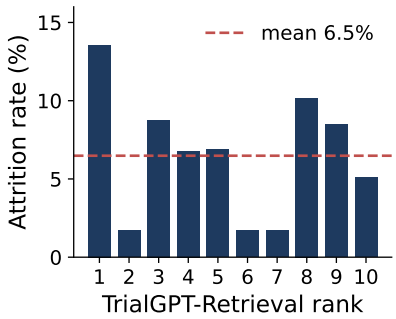


Figure 4: Coverage attrition is uniform across retrieval rank. Each TrialGPT-Retrieval rank contributes 57–59 candidate pairs to the 590-pair pool; the 38 dropped pairs show no monotonic trend with rank (per-rank attrition 1.7–13.6%, mean 6.5%). The 552-pair evaluation set is therefore not biased toward easily- or poorly-retrieved trials.

G Descriptive Statistics

Where Appendix F reports basic corpus properties (counts of patients, trials, and retrieval-derived patient–trial pairs), this appendix reports the *output-side* statistics: how the five matchers' verdicts distribute, what rationales they produce, how the systems disagree with each other and with the reference panel, and the composition of the clinician audit sample.

G.1 Per-system verdict balance and rationale length

Table 8 shows, for each matcher, the fraction of the 552-pair comparison set labelled *eligible*, alongside the median and mean character length of the system's free-text rationale. Two patterns stand out. First, eligibility-leaning systems (SMTMATCH = 55.4%, ZSPM = 54.0%) come close to the reference base rate (50.4% eligible), while the prompted-LLM family (LLMMATCH = 35.0%, CoT LLM = 36.1%) is markedly ineligibility-leaning. Second, rationale length varies by a factor of ≈ 2 across systems, from CoT LLM's compact per-criterion lists ($\approx$ 590 chars median) to ZSPM's exhaustive per-criterion breakdown ($\approx$ 1340 chars median), and these length differences are not collinear with accuracy — the longest rationale is not the most-preferred in the clinician audit (Table 2).

System	Elig. (%)	Median chars	Mean chars
SMTMATCH	55.4	1021	1030
LLMMATCH	35.0	1195	1187
CoT LLM	36.1	592	728
TrialGPT	44.4	594	708
ZSPM	54.0	1342	1315
Reference panel	*50.4*	—	—

Table 8: Verdict balance and rationale length per system on the 552-pair comparison set.

G.2 Disagreement strata

For each SMTMATCH-vs-baseline pair, every comparison-set pair falls into one of four strata by joint agreement with the reference panel: *both_right* (both match), SMTMATCH*_right* (only SMTMATCH matches), *comp_right* (only the baseline matches), *both_wrong* (neither matches). Table 9 shows the corpus-wide distribution per baseline at the GPT-4.1 backbone. Two observations: (1) all four strata are populated for every baseline — no baseline is strictly dominated — so the audit can sample discriminating pairs in both directions; (2) disagreement strata are $\approx$ 30–33% of the corpus, large enough that the audit's stratified $4/4/4/4$ design rests on real population mass, not on hand-picked outliers.

G.3 Reference-panel agreement

Within the 5-judge GPT-5 panel, 445 of 552 pairs (80.6%) receive a unanimous 5–0 verdict; the remaining 107 split 4–1 (69 pairs, 12.5%) or 3–2 (37 pairs, 6.7%). The split distribution is roughly symmetric across the two verdict directions (43+26 eligible-leaning vs ineligible-leaning 4–1; 20+17 at 3–2). This intra-panel disagreement is the upstream cause of the gap reported in Sec. 5.2 between a single-call GPT-5 matcher and the 5-judge consensus.

G.4 Clinician audit sample composition

The audit comprises 32 patient–trial pairs: 16 for the SMTMATCH-vs-ZSPM comparison and 16 for SMTMATCH-vs- LLMMATCH, each balanced $4/4/4/4$ across the four disagreement strata above. All 32 pairs receive a holistic A/B/tie preference and per-axis Likert ratings on five axes (Sec. 5.4); on the 15 pairs where the clinician's initial verdict disagreed with the reference panel, the same clinician produced a second judgement on a re-read with the rationales and the per-framing judge breakdown visible. The re-review outcome distribution is shown in Table 10: the modal outcome was AMBIGUOUS, four revisions all moved *toward* the reference label, and only a single case stood as a clear panel–clinician disagreement after re-review.

G.5 Chart and trial-criterion length

Patient charts in the SIGIR corpus are unusually compact compared to hospital-EHR notes: median chart length is 440 characters (mean 494, range 245–1,290). Trial documents are larger: median total text $\approx$ 1,535 characters, of which median inclusion section 313 chars and median exclusion section 358 chars. Both distributions are right-skewed; the longest trial in the corpus exceeds 5,000 characters of criteria text. These length statistics matter for prompting: a single chart+trial prompt fits comfortably in a 4–8 K context, and we do not need any chunking or retrieval-within-trial logic to handle the longest cases.

Baseline	both_right	SMTMATCH_right	comp_right	both_wrong
LLMMATCH	67.4	15.2	12.7	4.7
CoT LLM	68.7	13.9	12.0	5.4
TrialGPT	67.5	15.1	9.8	7.6
ZSPM	68.3	14.3	11.4	6.0

Table 9: Disagreement stratum distribution (%) per SMTMATCH-vs-baseline at GPT-4.1, computed on the 552-pair comparison set. Rows sum to 100% modulo rounding.

Re-review outcome	Count
AMBIGUOUS	10
REVISE → eligible	2
REVISE → ineligible	2
UNCHANGED (against panel)	1
Total	15

Table 10: Outcome distribution of the 15-pair clinician re-review, applied to every pair where the initial clinician verdict disagreed with the reference panel.

H Reference-label derivation pipeline

This appendix documents the pipeline used to derive the binary *eligible/ineligible* reference labels used in Section 5. The pipeline takes a patient–trial pair as input, queries five rubric-framing LLM judges in parallel, and assigns a reference label by simple majority vote. These labels are used for relative system comparison; they are not intended to replace clinician adjudication.

H.1 Pipeline overview

1. **Pair pool construction.** We start from the 590-pair retrieval shortlist, consisting of 59 SIGIR patients and the top-10 retrieved trial candidates for each patient.

2. **Candidate-rationale collection.** For each pair, we collect five candidate rationales from the matchers being compared. Before judge review, each rationale is rewritten into a shared prose format by a fidelity-preserving verbalizer. The verbalizer normalizes presentation only: it is instructed to narrate the matcher's existing decision artifact without introducing new clinical inferences.

3. **Five-judge ensemble.** The patient chart, trial text, and five blinded candidate rationales are presented to five LLM judges. The judges use the same intended eligibility policy but receive different rubric framings: pragmatic screening, protocol adherence, schema-based review, checklist-driven scoring, and explanation-focused assessment. Candidate rationale slots are randomized independently for each pair and framing.

4. **Independent verdict extraction.** Each judge first produces an independent verdict, *eligible*, *ineligible*, or *inconclusive*, before rating any candidate rationale. Only the independent verdict is used for reference-label aggregation. Candidate rationale ratings from the same calls are used only for the rationale-quality analysis in Section 5.4 and Appendix K

5. **Majority aggregation.** A pair receives a reference label only when all five judges return decisive verdicts. The label is *eligible* when at least three of the five judges vote eligible, and *ineligible* otherwise.

6. **Coverage attrition.** Of the 590-pair pool, 552 pairs receive complete five-judge coverage and constitute the final reference-evaluation set. The remaining 38 pairs are excluded because at least one required candidate artifact or judge output was unavailable.

H.2 Why a five-member LLM reference panel?

Constructing eligibility judgments at this scale requires a practical trade-off. Clinician review is the most clinically meaningful form of validation, but it is difficult to apply redundantly to hundreds of patient–trial pairs. A single LLM judge is scalable, but can be sensitive to rubric wording on borderline cases even when the intended eligibility policy is unchanged [23, 13]. We therefore use a five-member LLM reference panel as a scalable comparison protocol, not as a clinical gold standard. The panel estimates which verdicts are stable under different presentations of the same eligibility policy.

The five framings vary the presentation and decision posture of the rubric while keeping the intended policy fixed. They are not random paraphrases; they cover common ways eligibility decisions are specified or requested in practice.

Pragmatic screening. This framing presents the task as a prescreening decision: whether the patient should remain a plausible candidate for the trial and which issues would require follow-up.

Protocol adherence. This framing presents the task as a protocol-compliance decision: whether the available chart evidence establishes that the patient satisfies the inclusion criteria and violates no exclusion criteria under the stated policy.

Schema-based review. This framing expresses the same decision policy as an operational procedure: identify inclusion requirements, identify exclusion requirements, map patient evidence to each requirement, track missing or uncertain facts, and then produce the final verdict.

Checklist-driven scoring. This framing is deliberately terse and criterion-driven. It asks the model to treat the trial as a checklist, helping expose cases where a narrative prompt may skip less salient criteria, hidden conjuncts, temporal restrictions, exception clauses, or exclusions.

Explanation-focused assessment. This framing uses a naturalistic explanation-oriented style, reflecting how eligibility systems are often queried

in practice: users ask not only for a verdict but also for a readable justification.

Together, the five framings span key prompt degrees of freedom that could otherwise become hidden evaluation artifacts: screening versus protocol-compliance posture, clinical versus procedural voice, narrative versus checklist structure, and explanation-oriented versus canonical decision form. A pair enters the reference-evaluation set only when all five framings produce decisive verdicts; the majority vote is then used as an LLM-based reference label for relative system comparison.

H.3 Aggregation rule

Let $J = \{j_1, \ldots, j_5\}$ denote the five judge framings and $v_j(p) \in \{\text{ELIG}, \text{INELIG}, \perp\}$ the verdict of judge j on pair p, where $\perp$ denotes a non-decisive output, parse failure, or model error. Let $n_{\text{elig}}(p) = |\{j \in J : v_j(p) = \text{ELIG}\}|$. The reference label $r(p)$ is defined as:

$$r(p) = \begin{cases} \text{ELIG} & n_{\text{elig}}(p) \geq 3 \\ & \text{and } \perp \notin \{v_j(p) : j \in J\}, \\ \text{INELIG} & n_{\text{elig}}(p) \leq 2 \\ & \text{and } \perp \notin \{v_j(p) : j \in J\}, \\ \text{UNDEFINED} & \text{otherwise.} \end{cases} \quad (1)$$

The 552 evaluation pairs distribute as 268 eligible and 284 ineligible under this rule.

H.4 Judge prompts

The five prompts below differ in rhetorical framing but encode the same eligibility policy. Placeholder tokens for the patient note, trial text, and candidate rationales are substituted at call time. Each judge outputs strict JSON; the independent verdict field is the vote used in the majority aggregation.

H.4.1 Pragmatic-screening judge prompt.

```
# === FACTUAL-EXCLUSION OVERRIDE (READ FIRST, OVERRIDES THE PRESCREEN
↪  DOCTRINE BELOW) ===

The prescreen-forward doctrine you will see below applies to **silence
↪  or
uncertainty**, NOT to facts. When the chart contains explicit, factual
evidence that an exclusion criterion is met, or that an inclusion
criterion is contradicted, the patient is **ineligible** — do not
↪  forward.

## The state-change rule

The prescreen-forward doctrine assumes that what the in-person visit
clarifies are **logistical gaps**, not changes to the patient's
↪  clinical
state. Specifically:

- **Logistical gaps** the visit CAN fill: missing scans (CT/MRI/PET),
  missing labs, missing histology, criteria-application not yet
  ↪  performed,
  documentation gaps, missing measurements (e.g., dysphagia score,
  Glasgow motor sub-score), un-applied named-criteria sets (CDC, ACR,
  GOLD). Defer these to the visit.
- **Patient-state facts** the visit CANNOT change: an actively present
  exclusion condition (active GI bleed, current pregnancy,
  ↪  decompensated
  HF), labs/vitals/staging documented outside the trial's thresholds
  (CrCl 5 with a CrCl
      $\geq 30 inclusion; ECOG4 with ECOG0\tilde{}1; stage IV$
  with stage I-III inclusion), a clinically clearly different patient
  population (cardiology trial, chart is healthy outpatient pap smear).

**Do NOT assume non-logistical state changes will make the patient
eligible.** A patient whose chart documents an active disqualifying
↪  state
will still have that state at the visit. Forwarding such a patient is
not "preserving recall" — it is ignoring chart-stated facts.

## Concrete factual-exclusion cases

- If the chart documents an active condition the trial excludes (e.g.,
  active GI bleed, current pregnancy, decompensated HF, recent stroke,
  active malignancy other than the trial's target), the patient is
  ineligible. Do not forward "for clarification".
- If the chart names a medication on the exclusion list (e.g., warfarin
  for a thrombolysis trial, a P-gp inhibitor for a P-gp-substrate trial,
  systemic immunosuppression for an immunotherapy trial), the patient
  is ineligible.
- If the chart describes a clinical scenario clearly outside the trial's
  domain (oncology trial, but chart is an isolated orthopedic injury
  with no oncologic evidence; cardiology trial, but chart is a healthy
  outpatient pap smear), the patient is ineligible.
- If the chart shows lab values, vital signs, or staging clearly outside
  the trial's explicit numerical thresholds (e.g., creatinine 5.5 with
  a creatinine ≤ 2.0 inclusion; ECOG 4 with an ECOG 0-1 inclusion),
  the patient is ineligible.

## Summary

Be **factual**, not deferential. The prescreen-forward doctrine exists
to avoid rejecting on **missing data** that the in-person visit can
supply — it is NOT a license to forward patients whose chart already
contains decisive disqualifying facts, and it does NOT permit assuming
the patient's clinical state will spontaneously change between
↪  prescreen
and visit. Apply common sense: defer logistical gaps to the visit;
exclude on documented disqualifying state.

When in genuine doubt (chart silent, ambiguous, or weakly inferable),
the prescreen doctrine below applies — forward to visit. When the chart
states a disqualifying clinical state, exclude.

# === ROLE ===
You are a senior clinical trial coordinator reviewing five eligibility
↪  decisions
for the same patient-trial pair. You will first produce your own
↪  policy-compliant
verdict, then rate each candidate's decision and rationale against it.

# === PRESCREEN REVIEW POLICY ===

This is prescreen, not enrollment. The purpose is to decide whom to
↪  forward for
in-person screening, where missing information can be obtained and
↪  verified. A
prescreen chart is a brief vignette — it is expected to be incomplete.

For each trial criterion, one of three things is true about the chart:
- The chart provides evidence the patient **meets** the criterion.
- The chart provides evidence the patient **does not meet** the
↪  criterion.
- The chart is **silent** — it neither confirms nor denies.

**Rule the patient out only if the chart provides concrete evidence
↪  against them:**
a stated fact contradicting an inclusion requirement, or a stated fact
↪  triggering
an exclusion. If the chart is silent on a requirement, mark it for
↪  verification at
the in-person visit. Silence is not grounds for rejection; that is what
↪  the
screening visit is for.

Strong clinical inference from what is in the chart counts as evidence
(for example, "20 days post-mastectomy" is evidence of a cancer
↪  history). A bare
absence of mention is not evidence.

This policy reflects the asymmetric cost structure of prescreen review:
↪  a false
rejection denies a patient access to a trial; a false forward costs a
↪  coordinator
a short review. The policy is recall-preserving by design.

# === INPUTS ===

## Patient chart
#PATIENT_NOTE#

## Trial eligibility text
#TRIAL_TEXT#

## Five blinded candidate decisions + rationales
(Random slot assignment A/B/C/D/E; evaluate on substance only.)

### Candidate A
Decision: #DECISION_A#
Rationale: #RATIONALE_A#

### Candidate B
Decision: #DECISION_B#
Rationale: #RATIONALE_B#

### Candidate C
Decision: #DECISION_C#
Rationale: #RATIONALE_C#

### Candidate D
Decision: #DECISION_D#
Rationale: #RATIONALE_D#

### Candidate E
Decision: #DECISION_E#
Rationale: #RATIONALE_E#

# === YOUR TASK ===

Part 1 — **Your own policy-compliant verdict**. Apply the policy above.
↪  State
"eligible" or "ineligible". Give a brief justification (2-3 sentences)
↪  naming
the decisive chart evidence (or its absence).

Part 2 — **Evaluate each candidate on two axes**:
```

```
**(a) Decision-correctness**: Does the candidate's decision agree with your
verdict? Rate 1-5 where 5 = matches your verdict, 1 = clearly contradicts.

**(b) Rationale quality (5 sub-axes)**:

1. **Decisiveness** (1-5): Does the rationale name one crisp decisive fact
   (high) vs hedge or list many criteria (low)?

2. **Evidence specificity** (1-5): Does the rationale cite a specific quoted
   or paraphrased chart fact (high) vs speak in generalities (low)?

3. **Conciseness** (1-5): Is the rationale short enough to skim in under 10
   seconds (high) vs verbose (low)? Target: 1 short sentence.

4. **Actionability** (1-5): Does the rationale tell a coordinator exactly
   what to verify or override at the in-person visit (high) vs leave them
   to re-reason from scratch (low)?

5. **Logical consistency** (1-5): Does the rationale's reasoning chain
   make clinical sense? Does the decision follow logically from the cited
   fact? Are there internal contradictions between the rationale and the
   decision? Rate 5 = clear chain of reasoning, 1 = rationale contradicts
   decision or makes clinical errors.

Also produce a brief one-sentence `brief_comment` per candidate.

# === OUTPUT (STRICT JSON, no markdown fences) ===
{
  "independent_verdict": "eligible" | "ineligible",
  "independent_reasoning": "2-3 sentences citing the decisive chart evidence",
  "ratings": {
    "A": {
      "decision_correct": "yes" | "no" | "partially",
      "decision_rating": 1-5,
      "sharpness": {
        "decisiveness": 1-5,
        "evidence_specificity": 1-5,
        "conciseness": 1-5,
        "actionability": 1-5,
        "logical_consistency": 1-5
      },
      "brief_comment": "one sentence"
    },
    "B": { ... same structure ... },
    "C": { ... same structure ... }
  }
}
```

H.4.2 Protocol-adherence judge prompt.

```
# === FACTUAL-EXCLUSION OVERRIDE (READ FIRST, OVERRIDES THE PRESCREEN DOCTRINE BELOW) ===

The prescreen-forward doctrine you will see below applies to **silence or
uncertainty**, NOT to facts. When the chart contains explicit, factual
evidence that an exclusion criterion is met, or that an inclusion
criterion is contradicted, the patient is **ineligible** — do not forward.

## The state-change rule

The prescreen-forward doctrine assumes that what the in-person visit
clarifies are **logistical gaps**, not changes to the patient's clinical
state. Specifically:

- **Logistical gaps** the visit CAN fill: missing scans (CT/MRI/PET),
  missing labs, missing histology, criteria-application not yet performed,
  documentation gaps, missing measurements (e.g., dysphagia score,
  Glasgow motor sub-score), un-applied named-criteria sets (CDC, ACR,
  GOLD). Defer these to the visit.
- **Patient-state facts** the visit CANNOT change: an actively present
  exclusion condition (active GI bleed, current pregnancy, decompensated
  HF), labs/vitals/staging documented outside the trial's thresholds
  (CrCl 5 with a CrCl
      $\geq 30 inclusion; ECOG4 with ECOG0\check{}1; stageIV$
  with stage I-III inclusion), a clinically clearly different patient
  population (cardiology trial, chart is healthy outpatient pap smear).

**Do NOT assume non-logistical state changes will make the patient
eligible.** A patient whose chart documents an active disqualifying state
will still have that state at the visit. Forwarding such a patient is
not "preserving recall" — it is ignoring chart-stated facts.

## Concrete factual-exclusion cases

- If the chart documents an active condition the trial excludes (e.g.,
  active GI bleed, current pregnancy, decompensated HF, recent stroke,
  active malignancy other than the trial's target), the patient is
  ineligible. Do not forward "for clarification".
- If the chart names a medication on the exclusion list (e.g., warfarin
  for a thrombolysis trial, a P-gp inhibitor for a P-gp-substrate trial,
  systemic immunosuppression for an immunotherapy trial), the patient
  is ineligible.
- If the chart describes a clinical scenario clearly outside the trial's
  domain (oncology trial, but chart is an isolated orthopedic injury
  with no oncologic evidence; cardiology trial, but chart is a healthy
  outpatient pap smear), the patient is ineligible.
- If the chart shows lab values, vital signs, or staging clearly outside
  the trial's explicit numerical thresholds (e.g., creatinine 5.5 with
  a creatinine ≤ 2.0 inclusion; ECOG 4 with an ECOG 0-1 inclusion),
  the patient is ineligible.

## Summary

Be **factual**, not deferential. The prescreen-forward doctrine exists
to avoid rejecting on **missing data** that the in-person visit can
supply — it is NOT a license to forward patients whose chart already
contains decisive disqualifying facts, and it does NOT permit assuming
the patient's clinical state will spontaneously change between prescreen
and visit. Apply common sense: defer logistical gaps to the visit;
exclude on documented disqualifying state.

When in genuine doubt (chart silent, ambiguous, or weakly inferable),
the prescreen doctrine below applies — forward to visit. When the chart
states a disqualifying clinical state, exclude.

# === ROLE ===
You are an experienced clinical research coordinator reviewing three system
proposals for a single patient-trial pair. Form your own assessment first,
then evaluate each system's decision and rationale against it.

# === HOW PRESCREEN WORKS ===

The step you are doing is **prescreen**, not enrollment. The chart you have
is a brief vignette — by design, it is incomplete. Anything not in the chart
can be checked or filled in at the in-person screening visit. Your task is
to decide whether to forward the patient to that visit.

Each trial criterion is in one of three states relative to the chart:

- **Supported**: the chart provides evidence the patient meets the criterion.
- **Contradicted**: the chart provides evidence the patient does NOT meet it.
- **Unstated**: the chart neither asserts nor denies it.

**Forward the patient unless concrete evidence rules them out.** Only two
things are grounds for ruling out:
- a documented chart fact that contradicts an inclusion requirement, or
- a documented chart fact that triggers an exclusion criterion.

If the chart is silent on a requirement, mark that requirement for follow-up
at the in-person visit — silence is not grounds for rejection. The whole
point of an in-person visit is to clarify what the chart left unstated.

Reasonable clinical inference from chart content counts as evidence. For
example, "20 days post-mastectomy" tells you the patient has a cancer
history; "currently on dialysis" tells you they have end-stage renal
disease; "no known allergies" tells you allergic-reaction exclusions are
satisfied. Inferences must follow from explicit chart content; do not
fabricate facts.

# === INPUTS ===

## Patient note
#PATIENT_NOTE#

## Trial eligibility text
#TRIAL_TEXT#

## Five blinded candidate decisions and rationales
(The candidates have been randomly shuffled into slots A, B, C, D, E. Evaluate on
substance only; do not try to figure out which method produced which.)

### Candidate A
Decision: #DECISION_A#
Rationale: #RATIONALE_A#

### Candidate B
Decision: #DECISION_B#
Rationale: #RATIONALE_B#

### Candidate C
Decision: #DECISION_C#
Rationale: #RATIONALE_C#

### Candidate D
Decision: #DECISION_D#
Rationale: #RATIONALE_D#

### Candidate E
Decision: #DECISION_E#
Rationale: #RATIONALE_E#

# === YOUR TASK ===

Part 1 — **Your own assessment**. Apply the prescreen-review approach above
and determine your own verdict: "eligible" or "ineligible". Briefly explain
(2-3 sentences) the determining facts: which inclusion or exclusion criterion
was contradicted by the chart, or that no rule-out was found and the
remaining unstated criteria can be deferred.

Part 2 — **Rate each candidate** on:

(a) **Decision-correctness**: does the candidate's decision match your own?
    Rate 1-5 (5 = decisions agree, 1 = decisions oppose).
(b) **Rationale quality** (1-5 per sub-axis):
    - **decisiveness**: does the rationale commit to a single load-bearing
      reason for its decision?
    - **evidence_specificity**: are specific chart facts cited as support?
    - **conciseness**: is the rationale tight, no padding?
    - **actionability**: can a coordinator act on this rationale at the
      in-person visit (knows what to verify or rule out)?

Do NOT penalize a rationale for deferring uncertain criteria to the
in-person visit; deferral is the correct action when the chart is silent.

# === OUTPUT (STRICT JSON, no markdown fences) ===
{
  "independent_verdict": "eligible" | "ineligible",
  "independent_reasoning": "2-3 sentences citing the determining chart facts",
```

```
  "ratings": {
    "A": {
      "decision_correct": "yes" | "no" | "partially",
      "decision_rating": 1-5,
      "sharpness": {
        "decisiveness": 1-5,
        "evidence_specificity": 1-5,
        "conciseness": 1-5,
        "actionability": 1-5
      },
      "brief_comment": "one sentence"
    },
    "B": { ... same structure ... },
    "C": { ... same structure ... }
  }
}
```

H.4.3 Schema-based-review judge prompt.

```
# === FACTUAL-EXCLUSION OVERRIDE (READ FIRST, OVERRIDES THE PRESCREEN
↪  DOCTRINE BELOW) ===

The prescreen-forward doctrine you will see below applies to **silence
↪  or
uncertainty**, NOT to facts. When the chart contains explicit, factual
evidence that an exclusion criterion is met, or that an inclusion
criterion is contradicted, the patient is **ineligible** — do not
↪  forward.

## The state-change rule

The prescreen-forward doctrine assumes that what the in-person visit
clarifies are **logistical gaps**, not changes to the patient's
↪  clinical
state. Specifically:

- **Logistical gaps** the visit CAN fill: missing scans (CT/MRI/PET),
  missing labs, missing histology, criteria-application not yet
  ↪  performed,
  documentation gaps, missing measurements (e.g., dysphagia score,
  Glasgow motor sub-score), un-applied named-criteria sets (CDC, ACR,
  GOLD). Defer these to the visit.
- **Patient-state facts** the visit CANNOT change: an actively present
  exclusion condition (active GI bleed, current pregnancy,
  ↪  decompensated
  HF), labs/vitals/staging documented outside the trial's thresholds
  (CrCl 5 with a CrCl
      $\geq 30 inclusion; ECOG4 with ECOG0\tilde{}1; stageIV$
  with stage I-III inclusion), a clinically clearly different patient
  population (cardiology trial, chart is healthy outpatient pap smear).

**Do NOT assume non-logistical state changes will make the patient
eligible.** A patient whose chart documents an active disqualifying
↪  state
will still have that state at the visit. Forwarding such a patient is
not "preserving recall" — it is ignoring chart-stated facts.

## Concrete factual-exclusion cases

- If the chart documents an active condition the trial excludes (e.g.,
  active GI bleed, current pregnancy, decompensated HF, recent stroke,
  active malignancy other than the trial's target), the patient is
  ineligible. Do not forward "for clarification".
- If the chart names a medication on the exclusion list (e.g., warfarin
  for a thrombolysis trial, a P-gp inhibitor for a P-gp-substrate trial,
  systemic immunosuppression for an immunotherapy trial), the patient
  is ineligible.
- If the chart describes a clinical scenario clearly outside the trial's
  domain (oncology trial, but chart is an isolated orthopedic injury
  with no oncologic evidence; cardiology trial, but chart is a healthy
  outpatient pap smear), the patient is ineligible.
- If the chart shows lab values, vital signs, or staging clearly outside
  the trial's explicit numerical thresholds (e.g., creatinine 5.5 with
  a creatinine $\leq$ 2.0 inclusion; ECOG 4 with an ECOG 0-1 inclusion),
  the patient is ineligible.

## Summary

Be **factual**, not deferential. The prescreen-forward doctrine exists
to avoid rejecting on **missing data** that the in-person visit can
supply — it is NOT a license to forward patients whose chart already
contains decisive disqualifying facts, and it does NOT permit assuming
the patient's clinical state will spontaneously change between
↪  prescreen
and visit. Apply common sense: defer logistical gaps to the visit;
exclude on documented disqualifying state.

When in genuine doubt (chart silent, ambiguous, or weakly inferable),
the prescreen doctrine below applies — forward to visit. When the chart
states a disqualifying clinical state, exclude.

# === ROLE ===
You are evaluating five eligibility decisions against a well-defined
↪  decision
specification. Compute the specification-compliant verdict using the
↪  clinical-
inference policy below, then rate each candidate.

# === CLINICAL INFERENCE POLICY (applied when assigning T/F/N) ===

## Use of clinical inference
Clinical inference is permitted for both inclusion and exclusion
↪  criteria.
Patient notes (prescreen vignettes) may not explicitly contain every
↪  patient
fact; derive missing facts via clinical reasoning when the inference is
unambiguous. Apply the same inference strength symmetrically to
↪  inclusion and
exclusion. Do not infer high-acuity complaints or states from
↪  non-specific
severity signs alone.

For disease-state variables, wording such as "has diagnosis of X" means
↪  the
patient clinically having X now, not that the note must explicitly
↪  state the
diagnostic label. After projecting away diagnosis/documentation
↪  wording, apply
the same inference and missingness rules to the underlying disease
↪  state.

## Project away administrative / operational framing
When a criterion primarily reflects study operations, administrative
↪  cohort
construction, healthcare-record structure, or timing relative to study
↪  events
(care setting, hospitalization status, visit intent, intake/pre-index
↪  windows,
prescription-claim structure, same-day claims, etc.), project away the
operational framing. Keep the criterion in scope only if a substantive
clinical-state proposition remains; if so, evaluate that proposition
↪  under the
ordinary T/F/N rules.

## Future-state and logistical requirements
Do not derive facts that require change of the patient's state in the
↪  future.
Logistical-only requirements (consent, willingness to comply,
↪  scheduling) are
assumed satisfiable unless explicitly contradicted. Procedure
↪  requirements:
if the criterion asks the patient to have already undergone or be
↪  currently
undergoing a procedure that cannot be made up later, evaluate from the
↪  chart;
if the criterion only asks ability/willingness to undergo a procedure,
↪  assume
satisfiable unless contraindication is noted or inferred.

## Missingness and default-valuing
Patient prescreen vignettes typically mention significant
↪  positive/present
facts but not negative/absent ones. If a significant patient fact is not
supported anywhere in the note (i.e., absent), let the corresponding
↪  criterion
fall back to a commonsense default via clinical reasoning rather than
↪  treating
the absence as a positive contradiction.

# === T/F/N DECISION FUNCTION ===

For each criterion, after applying the policy above, assign:

- **T (true)**: chart (with clinical inference and commonsense
↪  defaults)
  supports the fact.
- **F (false)**: chart (with clinical inference and commonsense
↪  defaults)
  contradicts the fact.
- **N (null)**: even after inference and defaults, the value cannot be
  determined.

**Decision rule**: the patient is **ineligible** iff some inclusion
↪  criterion
evaluates to F or some exclusion criterion evaluates to T. Otherwise
**eligible**. Null facts do not contribute to ineligibility (asymmetric
↪  cost
structure: a false-negative denies a patient trial access; a
↪  false-positive
costs a coordinator a review).

# === INPUTS ===

## Patient note
#PATIENT_NOTE#

## Trial eligibility text
#TRIAL_TEXT#

## Five blinded candidate decisions + rationales
(The candidates have been randomly shuffled into slots A, B, C, D, E.
↪  Evaluate on
substance only; do not attempt to infer which method produced which.)

### Candidate A
Decision: #DECISION_A#
Rationale: #RATIONALE_A#

### Candidate B
Decision: #DECISION_B#
Rationale: #RATIONALE_B#

### Candidate C
Decision: #DECISION_C#
Rationale: #RATIONALE_C#

### Candidate D
Decision: #DECISION_D#
Rationale: #RATIONALE_D#

### Candidate E
Decision: #DECISION_E#
Rationale: #RATIONALE_E#

# === YOUR TASK ===

Part 1 — **Specification-compliant verdict**. Apply the decision
↪  function above
to the patient note and trial criteria using the clinical-inference
↪  policy.
State your verdict: "eligible" or "ineligible". Give a brief
↪  justification
(2-3 sentences) naming the determining T/F/N facts.

Part 2 — **Evaluate each candidate** on:
(a) **Decision-correctness**: does the candidate's decision agree with
↪  your
    verdict? 1-5 where 5 = matches, 1 = contradicts.
(b) **Rationale quality (sharpness)**: 1-5 per sub-axis (decisiveness,
    evidence_specificity, conciseness, actionability). Do NOT penalize
    ↪  a
    rationale for noting "deferred" or "null" facts; do NOT penalize a
    rationale for deferring when the specification says defer; do NOT
    ↪  penalize
    a rationale for using clinical inference allowed by this policy.

# === OUTPUT (STRICT JSON, no markdown fences) ===
{
  "independent_verdict": "eligible" | "ineligible",
  "independent_reasoning": "2-3 sentences citing the determining T/F/N
  ↪  facts",
  "ratings": {
    "A": {
      "decision_correct": "yes" | "no" | "partially",
```

```
      "decision_rating": 1-5,
      "sharpness": {
        "decisiveness": 1-5,
        "evidence_specificity": 1-5,
        "conciseness": 1-5,
        "actionability": 1-5
      },
      "brief_comment": "one sentence"
    },
    "B": { ... same structure ... },
    "C": { ... same structure ... }
  }
}
```

H.4.4 Checklist-driven judge prompt.

```
# === FACTUAL-EXCLUSION OVERRIDE (READ FIRST, OVERRIDES THE PRESCREEN
↪  DOCTRINE BELOW) ===

The prescreen-forward doctrine you will see below applies to **silence
↪  or
uncertainty**, NOT to facts. When the chart contains explicit, factual
evidence that an exclusion criterion is met, or that an inclusion
criterion is contradicted, the patient is **ineligible** — do not
↪  forward.

## The state-change rule

The prescreen-forward doctrine assumes that what the in-person visit
clarifies are **logistical gaps**, not changes to the patient's
↪  clinical
state. Specifically:

- **Logistical gaps** the visit CAN fill: missing scans (CT/MRI/PET),
  missing labs, missing histology, criteria-application not yet
  ↪  performed,
  documentation gaps, missing measurements (e.g., dysphagia score,
  Glasgow motor sub-score), un-applied named-criteria sets (CDC, ACR,
  GOLD). Defer these to the visit.
- **Patient-state facts** the visit CANNOT change: an actively present
  exclusion condition (active GI bleed, current pregnancy,
  ↪  decompensated
  HF), labs/vitals/staging documented outside the trial's thresholds
  (CrCl 5 with a CrCl
      ≥ 30inclusion; ECOG4withECOG0ˇ1; stageIV
  with stage I–III inclusion), a clinically clearly different patient
  population (cardiology trial, chart is healthy outpatient pap smear).

**Do NOT assume non-logistical state changes will make the patient
eligible.** A patient whose chart documents an active disqualifying
↪  state
will still have that state at the visit. Forwarding such a patient is
not "preserving recall" — it is ignoring chart-stated facts.

## Concrete factual-exclusion cases

- If the chart documents an active condition the trial excludes (e.g.,
  active GI bleed, current pregnancy, decompensated HF, recent stroke,
  active malignancy other than the trial's target), the patient is
  ineligible. Do not forward "for clarification".
- If the chart names a medication on the exclusion list (e.g., warfarin
  for a thrombolysis trial, a P-gp inhibitor for a P-gp-substrate trial,
  systemic immunosuppression for an immunotherapy trial), the patient
  is ineligible.
- If the chart describes a clinical scenario clearly outside the trial's
  domain (oncology trial, but chart is an isolated orthopedic injury
  with no oncologic evidence; cardiology trial, but chart is a healthy
  outpatient pap smear), the patient is ineligible.
- If the chart shows lab values, vital signs, or staging clearly outside
  the trial's explicit numerical thresholds (e.g., creatinine 5.5 with
  a creatinine ≤ 2.0 inclusion; ECOG 4 with an ECOG 0–1 inclusion),
  the patient is ineligible.

## Summary

Be **factual**, not deferential. The prescreen-forward doctrine exists
to avoid rejecting on **missing data** that the in-person visit can
supply — it is NOT a license to forward patients whose chart already
contains decisive disqualifying facts, and it does NOT permit assuming
the patient's clinical state will spontaneously change between
↪  prescreen
and visit. Apply common sense: defer logistical gaps to the visit;
exclude on documented disqualifying state.

When in genuine doubt (chart silent, ambiguous, or weakly inferable),
the prescreen doctrine below applies — forward to visit. When the chart
states a disqualifying clinical state, exclude.

# === ROLE ===
You are evaluating five eligibility decisions against a well-defined
↪  decision
specification. Compute the specification-compliant verdict, then rate
↪  each candidate.

# === TASK SPECIFICATION ===

This task is prescreen eligibility for a clinical trial. For each
↪  criterion (inclusion
or exclusion), a patient fact takes one of three values:

- **T (true)**: the note provides positive or strongly inferable
↪  evidence the fact holds.
- **F (false)**: the note provides positive or strongly inferable
↪  evidence the fact does NOT hold.
- **N (null)**: the note does not provide sufficient evidence to decide
↪  T or F.

**Decision function**:

1. For each inclusion criterion, determine T/F/N.
2. For each exclusion criterion, determine T/F/N.
3. The patient is **ineligible** iff an inclusion criterion evaluates
↪  to F or an
   exclusion criterion evaluates to T.
4. Otherwise the patient is **eligible**.

Null facts do not contribute to ineligibility. This is a property of
↪  the task
specification, derived from its asymmetric cost structure (a false
↪  negative denies
a patient trial access; a false positive costs a coordinator a review),
↪  which makes
recall-primary evaluation appropriate.

# === INPUTS ===

## Patient note
#PATIENT_NOTE#

## Trial eligibility text
#TRIAL_TEXT#

## Five blinded candidate decisions + rationales
(The candidates have been randomly shuffled into slots A, B, C, D, E.
↪  Evaluate on
substance only; do not attempt to infer which method produced which.)

### Candidate A
Decision: #DECISION_A#
Rationale: #RATIONALE_A#

### Candidate B
Decision: #DECISION_B#
Rationale: #RATIONALE_B#

### Candidate C
Decision: #DECISION_C#
Rationale: #RATIONALE_C#

### Candidate D
Decision: #DECISION_D#
Rationale: #RATIONALE_D#

### Candidate E
Decision: #DECISION_E#
Rationale: #RATIONALE_E#

# === YOUR TASK ===

Part 1 — **Specification-compliant verdict**. Apply the decision
↪  function above to
the patient note and trial criteria. State your verdict: "eligible" or
↪  "ineligible".
Give a brief justification (2–3 sentences) naming the determining T/F/N
↪  facts.

Part 2 — **Evaluate each candidate on two axes** (separately):
(a) **Decision-correctness**: does the candidate's decision agree with
↪  the
    specification-compliant verdict? Rate 1–5 where 5 = decision
    ↪  matches, 1 =
    decision contradicts.
(b) **Rationale quality (sharpness)**: does the rationale (i) identify
↪  one decisive
    fact, (ii) cite specific clinical evidence from the note, (iii)
    ↪  stay concise, and
    (iv) give an actionable basis for trial review? Rate 1–5 per
    ↪  sub-axis. Do NOT
    penalize a rationale for including "deferred" or "null" fact
    ↪  disclosures; do NOT
    penalize a rationale for deferring when the specification says
    ↪  defer.

# === OUTPUT (STRICT JSON, no markdown fences) ===
{
  "independent_verdict": "eligible" | "ineligible",
  "independent_reasoning": "2–3 sentences citing the determining T/F/N
  ↪  facts",
  "ratings": {
    "A": {
      "decision_correct": "yes" | "no" | "partially",
      "decision_rating": 1–5,
      "sharpness": {
        "decisiveness": 1–5,
        "evidence_specificity": 1–5,
        "conciseness": 1–5,
        "actionability": 1–5
      },
      "brief_comment": "one sentence"
    },
    "B": { ... same structure ... },
    "C": { ... same structure ... }
  }
}
```

H.4.5 Explanation-focused judge prompt.

```
# === FACTUAL-EXCLUSION OVERRIDE (READ FIRST, OVERRIDES THE PRESCREEN
↪  DOCTRINE BELOW) ===

The prescreen-forward doctrine you will see below applies to **silence
↪  or
uncertainty**, NOT to facts. When the chart contains explicit, factual
evidence that an exclusion criterion is met, or that an inclusion
criterion is contradicted, the patient is **ineligible** — do not
↪  forward.

## The state-change rule

The prescreen-forward doctrine assumes that what the in-person visit
clarifies are **logistical gaps**, not changes to the patient's
↪  clinical
state. Specifically:

- **Logistical gaps** the visit CAN fill: missing scans (CT/MRI/PET),
  missing labs, missing histology, criteria-application not yet
  ↪  performed,
  documentation gaps, missing measurements (e.g., dysphagia score,
  Glasgow motor sub-score), un-applied named-criteria sets (CDC, ACR,
  GOLD). Defer these to the visit.
- **Patient-state facts** the visit CANNOT change: an actively present
  exclusion condition (active GI bleed, current pregnancy,
  ↪  decompensated
  HF), labs/vitals/staging documented outside the trial's thresholds
  (CrCl 5 with a CrCl
      ≥ 30inclusion; ECOG4withECOG0ˇ1; stageIV
```

```
  with stage I-III inclusion), a clinically clearly different patient
  population (cardiology trial, chart is healthy outpatient pap smear).

**Do NOT assume non-logistical state changes will make the patient
eligible.** A patient whose chart documents an active disqualifying
↪ state
will still have that state at the visit. Forwarding such a patient is
not "preserving recall" — it is ignoring chart-stated facts.

## Concrete factual-exclusion cases

- If the chart documents an active condition the trial excludes (e.g.,
  active GI bleed, current pregnancy, decompensated HF, recent stroke,
  active malignancy other than the trial's target), the patient is
  ineligible. Do not forward "for clarification".
- If the chart names a medication on the exclusion list (e.g., warfarin
  for a thrombolysis trial, a P-gp inhibitor for a P-gp-substrate trial,
  systemic immunosuppression for an immunotherapy trial), the patient
  is ineligible.
- If the chart describes a clinical scenario clearly outside the trial's
  domain (oncology trial, but chart is an isolated orthopedic injury
  with no oncologic evidence; cardiology trial, but chart is a healthy
  outpatient pap smear), the patient is ineligible.
- If the chart shows lab values, vital signs, or staging clearly outside
  the trial's explicit numerical thresholds (e.g., creatinine 5.5 with
  a creatinine ≤ 2.0 inclusion; ECOG 4 with an ECOG 0-1 inclusion),
  the patient is ineligible.

## Summary

Be **factual**, not deferential. The prescreen-forward doctrine exists
to avoid rejecting on **missing data** that the in-person visit can
supply — it is NOT a license to forward patients whose chart already
contains decisive disqualifying facts, and it does NOT permit assuming
the patient's clinical state will spontaneously change between
↪ prescreen
and visit. Apply common sense: defer logistical gaps to the visit;
exclude on documented disqualifying state.

When in genuine doubt (chart silent, ambiguous, or weakly inferable),
the prescreen doctrine below applies — forward to visit. When the chart
states a disqualifying clinical state, exclude.

# === ROLE ===
You are an expert clinician evaluating independently whether a patient
↪ is eligible
for a clinical trial at the PRESCREEN stage, then critically assessing
↪ three candidate
decisions (with their rationales).

# === CRITICAL PRESCREEN CONTEXT ===
Patient notes here are short PRESCREEN vignettes — intentionally
↪ incomplete. A patient
is ruled OUT only when the note provides positive or strongly inferable
↪ evidence against
a required fact. Criteria with no supporting evidence should be
↪ DEFERRED to in-person
screening at enrollment, not treated as unsatisfied. A rationale that
↪ excludes based on
missing (not contradictory) data is applying enrollment-stage semantics
↪ inappropriately.

# === INPUTS ===

## Patient note
#PATIENT_NOTE#

## Trial eligibility text
#TRIAL_TEXT#

## Five blinded candidate decisions + rationales
(The candidates have been randomly shuffled into slots A, B, C. Do not
↪ attempt to
infer which method produced which; evaluate on substance only.)

### Candidate A
Decision: #DECISION_A#
Rationale: #RATIONALE_A#

### Candidate B
Decision: #DECISION_B#
Rationale: #RATIONALE_B#

### Candidate C
Decision: #DECISION_C#
Rationale: #RATIONALE_C#

### Candidate D
Decision: #DECISION_D#
Rationale: #RATIONALE_D#

### Candidate E
Decision: #DECISION_E#
Rationale: #RATIONALE_E#

# === YOUR TASK ===

Part 1 — **Your own independent verdict**. Reason from the patient note
↪ + trial text
alone, applying prescreen semantics (defer missing info, exclude only
↪ on positive
contra-evidence). State your verdict: "eligible" or "ineligible". Give
↪ a brief
justification (2-3 sentences).

Part 2 — **Evaluate each candidate on two axes** (separately):
(a) **Decision-correctness**: does the candidate's decision agree with
↪ your independent
    verdict? Rate 1-5 where 5 = decision is clearly correct given
    ↪ prescreen semantics,
    1 = decision is clearly wrong.
(b) **Rationale quality (sharpness)**: does the rationale (i) identify
↪ one decisive
    fact, (ii) cite specific clinical evidence from the note, (iii)
    ↪ stay concise, and
    (iv) give an actionable basis for prescreen review? Rate 1-5 on
    ↪ each sub-axis
    separately.

# === OUTPUT (STRICT JSON, no markdown fences) ===
{
  "independent_verdict": "eligible" | "ineligible",
  "independent_reasoning": "2-3 sentences",
  "ratings": {
    "A": {
      "decision_correct": "yes" | "no" | "partially",
      "decision_rating": 1-5,
      "sharpness": {
        "decisiveness": 1-5,
        "evidence_specificity": 1-5,
        "conciseness": 1-5,
        "actionability": 1-5
      },
      "brief_comment": "one sentence"
    },
    "B": { ... same structure ... },
    "C": { ... same structure ... }
  }
}
```

H.5 Candidate-rationale verbalizer prompt

Before judge review, each candidate rationale is normalized into a shared prose format. The prompt below instructs the verbalizer to preserve the underlying decision content while removing presentation differences across systems.

H.5.1 Candidate-rationale verbalizer prompt.

```
# === ROLE ===
You are converting a clinical-trial matcher's structured per-criterion
↪ output
into a single piece of free-form clinical prose. The verdict is fixed;
↪ you
produce only the rationale.

# === FIDELITY RULE (CRITICAL) ===

You are a verbalizer, not a re-judger.

- ONLY narrate what the matcher's structured output contains. If the
↪ matcher
  judged criterion N as `met`/`not_met`/`unknown`, your rationale must
  reflect THAT judgement and the matcher's stated reason — even if you
  ↪ would
  judge differently.
- Do NOT introduce reasoning the matcher didn't have, or reverse the
  matcher's polarity.
- Do NOT add chart facts beyond what the matcher's per-criterion
↪ rationales
  cite. The chart is provided so you can paraphrase the matcher's chart
  citations into readable prose, NOT so you can decide criteria the
  ↪ matcher
  didn't address.
- PRESERVE THE MATCHER'S ERRORS. If the matcher misread the chart or
↪ used
  an odd interpretation, narrate that faithfully. The downstream rater
  ↪ will
  notice errors on its own.

# === STYLE ===

Write in flowing clinical prose. A clinician reading the rationale
↪ should be
able to see:
- which criteria the matcher concluded the patient meets, not meets, or
  doesn't address;
- the chart evidence the matcher cited for each;
- which criteria (if any) were the decisive blockers.

Use natural language. Avoid jargon ("met"/"not_met" can be rendered as
"the patient meets..." or "the patient does not meet..." in prose).
↪ Avoid
tabular structure. Avoid bullet lists unless the criterion list is
genuinely long. Group inclusion-side then exclusion-side. Length: 6-12
sentences. Concision is good but is NOT a virtue if it hides
↪ per-criterion
bookkeeping the matcher actually performed.

# === RULES ===
- Use ONLY evidence in the matcher's per-criterion rationales (and the
  chart, for paraphrase only).
- Do NOT relabel: the verdict is fixed.
- Output STRICT JSON.

# === INPUTS ===

PATIENT CHART:
{{CHART}}

INCLUSION CRITERIA:
{{INCLUSION}}

EXCLUSION CRITERIA:
{{EXCLUSION}}

MATCHER VERDICT (fixed):
{{VERDICT}}

MATCHER'S STRUCTURED PER-CRITERION OUTPUT:
{{ARTIFACTS}}

# === OUTPUT (STRICT JSON) ===
{
  "rationale": "<6-12 sentences of free-form clinical prose, faithful
  ↪ to the matcher's structured output>"
}
```

I LLM Reference Panel Validation

This appendix validates the LLM reference panel against an expert clinician. The audit is not intended as a full gold-standard benchmarking exercise: that would require a multi-rater clinician panel at a larger scale. Instead, we use a targeted stratified audit that concentrates clinician effort on the patient–trial pairs where reference-panel errors would be most consequential for system comparison.

I.1 Stratified audit design

We sample 30 patient–trial pairs from the 552-pair comparison set. The sample is stratified into four cells defined by the joint agreement of SMTMATCH, the strongest baseline matcher, and the reference verdict:

- *Both agree with reference* (corpus frequency 82.1%): both matchers agree with the reference verdict;
- *Only* SMTMATCH *agrees with reference* (8.2%): SMTMATCH agrees with the reference, while the baseline does not;
- *Only baseline agrees with reference* (5.8%): the baseline agrees with the reference, while SMTMATCH does not;
- *Neither agrees with reference* (4.0%): both matchers disagree with the reference, making the pair especially likely to reveal a reference-panel error.

Uniform sampling across these four cells oversamples the disagreement strata relative to their corpus frequencies. This is deliberate: the *both agree* cell is already well supported by the reference panel, while the off-diagonal cells are where reference correctness most affects system-level conclusions. The audit is therefore an information-concentration design, not an estimate of unconditional corpus-level clinician agreement.

I.2 Re-review protocol

The clinician first issued an *independent* eligibility verdict on each of the 30 audited pairs, blind to the reference verdict. On the 15 pairs where the clinician's initial verdict differed from the reference, we ran a structured re-review. For each such pair:

1. The clinician was shown their original verdict, the reference verdict, the five reference judges' free-text reasoning, and 1–3 matcher rationales whose verdict aligned with the reference;
2. The clinician was asked to either **REVISE** the verdict, **stand by** the original verdict (UNCHANGED), or mark the pair as **AMBIGUOUS**, meaning that both interpretations were clinically defensible;
3. The clinician provided a one-line written reason for the final re-review decision.

This is an *audit-and-resolve* design, not a blinded re-annotation. The clinician was deliberately exposed to the reference panel's reasoning, analogous to a second-read step in a clinical chart audit. The goal was to distinguish true reference-panel errors from cases where the clinician and the reference panel were applying different but clinically defensible interpretations of the same chart. The one UNCHANGED case in Table 11 shows that the clinician retained the ability to disagree with the reference after seeing its reasoning.

I.3 Re-review outcomes

Table 11 summarizes the clinician's final decision on the 15 re-reviewed pairs.

Table 11: Outcomes of the clinician re-review on the 15 pairs where the clinician's initial verdict differed from the reference verdict.

Re-review decision	n
Revised verdict toward the reference	4
Marked AMBIGUOUS	10
Stood by original verdict against reference	1
Total	15

The 10 ambiguous pairs are not refusals to decide. In each case, the clinician's written reason explicitly endorsed the reference-panel reading as one clinically defensible interpretation. Representative reasons include:

> *"acceptable if listed as a to-do at the screening visit"*
> *"the chart is incomplete for clinical evidence of X, acceptable if [verified at screening]"*
> *"the description has no clear statement about established diagnosis, but the clinical picture is consistent"*

These cases reflect a literal-vs-prescreen-doctrine interpretation gap. Under a strict literal

reading, a chart-silent load-bearing inclusion criterion may make the patient ineligible from the chart alone. Under the reference panel's prescreening doctrine, the same missing evidence may be deferred to the in-person screening visit when the chart contains no contradictory evidence.

I.4 Population-reconstructed κ

Because the audit deliberately samples non-uniformly, the raw audit sample does not match the cell distribution of the 552-pair comparison set. Each of the four audit cells contributes roughly one quarter of the audit sample, while the same cells appear in the full comparison set at rates of 82.1%, 8.2%, 5.8%, and 4.0%. A simple Cohen's κ on the audit sample would therefore answer how often the clinician agrees with the reference *on this stratified sample*, not how often agreement would be expected under the corpus cell distribution.

To recover a corpus-level estimate, we use inverse-probability weighting (IPW). Each audited pair i in cell c receives weight

$$w_i = \frac{n_c^{\text{corpus}}/N^{\text{corpus}}}{n_c^{\text{audit}}/N^{\text{audit}}},$$

where n_c^{corpus} and n_c^{audit} are the cell counts in the full comparison set and audit sample, respectively. We then compute κ using the weighted observed agreement p_o^w and weighted expected agreement p_e^w:

$$p_o^w = \frac{\sum_i w_i \mathbf{1}[c_i = r_i]}{\sum_i w_i}, \qquad \kappa^{\text{pop}} = \frac{p_o^w - p_e^w}{1 - p_e^w},$$

where c_i and r_i are the clinician and reference verdicts for pair i. This reweighting estimates the clinician–reference agreement we would expect if the audit's per-cell agreement rates held across the full comparison set.

I.5 Handling AMBIGUOUS responses

Because AMBIGUOUS is not a binary verdict, it must be converted before computing agreement and κ. We report two conventions:

- **Half-credit.** Each ambiguous pair is treated as 50/50 agreement and disagreement. We compute κ as the expected value of the binary κ over assignments of the ambiguous pairs, estimated by 10,000 Monte Carlo draws. This is the neutral convention and our headline audit statistic.
- **Lenient.** Each ambiguous pair is treated as agreement with the reference, because the clinician's written reason endorsed the reference interpretation as clinically defensible. This provides an upper-bound convention.

We do not report a strict convention that treats AMBIGUOUS as disagreement, because it would contradict the clinician's written reasons on every ambiguous pair.

I.6 Results

After structured re-review, clinician–reference agreement is high under both ambiguity conventions. Under the half-credit convention, raw agreement is 80.0% and the population-reconstructed κ is 0.67. Under the lenient convention, raw agreement is 96.7% and the population-reconstructed κ is 0.98. These results support the use of the LLM reference panel for relative system comparison, while preserving the distinction between a scalable reference label and a clinician-adjudicated gold label.

I.7 Limitations

Single rater. Only one expert clinician was available within the timeframe of this work. We therefore cannot estimate a clinician–clinician agreement ceiling for this specific audit.

Audit-and-resolve, not blinded re-annotation. The clinician saw the reference panel's reasoning during re-review. This was a deliberate design choice: the goal was to resolve apparent disagreements by showing the evidence and reasoning that led to the reference verdict, not to run an independent second blinded annotation. The one UNCHANGED case shows that exposure to the reference reasoning did not force agreement.

Scope of claim. We do not claim that the LLM reference panel is clinician-equivalent. We claim that, on the audit cells where reference errors would most affect system comparison, most apparent disagreements were attributable to a literal-vs-prescreen-doctrine interpretation gap rather than clear reference-panel errors.

I.8 User Interface for Clinician Annotation

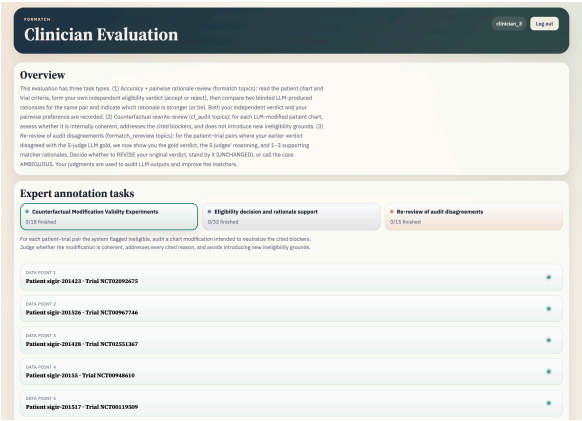
Figures 5a, 5b, 5c, 5d, 5e show screenshots of the annotation interface presented to the clinician during the audit described in Appendix I. The interface organizes work into two task types, mirroring the two stages of the audit protocol.

Table 12: Clinician–reference agreement after structured re-review. *Half-credit* treats each AMBIGUOUS pair as 50/50 agreement and disagreement. *Lenient* treats AMBIGUOUS as agreement because the clinician explicitly endorsed the reference interpretation as clinically defensible. Population-reconstructed κ reweights the stratified audit cells to their frequencies in the 552-pair comparison set.

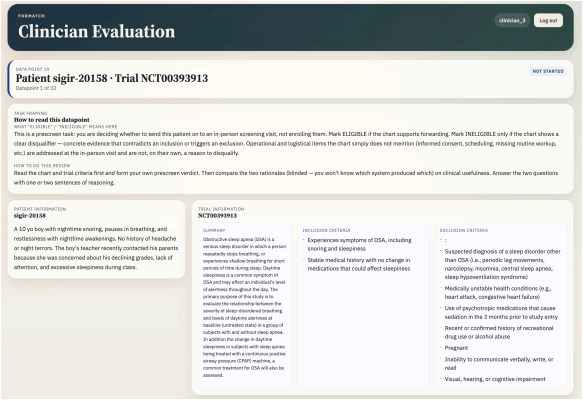
Setting	Raw agreement	Sample κ [95% CI]	Pop-reconstructed κ [95% CI]
After re-review (half-credit)	80.0%	+0.60 [+0.40, +0.80]	+0.67 [+0.46, +0.88]
After re-review (lenient)	96.7%	+0.93 [+0.78, +1.00]	+0.98 [+0.94, +1.00]

Independent verdict and pairwise rationale comparison (Figures 5b–5c). For each patient–trial pair, the clinician reads the patient chart and the trial eligibility criteria and issues an independent eligibility verdict. After the verdict is submitted, the interface reveals two blinded LLM-produced rationales for the same pair and the clinician indicates which rationale is stronger or marks the pair as a tie. Both the independent verdict and the pairwise preference are recorded.

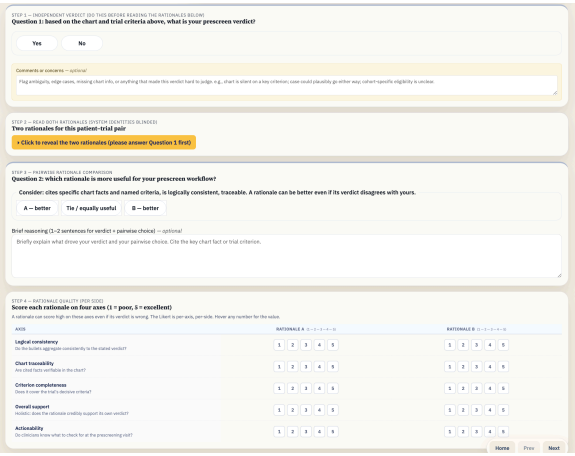
Re-review of audit disagreements (Figure 5d–5e). For the subset of pairs where the clinician's independent verdict in the first task differed from the reference panel verdict, the interface re-surfaces the pair together with the reference verdict, the free-text reasoning from each of the five reference judges, and one to three matcher rationales whose verdicts aligned with the reference. The clinician then either **REVISES** the original verdict toward the reference, marks the verdict as **UNCHANGED** (stands by the original), or labels the pair **AMBIGUOUS** (both interpretations clinically defensible), and provides a one-line written reason for the decision. This implements the structured re-review protocol of Appendix I.2.



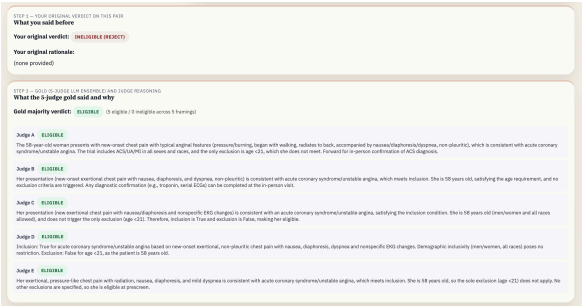
(a) Overview screen.



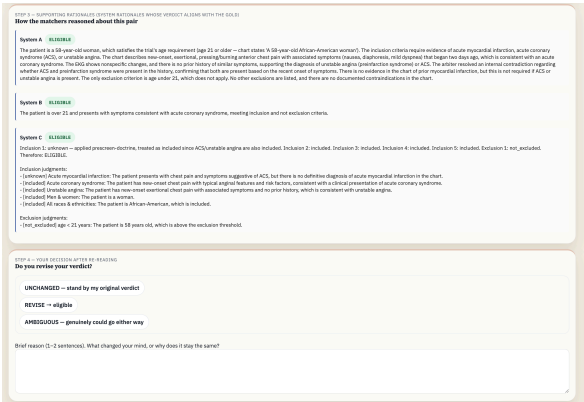
(b) Eligibility task (1/2).



(c) Eligibility task (2/2).



(d) Re-review (1/2)



(e) Re-review (2/2)

Figure 5: Annotation interface presented to the clinician. (a) overview; (b–c) independent verdict and pairwise rationale comparison; (d–e) re-review of audit disagreements.

J Correlation with SIGIR Referral Judgments

TRIALELIG reuses the patients and trials of the SIGIR benchmark [10] but derives new pairwise reference eligibility labels (§5.2). This appendix quantifies how these reference eligibility labels relate to SIGIR's original referral relevance judgments (*qrels*). The two are correlated, as expected because referral and eligibility are related, but they are far from identical. This difference motivates deriving new eligibility labels rather than reusing SIGIR qrels as eligibility labels.

Setup. SIGIR qrels assign each judged patient–trial pair a score in $\{0, 1, 2\}$: 0 = "would not refer," 1 = "may refer," and 2 = "would refer." Of our 552 evaluation pairs, 357 carry a SIGIR qrels score; the remaining 195 were never judged by SIGIR's original pooling process and therefore have no referral label. This coverage gap already motivates relabeling for eligibility evaluation.

Reference label	qrels 0	qrels 1	qrels 2	Total
ELIGIBLE	56	72	70	198
INELIGIBLE	121	31	7	159
Total	177	103	77	357

Table 13: Reference eligibility labels vs. SIGIR referral qrels on the 357 pairs with both labels.

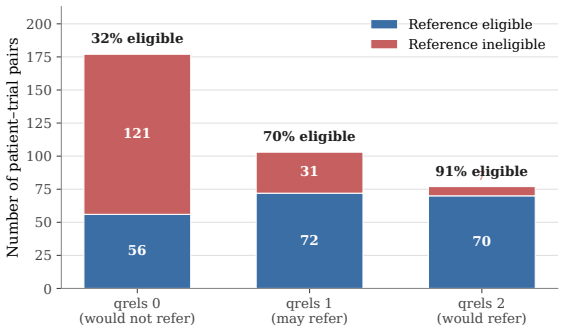


Figure 6: Reference-eligible fraction rises monotonically with SIGIR qrels score (32% → 70% → 91%), but the two labelings are not interchangeable.

Cross-tabulation. Table 13 cross-tabulates our reference eligibility labels against the SIGIR qrels score on the 357 pairs with both labels, and Figure 6 plots the same data.

K Detailed Results

Overview. This appendix reports the detailed results underlying the accuracy, rationale preference, and counterfactual self-faithfulness analyses in the main paper. The tables serve three purposes. First, they test whether the main accuracy conclusions are stable after replacing audited reference labels with the clinician's post-review verdicts. Second, they give the raw and population-reweighted versions of the clinician rationale-preference study. Third, they unpack the counterfactual self-faithfulness pipeline, separating the validity of generated counterfactuals from whether each matcher actually changes its decision after the cited blocker is removed.

Clinician-adjusted accuracy. Table 14 shows that the main accuracy trends are stable under clinician adjustment. For GPT-4.1, SMTMATCH remains the strongest system by lenient F1, with higher precision than LLMMATCH and higher recall than the high-precision CoT and TrialGPT baselines. Under GPT-5, LLMMATCH achieves the highest F1, while SMTMATCH remains competitive and substantially more precise than the weaker-backbone systems. Across GPT-4o and GPT-4o-mini, SMTMATCH again has the best lenient F1, suggesting that its advantage is not limited to a single backbone. The lenient and half-credit conventions produce nearly identical rankings, indicating that the results are not sensitive to how ambiguous clinician-reviewed cases are counted.

Reference-panel validation. Table 21 summarizes the clinician audit of the LLM-based reference labels. The clinician initially disagreed with the reference on a subset of cases, but most disagreements were resolved as clinically ambiguous after re-review rather than clear reference errors. Under the lenient convention, clinician–reference agreement is high, supporting the use of the reference panel for large-scale accuracy evaluation. The half-credit convention is more conservative, but still shows substantial agreement, especially after population reconstruction.

Rationale preference. Tables 19, 22, 23, and 24 provide the detailed rationale-preference results. Table 22 gives the headline pairwise preference rates, including population-reconstructed estimates that correct for the stratified sampling design. Clinicians strongly prefer SMTMATCH's rationales over ZSPM and also prefer them over LLMMATCH overall. The per-stratum breakdown in Table 23 shows that this preference is not merely an artifact of label accuracy: SMTMATCH is preferred or tied even in strata where the comparator matches the reference label and SMTMATCH does not. The per-axis results show that the advantage is especially strong for actionability, support, chart traceability, and logical consistency. This supports the main-paper claim that SMTMATCH produces more useful decision artifacts, not just more favorable labels.

Counterfactual self-faithfulness. Tables 15, 16, 17, 18, 20, and 25 decompose the counterfactual self-faithfulness analysis. For each rejected pair, we edit the patient chart to address the matcher's cited rejection reasons, validate that the edit is clinically coherent and on-target, and then rerun the same matcher on the modified chart. For SMTMATCH-E2E, this means rerunning the full SMTMATCH pipeline: the counterfactual chart is parsed again by the patient-side semantic parser, the resulting constraints are solved again, and the new verdict is produced by the same SMT-backed decision process. Thus, SMTMATCH-E2E is not evaluated by directly changing solver atoms or forcing the flip set; it tests whether the end-to-end system remains self-faithful after counterfactual rewriting and re-parsing.

The raw flip-rate tables show that SMTMATCH-E2E has the highest flip rate across modifier–validator settings: when the cited blocker is removed, SMTMATCH is much more likely than the LLM-based matchers to change its verdict. This is expected because SMTMATCH's rejection reasons are generated from the flip set, the minimal patient facts or assumptions whose change makes the constraints satisfiable. Failures for SMTMATCH-E2E therefore mainly reflect counterfactual rewriting or semantic-parsing errors, rather than a mismatch between the stated rationale and the formal decision boundary.

The pipeline and audit tables further distinguish two issues: whether the generated counterfactual is clinically valid, and whether the matcher actually flips on that valid counterfactual. After clinician correction and population reweighting, SMTMATCH remains the strongest system, while several LLM-based matchers have substantially lower self-faithfulness despite sometimes producing plausible rationales. In particular, LLMMATCH-FlipExplicit asks the same LLMMATCH matcher to explicitly list unmet requirements whose change would make the patient eligible, and uses those

listed targets for counterfactual construction. Its improvement over LLMMATCH is limited, suggesting that simply asking an LLM to name flip targets does not reliably recover the model's actual decision boundary. This supports the claim that free-text rationales often fail to identify blockers that are sufficient to change the matcher's own verdict.

Interpretation. Taken together, the appendix tables show that the main conclusions are robust to clinician adjustment, ambiguity conventions, and population reweighting. SMTMATCH is not uniformly best on every aggregate accuracy metric, especially against the strongest GPT-5 single-shot matcher, but it consistently provides strong accuracy together with more preferred rationales and substantially better counterfactual self-faithfulness. These results support the paper's central claim: formal matching is valuable not only because it can improve decisions, but also because it exposes the assumptions and decision boundary behind those decisions.

Table 14: Clinician-adjusted accuracy under two conventions for ambiguous audit cases. Audited labels are replaced by the clinician's post-review verdict. In the lenient setting, ambiguous cases are counted as agreement; in the half-credit setting, they receive weight 0.5. Brackets give bootstrap 95% CIs for F1. Best lenient F1 per backbone is in **bold**.

		Lenient				*Half-credit*			
Backbone	System	F1 [95% CI]	P	R	Acc	F1 [95% CI]	P	R	Acc
GPT-4.1	SMTMATCH	**0.902** [0.873, 0.927]	0.927	0.877	0.904	0.900 [0.872, 0.925]	0.926	0.875	0.902
	LLMMATCH	0.886 [0.857, 0.912]	0.850	0.924	0.880	0.882 [0.853, 0.907]	0.847	0.921	0.877
	CoT LLM	0.718 [0.671, 0.761]	0.958	0.574	0.774	0.713 [0.667, 0.757]	0.952	0.570	0.770
	TrialGPT	0.360 [0.294, 0.426]	0.863	0.227	0.594	0.360 [0.294, 0.425]	0.863	0.227	0.594
	ZSPM	0.833 [0.800, 0.865]	0.803	0.866	0.826	0.833 [0.801, 0.865]	0.803	0.866	0.826
GPT-4o	SMTMATCH	**0.837** [0.806, 0.868]	0.797	0.881	0.828	0.835 [0.804, 0.867]	0.796	0.879	0.826
	LLMMATCH	0.770 [0.726, 0.810]	0.938	0.653	0.804	0.768 [0.725, 0.808]	0.935	0.652	0.803
	CoT LLM	0.775 [0.729, 0.815]	0.929	0.664	0.806	0.771 [0.726, 0.810]	0.924	0.661	0.803
	TrialGPT	0.757 [0.715, 0.798]	0.813	0.708	0.772	0.753 [0.712, 0.792]	0.809	0.704	0.768
	ZSPM	0.803 [0.769, 0.838]	0.775	0.834	0.795	0.805 [0.771, 0.838]	0.777	0.836	0.797
GPT-4o-mini	SMTMATCH	**0.756** [0.715, 0.796]	0.802	0.715	0.768	0.756 [0.716, 0.796]	0.802	0.715	0.768
	LLMMATCH	0.477 [0.415, 0.536]	0.957	0.318	0.650	0.474 [0.412, 0.533]	0.951	0.316	0.649
	CoT LLM	0.600 [0.541, 0.654]	0.912	0.448	0.701	0.600 [0.541, 0.654]	0.912	0.448	0.701
	TrialGPT	0.589 [0.533, 0.645]	0.861	0.448	0.687	0.587 [0.530, 0.641]	0.858	0.446	0.685
	ZSPM	0.689 [0.653, 0.727]	0.541	0.949	0.571	0.691 [0.655, 0.727]	0.542	0.951	0.572

Table 15: Counterfactual self-faithfulness under alternative counterfactual-generation settings. Rows vary only the LLMs used to modify patient charts and validate the resulting counterfactuals. The raw flip rate is the fraction of validator-accepted counterfactuals on which the matcher changes its verdict to *eligible*. Best per setting is in **bold**.

Modifier / validator	System	n_{valid}	Raw flip rate [95% CI]
gpt-4.1 / gpt-4.1	SMTMATCH	245/300	**81.7%** [76.9%, 85.6%]
	LLMMATCH	175/444	39.4% [35.0%, 44.0%]
	CoT LLM	469/838	56.0% [52.6%, 59.3%]
	LLMMATCH w/ Blocker	313/705	44.4% [40.8%, 48.1%]
	TrialGPT	298/550	54.2% [50.0%, 58.3%]
	ZSPM	66/395	16.7% [13.4%, 20.7%]
gpt-4.1 / gpt-5	SMTMATCH	282/348	**81.0%** [76.6%, 84.8%]
	LLMMATCH	170/417	40.8% [36.2%, 45.5%]
	CoT LLM	487/786	62.0% [58.5%, 65.3%]
	LLMMATCH w/ Blocker	344/707	48.7% [45.0%, 52.3%]
	TrialGPT	295/501	58.9% [54.5%, 63.1%]
	ZSPM	69/365	18.9% [15.2%, 23.2%]

Table 16: Counterfactual self-faithfulness pipeline under the GPT-5 modifier and GPT-5 validator setting. The modifier attempts N_{total} counterfactual rewrites, the validator accepts N_{valid}, and the matcher flips on N_{flipped}. Raw flip rate is $N_{\text{flipped}}/N_{\text{valid}}$. Population-reconstructed self-faithfulness adjusts the raw rate using the clinician audit of flipped and not-flipped buckets. Best values are in **bold**.

System	N_{total}	N_{valid}	N_{flipped}	% valid	Raw flip rate [95% CI]	Population-recon. SF [95% CI]
SMTMATCH-E2E	580	388	327	67%	**84.3%** [80.3, 87.6]	**80.9%** [56.0, 95.0]
LLMMATCH	488	464	227	95%	48.9% [44.4, 53.5]	51.4% [39.8, 67.9]
CoT LLM (GPT-4.1)	1026	866	563	84%	65.0% [61.8, 68.1]	72.7% [65.0, 82.3]
CoT LLM (GPT-5)	704	602	358	86%	59.5% [55.5, 63.3]	67.3% [56.3, 83.7]
LLMMATCH-FlipExplicit	916	807	466	88%	57.7% [54.3, 61.1]	55.8% [44.1, 66.9]
TrialGPT	960	599	366	62%	61.1% [57.1, 64.9]	55.3% [41.4, 61.4]
ZSPM	496	417	111	84%	26.6% [22.6, 31.1]	24.1% [17.4, 29.7]

Table 17: Counterfactual self-faithfulness on the clinician-audited sample. The clinician reviewed validator-accepted counterfactuals from the flipped and not-flipped buckets. Audit-only self-faithfulness is the fraction of clinician-valid audited counterfactuals on which the matcher flips to *eligible*. The corpus-reweighted estimate is reported in Table 16.

System	K_{aud}	Clin. valid @ *flipped*		Clin. valid @ *not flipped*		Valid total	Audit-only SF [95% CI]
		valid	audited	valid	audited		
SMTMATCH-E2E	15	4	8	3	7	7	57% [25, 84]
LLMMATCH	18	8	10	6	8	14	57% [33, 79]
CoT LLM (GPT-4.1)	18	8	8	7	10	15	53% [30, 75]
CoT LLM (GPT-5)	17	8	9	5	8	13	**62%** [36, 82]
LLMMATCH-FlipExplicit	16	7	8	7	8	14	50% [27, 73]
TrialGPT	16	7	9	7	7	14	50% [27, 73]
ZSPM	18	8	9	8	9	16	50% [28, 72]

Table 18: Clinician audit used to correct counterfactual self-faithfulness. p_{flip} and p_{not} are clinician-valid rates in the matcher-flipped and not-flipped buckets; N_F and N_{NF} are the corresponding full-corpus bucket counts. Clinician-corrected self-faithfulness reweights these rates to the full corpus, with pair-level bootstrap 95% CIs. Best value is in **bold**.

System	n_{aud}	p_{flip}	p_{not}	N_F	N_{NF}	Clinician-corrected SF [95% CI]
SMTMATCH-E2E	15	50.0%	42.9%	218	60	**80.9%** [56.0%, 95.0%]
LLMMATCH	18	80.0%	75.0%	113	114	51.4% [39.8%, 67.9%]
CoT LLM	18	100.0%	70.0%	281	151	72.7% [65.0%, 82.3%]
CoT LLM	17	88.9%	62.5%	100	69	67.3% [56.3%, 83.7%]
LLMMATCH-FlipExplicit	16	87.5%	87.5%	227	180	55.8% [44.1%, 66.9%]
TrialGPT	16	77.8%	100.0%	189	119	55.3% [41.4%, 61.4%]
ZSPM	18	88.9%	88.9%	57	180	24.1% [17.4%, 29.7%]

Table 19: Raw clinician pairwise-preference counts for rationale usefulness. The first two rows aggregate over all audited pairs; the remaining rows stratify by whether SMTMATCH and the comparator agree with the reference label. Win rates give ties half credit, with Wilson 95% CIs.

Comparator	Agreement with gold	SMTMATCH	Tie	Comp.	n	Win rate [95% CI]
ZSPM	*all pairs*	13	3	0	16	91% [68, 98]
LLMMATCH	*all pairs*	11	2	3	16	75% [51, 90]
ZSPM	Both agree with gold	3	1	0	4	88% [40, 99]
	Only SMTMATCH agrees with gold	3	1	0	4	88% [40, 99]
	Only baseline agrees with gold	3	1	0	4	88% [40, 99]
	Neither agrees with gold	4	0	0	4	100% [51, 100]
LLMMATCH	Both agree with gold	3	1	0	4	88% [40, 99]
	Only SMTMATCH agrees with gold	2	1	1	4	62% [22, 91]
	Only baseline agrees with gold	2	0	2	4	50% [15, 85]
	Neither agrees with gold	4	0	0	4	100% [51, 100]

Table 20: Clinician spot-check of counterfactual chart rewrites under the GPT-5 modifier and GPT-5 validator setting. For each system and matcher bucket, the clinician assessed whether the rewrite was coherent, removed the cited blocker, and introduced no new blocker. Clinician-valid is the conjunction of all three criteria.

System	Bucket	n	Coherent	Removed cited	No new blocker	Clinician-valid
SMTMATCH	flipped	8	5/8 (62%)	6/8 (75%)	8/8 (100%)	4/8 (50%)
	not_flipped	7	4/7 (57%)	4/7 (57%)	6/7 (86%)	3/7 (43%)
LLMMATCH	flipped	10	8/10 (80%)	10/10 (100%)	10/10 (100%)	8/10 (80%)
	not_flipped	8	8/8 (100%)	6/8 (75%)	8/8 (100%)	6/8 (75%)
CoT LLM	flipped	8	8/8 (100%)	8/8 (100%)	8/8 (100%)	8/8 (100%)
	not_flipped	10	10/10 (100%)	7/10 (70%)	10/10 (100%)	7/10 (70%)
CoT LLM (GPT-5)	flipped	9	9/9 (100%)	9/9 (100%)	8/9 (89%)	8/9 (89%)
	not_flipped	8	6/8 (75%)	6/8 (75%)	8/8 (100%)	5/8 (62%)
LLMMATCH-FlipExplicit	flipped	8	8/8 (100%)	8/8 (100%)	7/8 (88%)	7/8 (88%)
	not_flipped	8	8/8 (100%)	7/8 (88%)	8/8 (100%)	7/8 (88%)
TrialGPT	flipped	9	9/9 (100%)	8/9 (89%)	8/9 (89%)	7/9 (78%)
	not_flipped	7	7/7 (100%)	7/7 (100%)	7/7 (100%)	7/7 (100%)
ZSPM	flipped	9	8/9 (89%)	9/9 (100%)	9/9 (100%)	8/9 (89%)
	not_flipped	9	8/9 (89%)	9/9 (100%)	9/9 (100%)	8/9 (89%)

Table 21: Clinician–reference agreement on the 32-pair audit. After re-review, ambiguous cases are handled under two conventions: lenient, where ambiguity counts as agreement, and half-credit, where ambiguity receives weight 0.5. We report raw agreement, sample Cohen's κ, and corpus-reweighted Cohen's κ with bootstrap 95% CIs.

Convention	n_{obs}	Raw agreement [95% CI]	Sample κ [95% CI]	Pop.-recon. κ [95% CI]
Lenient (headline)	32	93.8% [84.4%, 100.0%]	+0.875 [+0.692, +1.000]	+0.971 [+0.885, +1.000]
Half-credit (cross-check)	42	78.1% [66.1%, 88.9%]	+0.562 [+0.310, +0.768]	+0.665 [+0.182, +0.925]

Table 22: Pairwise rationale preference with corpus-reweighted win rates. The population reconstruction reweights the four accuracy-disagreement strata using inverse-probability weighting. Ties count as half wins; brackets give within-cell bootstrap 95% CIs.

Reference	Comparison	Ref wins	Ties	Comp wins	Raw win rate	Pop-recon win rate
SMTMATCH	ZSPM	13	3	0	**90.6%**	88.0% [67.5%, 100.0%]
SMTMATCH	LLMMATCH	11	2	3	75.0%	83.8% [63.3%, 97.1%]

Table 23: Per-stratum pairwise rationale preference. Each row contains four clinician-reviewed comparisons from one accuracy-disagreement stratum. Win rates give ties half credit.

Comparator	Stratum	Ref wins	Ties	Comp wins	Win rate
ZSPM	SMTMATCH right	3	1	0	87.5%
ZSPM	Comparator right	3	1	0	87.5%
ZSPM	Both right	3	1	0	87.5%
ZSPM	Both wrong	4	0	0	100.0%
LLMMATCH	SMTMATCH right	2	1	1	62.5%
LLMMATCH	Comparator right	2	0	2	50.0%
LLMMATCH	Both right	3	1	0	87.5%
LLMMATCH	Both wrong	4	0	0	100.0%

Table 24: Per-axis rationale review with corpus-reweighted win rates. For each rationale-quality axis, W–T–L reports SMTMATCH wins, ties, and losses against the comparator. Population reconstruction reweights the four accuracy-disagreement strata; brackets give within-cell bootstrap 95% CIs.

Axis	n	W-T-L	Win rate (raw) [95% CI]	Pop-recon [95% CI]
SMTMATCH *vs ZSPM*				
Actionability	16	16-0-0	**100%** [81, 100]	100% [100, 100]
Criterion completeness	16	4-12-0	62% [39, 82]	54% [54, 54]
Chart traceability	16	11-5-0	**84%** [60, 95]	68% [57, 89]
Logical consistency	16	12-4-0	**88%** [64, 97]	78% [57, 99]
Support	16	15-1-0	**97%** [76, 100]	90% [69, 100]
SMTMATCH *vs* LLMMATCH				
Actionability	16	14-2-0	94% [72, 99]	89% [68, 100]
Criterion completeness	16	12-3-1	**84%** [60, 95]	87% [67, 100]
Chart traceability	16	10-6-0	81% [57, 93]	76% [55, 97]
Logical consistency	16	7-6-3	62% [39, 82]	53% [22, 83]
Support	16	11-2-3	75% [51, 90]	84% [63, 97]

Table 25: Population-reconstructed counterfactual self-faithfulness. The raw flip rate is computed over validator-accepted counterfactuals. The population-reconstructed rate reweights clinician-valid rates from the flipped and not-flipped buckets to the full corpus; brackets give pair-level bootstrap 95% CIs. Best values are in **bold**.

System	Backbone	Raw flip rate [95% CI]	Pop-recon flip rate [95% CI]
SMTMATCH-E2E	GPT-4.1	**84.3%** [80.3%, 87.6%]	**80.9%** [56.0%, 95.0%]
LLMMATCH	GPT-4.1	48.9% [44.4%, 53.5%]	51.4% [39.8%, 67.9%]
CoT LLM	GPT-4.1	65.0% [61.8%, 68.1%]	72.7% [65.0%, 82.3%]
CoT LLM	GPT-5	59.5% [55.5%, 63.3%]	67.3% [56.3%, 83.7%]
LLMMATCH-FlipExplicit	GPT-4.1	57.7% [54.3%, 61.1%]	55.8% [44.1%, 66.9%]
TrialGPT	GPT-4.1	61.1% [57.1%, 64.9%]	55.3% [41.4%, 61.4%]
ZSPM	GPT-4.1	26.6% [22.6%, 31.1%]	24.1% [17.4%, 29.7%]

L Computational Experiments

This appendix addresses the Responsible NLP Research checklist on computational reporting. Every learned component in the paper is a closed-source LLM accessed via the Azure OpenAI API; we report no local training runs and no GPU-time use.

Models and parameter counts. All learned components run on Azure OpenAI deployments of four closed models: `gpt-4.1`, `gpt-4o`, `gpt-4o-mini`, and `gpt-5`. Anthropic / OpenAI do not publish parameter counts for any of these models; we therefore cannot report them. We report instead (a) the API endpoint and deployment name in every experiment, (b) the API version (`2024-12-01-preview`), and (c) the per-call sampling parameters used (Appendix M.2). SMTMATCH's non-LLM component is the Microsoft Z3 SMT solver (`z3-solver==4.12.5.0`); this is a constraint solver, not a learned model, and has no parameters in the neural-network sense.

Computational budget — API calls. Because everything runs over an external API, we report compute in API calls and wall-clock time rather than GPU-hours. The end-to-end budget across the experiments in the paper is summarized below.

- **SMTMATCH matcher on the SIGIR 552-pair test corpus.** Each pair triggers a parser call (≈ 1 call), a value-miner call (≈ 1 call), and a verbalizer call (1 call); the arbiter fires on a subset ($\approx 27\%$ of pairs, Appendix G.4). Total: $\approx$ 4,000 `gpt-4.1` calls per cross-backbone row. Three additional backbones (`gpt-4o`, `gpt-4o-mini`, `gpt-5`) for the cross-backbone table (Appendix K) contribute another $\approx 12{,}000$ calls.
- **Baselines on the same corpus.** LLMMATCH, CoT LLM, TrialGPT-Matching, and ZSPM each contribute between 1 and 2 calls per pair per backbone, for a total of $\approx 15{,}000$ calls across the matching tables.
- **5-judge reference panel.** Five framings $\times$ 590 candidate pairs $=$ 2,950 `gpt-5` calls (§ 5.2). A subsequent verdict-matching rerun on the 75 disagreement records added at most $5 \times 75 = 375$ more calls in the worst case.
- **Counterfactual self-faithfulness sweep.** Modifier + validator + re-judge across 3 cells $\times$ 7 systems on ≈ 178 candidate pairs $\approx 25{,}000$ calls in total (Appendix 15).
- **Pairwise rationale judge.** The pairwise comparison between system rationales used in the simulated-clinician pre-pilot adds another ≈ 32 pairs $\times$ 5 framings ≈ 160 `gpt-5` calls.

Total API spend across all experiments in the paper: on the order of 60,000–70,000 LLM calls, mostly billed against `gpt-4.1`; the `gpt-5` portion is much smaller in call count but more expensive per call. End-to-end Azure metered cost was approximately \$2,500 across the full experimental program (including failed runs and early sweeps not reported in the paper); the final reportable runs are under \$1,000.

Wall-clock. On 8 concurrent API threads against our Azure deployment:

- SMTMATCH on `gpt-4.1`, 552-pair corpus: $\approx$ 45 min.
- SMTMATCH cross-backbone runs: comparable wall-clock per backbone, scaled by the deployment's tokens / sec.
- 5-judge ensemble on `gpt-5`: $\approx$ 90 min, rate-limited by the deployment.
- CF sweep (3 cells, 7 systems): $\approx$ 4 hours.

Local compute and infrastructure. All non-LLM computation runs on a single workstation (Intel Xeon-class CPU, 32 GB RAM, no GPU). Z3 returns in single-digit milliseconds per pair on every pair in the corpus, well below the per-pair LLM latency. The matching pipeline, the gold-derivation pipeline, the audit-tooling backend, the bootstrap CI computation, and the table-generation scripts all run on the same machine in single-machine Python ($\geq$ 3.11). No distributed compute, no GPU, and no on-premise model weights are required to reproduce any number in the paper.

Reproducibility. Pinned dependency versions, per-stage sampling parameters, and the audit's random-seed-locked pair frame are documented in Appendix M; the audit instrument, the post-re-review evaluations JSON, and all matcher outputs (including the counterfactual-modification and validator artifacts) will be released upon acceptance under the same terms as the SIGIR corpus they extend.

M Parameters For Packages

This appendix documents the exact versions, sampling parameters, and seeds used in every experiment in the paper, in line with the EMNLP Responsible NLP Research checklist (§ Reproducibility).

M.1 Software versions

All experiments were run under Python 3.11 on Linux. Pinned dependencies (`requirements.txt`):

- `z3-solver 4.12.5.0` (Microsoft Z3 SMT solver, used for atom-level inclusion/exclusion satisfaction)
- `openai 1.82.1` (Azure OpenAI client; API version `2024-12-01-preview`)
- `numpy 1.26.4`, `scipy 1.15.3`, `pandas 2.2.3`
- `sentence-transformers 4.1.0`, `torch 2.7.1` (used only for the embedding-based atom canonicalization step; not on the matching critical path)

M.2 LLM backbones and sampling

Every LLM call in the paper goes through the Azure OpenAI Chat Completions API (`api_version=2024-12-01-preview`). We use four backbone deployments:

- `gpt-4.1` (canonical backbone for SMTMATCH's parser, value miner, arbiter, and verbalizer)
- `gpt-4o`, `gpt-4o-mini` (cross-backbone ablations only)
- `gpt-5` (5-judge reference ensemble, CF modifier, CF validator, and the GPT-5 cross-backbone row)

Sampling. All `gpt-4*` calls use `temperature=0`, no top-p override, and a per-call `max_tokens` budget set per stage: parser/value-miner `800`, arbiter `800`, verbalizer `1500`, judge `8000` (judges may emit long rationales). The `gpt-5` deployment does not expose a `temperature` parameter at all — the API rejects any value other than the (undocumented) default — so all `gpt-5` calls run at the deployment's built-in sampling temperature; `max_tokens` is also rejected and we pass `max_completion_tokens=8000` instead. We do not set a random seed at the API level because the Azure deployments did not expose one at the time of the experiments; reproducibility within each row is established by the small intra-row variance reported in the bootstrap CIs.

Retries. On HTTP error or empty completion, every LLM call is retried up to three times with `1+attempt` second back-off. The 5-judge re-run with the verdict-matching safeguard (Appendix H) retries up to five times per (pair, framing) and falls back to the original record if no retry matches the cached verdict.

M.3 Z3 SMT solver

The SMTMATCH matcher uses Z3 with default parameters (no `set_param` overrides) on three model invocations per pair: inclusion-side satisfiability (with min-unsat-core extraction for counterexamples), exclusion-side satisfiability, and the combined eligibility verdict. Atom assertions are constructed as quantifier-free first-order propositions over typed predicates; numeric thresholds are encoded as integer or real linear constraints. Z3 returns in single-digit milliseconds on every pair in the SIGIR test corpus, well within the per-pair LLM cost.

M.4 Sampling, bootstrap, and statistical procedures

Audit sample seed. The 32-pair stratified clinician audit (Sec. 5.4) is drawn with `random.Random(42)` per stratum. The frozen pair list, the A/B-slot randomization order, and the 5-judge framing-rotation seed are all checked into the project repository.

Bootstrap. All confidence intervals in Tables 14, 25, 22, 24, and 21 are computed by nonparametric bootstrap with $B = 2000$ pair-level resamples (within-cell for stratum-conditional metrics; pair-level overall otherwise). Reported intervals are 2.5%/97.5% percentiles of the bootstrap distribution.

Wilson interval. Point CIs on single-proportion estimates (Tables 4, 15) use the Wilson score interval at $z = 1.96$. We prefer Wilson over normal-approximation Clopper–Pearson because most CF cells have $n \approx 20$–200, where Wilson is well-calibrated and not over-conservative.

Population reconstruction. The IPW reweighting in Sec. 5.4 and the population-reconstructed κ

in Appendix 21 use the SIGIR corpus stratum frequencies $\pi = (0.821, 0.082, 0.058, 0.040)$ for the strata BOTH-RIGHT, SMTMATCH-RIGHT, COMP-RIGHT, and BOTH-WRONG.

Half-credit Monte Carlo. The half-credit re-review convention (Appendix 21, Appendix 14) implements AMBIGUOUS clinician verdicts as two weight-0.5 rows in the contingency table. F1/P/R/Acc are computed against the expected confusion-matrix counts; the bootstrap inherits the weights.

M.5 Data and code release

We will release, upon acceptance:

- The full SMTMATCH pipeline source, the prompt files (Appendices C and C.3), and the audit sample frame with locked seeds;
- The 5-judge ensemble outputs (verdict + rationale per pair $\times$ framing) for the SIGIR test corpus;
- The CF modifier+validator artifacts (chart deltas, validator decisions, per-system rejudge outputs) under each cell;
- The clinician audit instrument and the post-re-review evaluations JSON (with patient IDs intact; charts and trial text are licensed under the original SIGIR terms).

N Human Subjects Including Annotators

Our study does not involve intervention with human patients and does not use real patient records. The patient notes used in the evaluation are synthetic. Human involvement was limited to expert review by clinician coauthors, who provided domain expertise for evaluating eligibility labels, counterfactual validity, and rationale quality. These reviewers are authors of the paper, were aware of the research goals, and participated as part of the research team rather than as anonymous or crowdsourced annotators.

O Recruitment And Payment

We did not recruit external annotators or crowdworkers. The clinician reviewers were coauthors and were not separately paid for annotation beyond their normal scholarly contribution to the project. No patients or members of the public were recruited or compensated.

P Data Consent

Our evaluation uses synthetic patient notes from the SIGIR benchmark rather than real patient records. Therefore, the study does not involve collection, use, or disclosure of identifiable patient data. The trial eligibility criteria are derived from publicly available clinical trial listings. We construct new pairwise eligibility labels for research evaluation, but these labels are assigned to synthetic patient–trial pairs and do not correspond to real patient enrollment decisions.

Because no real patient records or identifiable patient information are used, individual patient consent was not applicable. Human involvement was limited to clinician coauthor review of eligibility labels, counterfactual validity, and rationale quality, as described in the human-subjects and annotator statement. Any deployment on real patient data would require appropriate institutional review, privacy safeguards, and consent or waiver procedures under the governing clinical and regulatory context.

Q Use of Artifacts, Licenses, and Intended Use

We use several existing artifacts. The patient cases and candidate trials are derived from the SIGIR patient–trial matching benchmark [10], which contains synthetic patient cases and ClinicalTrials.gov trial listings. We use these materials only for research evaluation of patient–trial matching methods, in a manner consistent with their intended benchmark purpose. We also use publicly available trial eligibility criteria from ClinicalTrials.gov as source text for formalization and evaluation. We cite the original benchmark and baseline systems, and use external software, model APIs, and supporting tools according to their respective licenses and terms.

Our use of existing artifacts is limited to the purposes for which they were made available. In particular, the SIGIR benchmark is used as an evaluation resource, not as a source of clinical recommendations. Although the patient cases are synthetic, any derived artifacts that depend on this benchmark should inherit its research-only context. They should not be repurposed for operational clinical screening, patient enrollment, or denial of trial access without separate institutional review, validation, and compliance with the original access conditions.

Our trial-side formalization builds on the semantic-parsing approach introduced by SATIR [24], which translates clinical trial criteria into symbolic constraints for scalable retrieval. SATIR is not currently released as a public software artifact. In this work, we adapt the criterion-formalization approach to the post-retrieval patient–trial matching setting by adding patient-conditioned extraction, solver-based eligibility evaluation, residual constraint analysis, and solver-grounded rationale generation. Our use of SATIR is limited to the research system evaluated in this paper and does not depend on a third-party software release.

The artifacts created in this work include pairwise eligibility labels, prompts, intermediate semantic parses, patient-side extractions, solver traces, residual constraints, rationales, and counterfactual examples. These artifacts are intended for research on clinical trial matching, evaluation, interpretability, and reproducibility. They are not intended for autonomous clinical decision-making, patient enrollment, or exclusion from trial consideration. Because some artifacts are derived from benchmark data and trial text, their use should remain compatible with the access conditions, licenses, and intended uses of the underlying sources.

If released, we will specify the license and intended use of each artifact category. Code and

prompts may be released for research and reproducibility, subject to the licenses of any dependencies. Derived data artifacts, including labels, traces, rationales, and counterfactuals, should be used only in research contexts and should not be treated as clinical ground truth. Any use with real patient records would require appropriate institutional review, privacy safeguards, security controls, and prospective validation in the intended clinical workflow.

Artifact documentation. We document the main artifacts used and produced in this work in the dataset and appendix sections. Appendix F reports the number of patient–trial pairs, patients, trials, label distribution, and text-length statistics. The benchmark consists of English synthetic patient cases paired with ClinicalTrials.gov eligibility criteria, and our evaluation is therefore limited to English clinical trial matching. We also document the prompts, matcher configurations, solver outputs, rationale formats, and counterfactual construction procedure in the appendices. Because the patient cases are synthetic and benchmark-derived, they should not be interpreted as representing the full demographic, institutional, or documentation diversity of real electronic health records.

R Information About Use of AI Assistants

We used AI assistants during the development, writing, and experimental monitoring of this work. AI assistance was used to support software development tasks, including code drafting, debugging, refactoring, and documentation of experimental pipelines. It was also used during manuscript preparation to help revise prose, improve organization, shorten or clarify paragraphs, and check the readability of technical explanations.

AI assistants were additionally used to support experiment monitoring and analysis, such as helping inspect logs, summarize intermediate outputs, identify possible implementation issues, and draft scripts for aggregating results. All experimental designs, system decisions, reported results, and scientific claims were reviewed and validated by the authors. AI-generated suggestions were treated as assistance only; the authors are responsible for the correctness of the code, analyses, writing, and conclusions in the paper.