Adaptive Domain Intelligence: A Falsifiable Protocol for LLM Feature Engineering and Outcome Judging

David C. Liu
Independent Research
https://daliu.github.io/research/adaptive-domain-intelligence/

24 August 2026 *(working paper)*

Locally precommitted FJC retrospective replay; all scores derive from committed artifacts.

Abstract

Can a language model discover useful predictors and make useful probabilistic predictions about outcomes that occurred after its knowledge cutoff without a model-visible outcome? A convincing affirmative test also requires proof that every input existed at forecast time. This paper specifies a falsifiable protocol for that claim and describes **Metis**, the supporting research architecture. Metis separates four functions that are often conflated: proposing features, extracting or computing them, fitting a statistical predictor, and judging an individual record. It records feature provenance, binds every run to its data and evaluation policy, uses entity- or time-aware splits, and propagates accepted, rejected, and inconclusive insights differently through a persistent hypothesis tree.

The principal experiment is a locally precommitted retrospective replay of 2026 Federal Judicial Center (FJC) civil-case records. A `gpt-5.6-sol` feature engineer proposed restricted, label-free feature plans; a fixed logistic regression selects and admits a plan on successive temporal blocks; and a fresh-context instance of the model forecasts a later lockbox in direct-record and feature-packet conditions. The target is the narrow administrative code "dismissed: settled," not latent settlement or case merit. Primary uncertainty is a deterministic district-cluster bootstrap, and abstentions count as reference-prior forecasts. The submitted feature plan did not pass held-out Admission and had a worse point estimate than the raw-field baseline on Lockbox. By contrast, both judge conditions improved Brier loss over the fixed historical prior: +0.0272 for direct records (95% CI +0.0106 to +0.0439) and +0.0262 for feature packets (+0.0120 to +0.0400). Feature packets did not improve on direct records (-0.0010; -0.0067 to +0.0046). Expected calibration error was worse than the fixed prior in both arms, so the result is about proper-score improvement, not demonstrated calibration improvement.

The replay excludes disposition, termination, judgment, and procedural-progress fields from model packets. It does not, however, establish a clean ex-ante forecast: the local source is a final-vintage termination extract, and official FJC guidance says that quarterly records may overwrite changing fields. Pro-se, in-forma-pauperis, and class-action values can change during a case; vintage availability of other supplied metadata was not independently established. The positive result therefore shows predictive signal in an outcome-field-excluded packet, not verified foresight from a point-in-time record. It supports neither adaptive feature-engineering utility nor the stronger claim posed above.

Earlier coded-data replications are reported only as exploratory construct checks. They do not show LLM feature engineering because their features were specified by researchers. After removing same-term leakage, the Supreme Court Database/Martin–Quinn check yields a mean five-fold temporal improvement of only 0.0081 AUC, with mixed fold effects. An ECtHR session extraction over 150 cases has 14.77% syntactic grounding failures and downstream AUC 0.498; semantic grounding has not yet been audited. These negative and qualified results are central. Metis is presented as an architecture and protocol, not evidence of deployable or general adaptive intelligence.

1 Introduction

Language models can translate qualitative records into candidate variables, explanations, and probability judgments. That combination suggests an adaptive empirical workflow: let a model formulate features, test them against observed outcomes, retain what survives, and use a fresh model context to forecast new records. The attractive version of this idea is also easy to make unfalsifiable. A model can receive outcome clues through document provenance, feature selection can overfit a reused validation set, entities can recur across nominally independent folds, and an LLM judge can reward artifacts produced by the same model family. Retrospective success can then be described as discovery even when the answer entered upstream.

We use **adaptive domain intelligence** in a deliberately narrow sense: the capacity of a precommitted system to improve out-of-sample probabilistic prediction by proposing auditable representations of domain records, while the outcomes, split, metrics, and stopping rule remain outside the model-visible adaptive loop. This definition makes failure observable. An LLM-derived feature plan has no demonstrated utility if it fails held-out admission, if its lockbox interval includes no improvement, or if its apparent evidence depends on information unavailable at the intended forecast time. A judge has no demonstrated utility against a specified comparator if it does not improve Brier loss over that comparator on all assigned cases, including abstentions.

The paper makes three contributions.

1. It describes an architecture that separates feature proposal, grounded extraction, statistical estimation, and judgment; records section-level provenance; and binds adaptive search to an immutable experiment identity.

2. It reports a post-knowledge-cutoff FJC temporal replay in enough detail to make its asymmetric outcome interpretable: positive judge scores against a fixed prior, no admitted feature plan, and no feature-over-direct judge gain. It also identifies a final-vintage metadata limitation that prevents the run from serving as a clean ex-ante test.

3. It reclassifies the existing evidence. Coded-feature replications are exploratory construct checks, the corrected SCDB effect is small and mixed, and the ECtHR extraction result reveals both syntactic grounding failures and chance-level prediction.

The inference is modest. Sol's probabilities contain predictive information for one administrative label in one termination-selected cohort, relative to a fixed prior, when explicit outcome fields are removed from a final-vintage extract. The generated representation did not improve the classical model or the paired judge, and the raw-field logistic model had lower observed Brier loss than either judge arm. Nothing here establishes ex-ante forecasting, causal discovery, legal validity, deployment fitness, economic sustainability, or general intelligence.

2 Architecture and research protocol

2.1 Separation of roles

Metis treats an LLM output as a proposal to be tested, not as an empirical result. The architecture separates the following roles.

This separation addresses two distinct risks. First, intrinsic self-correction is unreliable without an external signal [9, 11, 25]. Second, LLM evaluators can prefer outputs associated with their own generation process [19]. Metis does not assume that role separation eliminates model-family dependence; the paired direct and feature judge conditions are designed to measure one manifestation of it.

Table 1: Separation of authority in Metis.

Role	Inputs	Output	Authority it does **not** have
Feature engineer	Label-free schema and allowed source fields	Typed feature plan or extraction schema	Cannot inspect case outcomes or accept its own plan
Extractor/ transformer	Allowed record sections or fields	Values plus source evidence	Cannot change the target or evaluation split
Classical estimator	Frozen design matrix and training labels	Probabilities under a registered model	Cannot revise features or tune against the lockbox
Outcome judge	A fresh, outcome-field-excluded packet and a fixed reference prior	Probability, abstention, decisive quotes	Cannot see development scores, outcomes, or external tools
Evaluator	Frozen predictions, outcomes, groups, and metric	Effect estimate, interval, and decision state	Is not editable by the adaptive search

For tabular records, a plan uses a small deterministic feature language. For textual records, an extraction schema declares each feature's type, allowed values, and `document_source`. The validator can establish whether a cited span is literally present in the permitted input. It cannot establish that the span entails the extracted value. We therefore reserve **syntactic grounding** for substring validation and require a separate human or independent semantic audit before making a faithfulness claim.

2.2 Section provenance and fail-closed extraction

Text provenance is part of the feature definition. Current schemas restrict `document_source` to `complaint`, `brief`, `facts`, `arguments`, or `docket`; `ruling` is not an allowed predictive source. Extraction calls are grouped by section, and a call receives only the section it is permitted to inspect. If a required section is absent, the feature abstains rather than falling back to the whole document. Each committed value must carry at least one evidence span, and each span must be an exact substring of that section. Missing or non-verbatim evidence causes the value to be dropped and the event to be counted.

This is stricter than attaching citations after generation, but it is not a semantic guarantee. A model may quote a real but irrelevant sentence, reverse its meaning, or rely on an inference not supported by the quotation. Moreover, legacy unsectioned records are exposed only as `facts`, which is a compatibility rule rather than proof that their contents were available before decision. The ECtHR session run in Section 5.2 predates the repaired section-isolation path and is not retroactively certified by it.

2.3 Experiment identity and split discipline

Adaptive search is meaningful only if the object being searched does not change silently. An experiment identity binds the ordered full case records, labels, group and time arrays, target definition, and seed. The protocol identity also records the evaluator, gate settings, backend and default model, task, and seed policy. A policy fingerprint covers the schema, task, extraction configuration, guardrails, decision rule, and model configuration while excluding lineage metadata. Resuming against a mismatched identity fails instead of inheriting memory from a different experiment.

A serializable `SplitSpec` supports three modes:

- `random`, for exchangeable observations when no relevant entity or time structure is available;
- `group`, which keeps a recurring entity wholly within a fold and fails explicitly when groups are missing, unknown, or otherwise unusable; and

- `temporal_group`, which cuts only at complete time values. Every evaluation observation is later than every training observation; a term or date value never straddles the boundary.

The last property matters for court-term data: splitting rows within a term can allow the same institutional state to appear on both sides even if row order is chronological. Deterministic cluster/bootstrap and block-permutation primitives move observations together by entity or time block. Their presence does not repair older pooled or row-resampled analyses; each reported experiment must state which primitive it actually used.

2.4 Persistent hypothesis trees and tri-state memory

Metis represents adaptive work as a tree of hypotheses, artifacts, evidence, and evaluation receipts. A candidate receives one of three states:

- **accepted**: it satisfies the registered held-out gate and may inform later branches;
- **rejected**: it fails a decisive criterion; and
- **inconclusive**: available evidence does not distinguish it from the registered boundary.

Only accepted insights and genuine refutations persist as cross-run guidance. A rejection suppresses a proposal only when its interval wholly misses the required boundary or a registered fairness veto applies. An inconclusive result remains in the tree as an audit receipt but is not converted into negative memory. This distinction prevents absence of evidence from becoming a durable claim that a feature "does not work."

The tree mechanism is inspired primarily by the RUC-NLPIR/Microsoft Research **Arbor** project, described in *Toward Generalist Autonomous Research via Hypothesis-Tree Refinement* [10]. A different paper with the same system name—AMD's *Arbor: Tree Search as a Cognition Layer for Autonomous Agents* [22]—is conceptually related but is not the implementation basis for Metis.

Table 2: Relationship between published Arbor mechanisms and Metis.

Source mechanism	Status in Metis
Persistent branching among hypotheses	Adopted from RUC Hypothesis-Tree Refinement
Evidence-conditioned expansion and insight propagation	Adopted, with tri-state memory added
Held-out admission before promotion	Adopted as an experimental gate
Long-lived coordinator with isolated short-lived executors/worktrees	Not reproduced as the RUC runtime
Dynamic domain specialists, critic, persistent knowledge-base scoring, and GPU-serving optimization	Not ported from AMD Arbor
Typed legal policies, full experiment identity, section provenance, split validation, cluster-aware inference, and outcome-field-excluded judge packets	Metis extensions

The AMD paper is therefore a related cognition-layer design, not evidence that the FJC experiment implements its profiler, specialist, critic, or throughput stack. Conversely, invoking "Arbor" without an arXiv identifier obscures which research procedure is being claimed.

Table 3: Evidence classes and their permitted inferences.

Evidence class	Examples in this paper	Permitted inference
Locally precommitted retrospective replay	Post-cutoff FJC feature and judge arms	Test of constrained LLM feature-engineering and judging utility for the specified target and final-vintage cohort; not a clean ex-ante forecast
Exploratory construct check	SCDB/Martin–Quinn, Carp–Manning, Songer, earlier coded FJC analyses	Whether researcher-coded variables behave broadly as domain theory suggests under the stated split; not evidence that an LLM discovered them
Mechanism/ diagnostic check	ECtHR extraction, planted synthetic faults, identity and split tests	Whether a component detects or exposes a specified failure; not real-world predictive validity

3 Evidence hierarchy

The present evidence falls into three classes, which support different claims.

This hierarchy corrects an important category error in earlier drafts. The SCDB, Carp–Manning, Songer, and legacy FJC schemas were hand-coded. Their results can test construct definitions and statistical plumbing, but cannot show that an LLM is an effective feature engineer. Likewise, recovering a familiar empirical association retrospectively is not equivalent to predicting a future outcome without access to it.

4 Post-cutoff FJC outcome-withheld retrospective replay

4.1 Research question and target

The replay asks whether a frontier model, first as a constrained feature engineer and later in a fresh context as a probabilistic judge, can extract signal about a verified administrative outcome that occurred after its documented knowledge cutoff when the packet omits the outcome. This operational question is weaker than ex-ante forecasting, which additionally requires every input to be reconstructed from a point-in-time source.

The primary model is `gpt-5.6-sol`, run through Codex CLI 0.144.6. The local CLI configuration recorded `model_reasoning_effort="ultra"`; this is a run label, not the public API's name for an effort tier. OpenAI's model documentation lists 16 February 2026 as the model's knowledge cutoff and currently documents effort levels through `max` [20]. The run did not retain a provider snapshot identifier that would establish parameter-level immutability.

The target is `DISP = 13` in the FJC Integrated Database, labeled "dismissed: settled." It is a reliably scorable database code, but it is an incomplete proxy for settlement. Voluntary dismissals and consent judgments may encode settlements elsewhere, and the label does not measure the merits of a case. All conclusions below use the strict-code name rather than silently replacing it with latent settlement.

4.2 Cohorts and commitments

The source is a locally retained FJC FY2021–2026 civil IDB bundle. Eligible cases were filed on or before the model cutoff and received a merits-style termination in the designated window. Deterministic, unstratified sampling uses base seed `20260823`; labels were not used to balance samples.

"Merits-style" is itself an outcome-conditioned rule. Before sampling, the loader excludes final `DISP` codes `{0, 1, 10, 11, 16, 18, 19, 20}` as well as statistical closures. A post-run audit of the Lockbox date window found 14,473 otherwise valid terminations; the disposition rule removed 1,977 and left the 12,496-case eligible population recorded in the commitment. Strict-code prevalence changes from 17.90% before that exclusion to 20.73% after it. Sampling is unstratified only *within this outcome-defined population*. Membership could not have been known at the forecast cutoff.

Table 4: Precommitted temporal cohorts for the retrospective FJC replay.

Cohort	Temporal definition	Fixed n	Role
Historical landmark	Pending at 31 Dec. 2025; terminated 1 Jan.–16 Feb. 2026	5,000	Fit classical comparators and estimate the reference prevalence
Development	Terminated 17–28 Feb. 2026	160	Select among three roots and at most one two-child refinement round
Admission	Terminated 1–15 Mar. 2026	160	One held-out admission decision
Lockbox	Terminated 16–31 Mar. 2026	160	One final feature and judge evaluation

The scored Lockbox contains 40 positive cases (25.0%) across 57 district clusters. This differs from the frozen Historical prevalence of 21.64%, which is the reference forecast supplied to the judge.

The historical age variable is measured at the 31 December 2025 landmark, when all historical cases were still pending, rather than at their later termination. This correction was recorded in Protocol Amendment 1 after the root feature proposals but before any case-level score, reference prevalence, or judge packet was constructed. Source, selected-row, label, packet, prompt, trace, and prediction hashes are retained. The freeze is a local commitment dated 23 August 2026 and has no external timestamp; it establishes consistency of retained artifacts after that point, not prospective registration before the outcomes existed. The controller had inspected aggregate cohort counts and base rates. The retained procedure and artifacts show no case-level outcome in a model packet and no outcome-based balancing of sampled rows, but they do not provide the independent controls of a prospective registration.

4.3 Packet exclusion, field vintage, and access boundary

The packet contains circuit, an opaque district code, nature-of-suit bucket, jurisdiction basis, class-action allegation, jury demand, pro-se status, in-forma-pauperis status, removal status, demanded amount or band when present, filing year and month, and age computed at the designated cutoff. Party names, docket numbers, source row numbers, termination dates, disposition codes, judgment fields, procedural-progress codes, and later court-congestion measures are excluded. Opaque case identifiers are not sufficient to join packets to public dockets, and a recursive firewall checks canonical packet JSON before hashing.

This is an *outcome-field exclusion boundary*, not a verified point-in-time information firewall. The IDB integrates filing, pending, and termination records, and each quarterly release reflects the current record. Official FJC guidance warns that changed information can be overwritten and recommends retaining quarterly vintages; it specifically says that pro-se, IFP, and class-action fields may change over a case's life and lack quality-control checks [4]. The local bundle contains final termination records, not archived pending snapshots.

There is also a narrower software-boundary limitation. Labels were absent from rendered packets, prompts, feature transformations, and provider calls. But the tree-evaluation and lockbox-staging controller loaded the private answer file in the same process as public records; staging used Historical answers to compute the reference prior. Thus outcome non-use in model-facing computations is auditable, but label access was not enforced by process isolation. The literal statement in Protocol

Table 5: Model-visible FJC fields and temporal-vintage status.

Model-visible fields	What the retained source establishes	Ex-ante status
Filing date; derived filing year, month, and cutoff age	Filing date is an IDB core field; age is deterministically computed	Best-supported as pre-cutoff, though read from a later extract
Circuit and district token	Location in which the case was filed	Likely stable; no archived row was retained
Nature of suit, jurisdiction, and origin/removal	Present in the final civil record	Filing-time vintage not independently verified
Pro-se, IFP, and class-action status	Present in the final civil record	Explicitly mutable under FJC guidance
Jury demand and demanded amount/band	Present in the final civil record	Filing-time vintage not independently verified

Amendment 2 that Lockbox labels remained behind the scoring boundary throughout is superseded by this implementation audit.

These controls prevent direct disposition-field exposure and reduce public docket linkage. They cannot prove that a pretrained model never encountered a related case, that a final-vintage field contains no post-cutoff update, or that the replay is prospective. A clean replication requires an archived pending-record snapshot created on or before 16 February 2026, or a genuinely future unresolved cohort.

4.4 Feature-tree procedure

Root `F0` is the registered set of allowlisted raw fields. In a fresh, label-free session, the feature engineer proposes exactly three plans, each containing no more than 12 derived features. Plans are executable only in a deterministic DSL with `copy`, `equals`, `log1p`, `threshold_ge`, `boolean_and`, `categorical_cross`, and `product` over allowlisted fields. Free-form generated text is not a model input.

Every plan is evaluated by the same scikit-learn pipeline [21]: `DictVectorizer`, followed by `StandardScaler` with `with_mean=False`, and `LogisticRegression` with `C=1` and `max_iter=2000`. There is no class weighting, hyperparameter search, or model-family search. Models fit on Historical and are scored on Development by Brier loss. The engineer receives only aggregate Development metrics and plan definitions, not cases, predictions, or labels, and may propose at most two children of the best root. Because Development is reused after this feedback, all child performance is adaptive and descriptive.

The implementation treats a feature plan as a *replacement representation*, not an augmentation of `F0`. `F0` receives all 14 allowlisted raw fields; a planned model receives only its derived values. Selected plan `F2` therefore compresses the record to transformations of pro-se status, IFP status, nature of suit, and jurisdiction basis, dropping the other raw fields. Its comparison asks whether that generated compression can replace the raw representation, not whether engineered features add incremental value when appended to it.

The first child-plan request failed at provider schema validation. A repaired schema produced two children, but `F2A` contained a categorical cross between a categorical and a Boolean field that the local DSL rejected. A second, label-free model call removed that feature before Development evaluation. This structural-conformance retry was not authorized by the frozen retry rule, which covered judge formatting, and was not recorded in the two contemporaneous amendments. Traces and outputs are retained, but the exact retry prompt bytes were reconstructed post-run from the retained rollout command, base prompt, and invalid response; the reconstruction matches the recorded SHA-256 `46136a9b190bf91f02ecbad8c50c79936bd4a0ec1fb71c4a10e6af0083744d29`. The retry prompt

was not included in the contemporaneous tree manifest, and the current generalized batch script does not reproduce its historical wording. F2, not either repaired child, remained selected, so the repair does not change the submitted-plan result; it does weaken procedural completeness.

Root F3 also reached the registered 2,000-iteration optimizer limit without convergence. A post-run 5,000-iteration sensitivity converged after 2,011 iterations and changed its Development Brier only from 0.16054 to 0.16056; F3 remains last and the selected plan is unchanged.

The best Development plan is then frozen. Models refit on Historical + Development and are evaluated once on Admission. Admission requires positive district-cluster-bootstrap Brier improvement over F0, with the lower endpoint of the 95% interval above zero. If the gate is not passed, the adaptive classical artifact falls back to F0. Failure to pass is not itself a refutation: a confidence interval spanning zero is inconclusive under the tri-state evidence policy. The best submitted LLM plan remains frozen for the judge's feature-packet arm even if not admitted, so an unsuccessful proposal cannot disappear from the record. The final classical models fit on Historical + Development + Admission and are evaluated once on Lockbox.

4.5 Judge conditions and execution boundary

Each lockbox case appears in two paired information conditions:

1. direct_text: deterministic field=value lines from the allowlisted final-vintage record;
2. feature_packet: values computed by the frozen submitted feature plan, each accompanied by its exact source lines.

The judge receives the strict-code Historical prevalence as an explicit prior and returns $P(\texttt{DISP} = 13)$, an abstention flag, and exact packet quotes for the features it considered decisive. Quote presence is checked mechanically and is reported only as syntactic grounding.

The prompt asks whether the case will eventually receive code 13, but does not tell the judge the Lockbox termination horizon or the final-disposition eligibility exclusions. The probabilities are therefore scored on a population whose selection rule was not fully stated to the forecaster, another reason not to interpret them as calibrated forecasts for a well-defined cutoff-time population.

Feature engineering and judging use separate ephemeral sessions. The judge is not shown Development or Admission scores. Batches contain at most 40 cases, ordered by opaque identifier, and run in a clean temporary directory with project instructions ignored, a read-only sandbox, a no-tools instruction, and a trace audit that rejects tool calls. One retry is permitted to repair output format only; performance-driven retries are prohibited. These measures limit within-experiment contamination but cannot establish independence between two instances of the same pretrained model.

Protocol Amendment 2 records implementation-conformance repairs made after Admission and describes them as preceding Lockbox outcome access. Two preflight calls were rejected by the provider because Pydantic defaults did not satisfy its strict schema and produced no forecasts. One case was then forecast in both arms to test the grounding path; neither forecast was joined to its answer or scored. The final path normalized nullable required fields, implemented the already specified 40-case batches, and stated the exact quote-to-source mapping in the prompt. Failed and smoke traces are retained separately. No packet, feature, model, outcome, estimand, or decision rule changed in response. The later code audit in Section 4.3 qualifies the amendment's access claim: the controller could deserialize Lockbox answers during staging even though those answers were not inputs to packet construction or provider calls.

4.6 Estimands, uncertainty, and decision rules

The frozen document names mean per-case Brier improvement over the fixed Historical-prevalence forecast as the primary judge estimand, but it does not unambiguously say whether this applies to the direct arm alone or to both arms; its claim map separately names direct versus prior and feature versus direct. We do not resolve that ambiguity after seeing outcomes. We report both arm-versus-prior contrasts and add a conservative sensitivity treating them as two co-primary tests. Feature-packet versus direct-text remains the representation contrast. Missing and abstained forecasts receive the reference prevalence, making every score intention-to-forecast; complete-case scores and coverage are secondary.

The primary adaptive-feature estimand is paired Lockbox Brier improvement of the admitted classical artifact over `F0`. Secondary estimands are feature-packet versus direct-text Brier difference, log loss, AUROC, fixed-bin expected calibration error, coverage, syntactic quote-grounding, token and failure counts, and exploratory nature-of-suit and circuit slices. No threshold-accuracy headline is used for the imbalanced target [2]. Because `F2` was not admitted, the primary admitted-artifact contrast is mechanically `F0` versus itself and equals zero. The reported `F2`-versus-`F0` Lockbox contrast is descriptive; it prevents the submitted plan from disappearing, but is not a successful feature-utility test.

All reported paired intervals use 10,000 district-cluster bootstrap draws, moving cases from the same district together, and displayed permutation p-values use 10,000 one-sided district-block permutations. `20260823` is the base seed, not the seed of every contrast. Tree receipts use `20260823`, `20260833`, and `20260843` for roots `F1`–`F3`, `20260853` and `20260863` for children `F2A`–`F2B`, and `20270823` for Admission. This offset schedule was implemented but not stated in the frozen protocol, which named only the base seed. Lockbox inference uses `20260823` for direct versus prior, then `20260824`, `20260825`, and `20260826` for feature versus prior, feature versus direct, and submitted `F2` versus `F0`. The negative-control label permutation uses `20260823`. Positive utility requires an improvement interval whose 95% lower endpoint exceeds zero. The Lockbox is scored once; prompts, models, plans, samples, metrics, and stopping rules do not change after prediction commitment. Recomputing the judge and Admission intervals with the base seed for every contrast leaves all decisions unchanged.

The claim-to-test mapping is explicit.

4.7 Committed replay results

The following table was populated mechanically after all 320 forecasts were committed and the lockbox was scored once. Improvements are oriented so that positive values favor the named treatment.

Table 7: Committed FJC replay results.

Quantity	Committed value
Historical strict-code prevalence	0.2164
Lockbox outcome prevalence	40/160 = 25.0%
Lockbox inference clusters	57 districts
Selected submitted feature plan	F2
Development selection summary	F2 selected (Brier: F0 0.1204; F1 0.1374; F2 0.1097; F2A 0.1335; F2B 0.1235; F3 0.1605)
Admission `F2` versus `F0`	ΔBrier -0.0080; 95% CI [-0.0263, +0.0097]; one-sided permutation $p = 0.8051$
Admission decision	not admitted; inconclusive under the tri-state rule; classical fallback to F0
Lockbox admitted-classical Brier	0.1400

Quantity	Committed value
Lockbox classical ΔBrier over `F0`	+0.0000 (F0 fallback)
Lockbox classical district-cluster 95% CI	not applicable (F0 fallback)
Submitted `F2` versus `F0`, Lockbox	ΔBrier -0.0160; 95% CI [-0.0395, +0.0067]; one-sided permutation $p = 0.9016$
Direct judge versus reference	ΔBrier +0.0272; 95% CI [+0.0106, +0.0439]; one-sided permutation $p = 0.0016$
Feature judge versus reference	ΔBrier +0.0262; 95% CI [+0.0120, +0.0400]; one-sided permutation $p = 0.0003$
Two-test multiplicity sensitivity	individual 97.5% CIs: direct [+0.0083, +0.0470]; feature [+0.0098, +0.0418]
Feature versus direct judge (paired)	ΔBrier -0.0010; 95% CI [-0.0067, +0.0046]; one-sided permutation $p = 0.6308$
Syntactic quote-grounding rate	1227/1227 = 100.0% syntactic
Statistical interpretation	feature-engineering utility not demonstrated; both arm-versus-prior improvements observed; feature-over-direct gain not demonstrated; ex-ante utility not established

The adaptive classical path did not clear its gate. `F2` looked best on the reused Development block (Brier 0.1097 versus 0.1204 for `F0`) but its point estimate reversed on Admission: Brier 0.1441 versus 0.1361, for an improvement of -0.0080 (95% CI -0.0263 to +0.0097; one-sided permutation $p = 0.8051$). Because this interval crosses zero, `F2` was *not admitted* and the evidence is *inconclusive* under the precommitted tri-state rule, rather than a decisive rejection. On Lockbox, the frozen submitted plan again had a worse point estimate than `F0` (Brier 0.1560 versus 0.1400; improvement -0.0160, 95% CI -0.0395 to +0.0067; $p = 0.9016$). That interval also crosses zero. The admitted artifact is therefore `F0` itself, so its displayed improvement over `F0` is zero by construction.

The judge result has lower proper-score loss than the reference and no representation lift. Direct-record forecasts achieved Brier 0.1614 and AUROC 0.7680; feature-packet forecasts achieved Brier 0.1624 and AUROC 0.7739. Both had 100% coverage and improved on the historical-prevalence Brier of 0.1886, with cluster intervals excluding zero (raw one-sided permutation $p = 0.0016$ and 0.0003). Treating the two arm-versus-prior tests as co-primary yields Bonferroni-adjusted simultaneous 95% coverage via individual 97.5% bootstrap intervals: +0.0083 to +0.0470 for direct and +0.0098 to +0.0418 for feature. Their paired difference did not clear zero: the feature packet was 0.0010 Brier worse, with a 95% interval of -0.0067 to +0.0046 ($p = 0.6308$). The raw-field logistic model's Brier of 0.1400 was lower than either judge arm.

Calibration diagnostics require separate interpretation. The reference prior has ECE_{10} 0.0336, below the direct judge's 0.0650 and the feature judge's 0.0808, even though the judge arms improve Brier score and log loss. Mean forecast probabilities were 0.1997 for direct and 0.1826 for feature, both below the observed 0.2500 prevalence. Fixed-bin ECE is sample- and bin-dependent [7]. Thus the run does not establish that judge probabilities are calibrated in an absolute sense or better calibrated than the reference. In the classical negative control, permuting training answers drove AUROC to 0.4238 for `F0` and 0.4219 for `F2`, near or below chance as intended. This diagnostic supports signal in the supplied final-vintage fields and the scoring plumbing, not calibrated ex-ante forecasting or an additional confirmatory hypothesis test.

All 1,227 cited spans across 320 forecasts passed the precommitted syntactic audit. This is exact quote presence, not semantic entailment. The eight 40-case Sol Ultra batches completed without a format retry or tool event over an 18.1-minute two-worker wall span. Provider metadata records

Table 6: Prespecified claim-to-test mapping.

Claim	Required observation	Result if criterion is not met
LLM plan is admissible	Admission ΔBrier over `F0` > 0 with cluster 95% CI entirely above 0	Plan not admitted; inconclusive if the interval crosses 0; classical artifact falls back to `F0`
LLM feature engineering has Lockbox utility	Submitted plan's prespecified Lockbox contrast improves Brier with cluster 95% CI entirely above 0	No demonstrated feature-engineering utility; a CI spanning zero is inconclusive
Direct judge has predictive utility	Direct-arm intention-to-forecast Brier improves over the fixed prior with cluster 95% CI entirely above 0	Negative or inconclusive judge result
Feature representation helps the judge	Feature-packet Brier improves over paired direct packets with cluster 95% CI entirely above 0	No evidence that the representation helps judgment; a CI spanning zero is inconclusive
Feature packet improves over the prior	Feature-arm Brier contrast versus the fixed prior	Report alongside direct versus prior; multiplicity sensitivity treats both as co-primary because the registration is ambiguous
Quotes are semantically grounded	Independent semantic audit supports the value-evidence relation	Report syntactic quote presence only
Cross-family replication	All Claude predictions committed before Sol lockbox scoring	Report replication as unrun

Table 8: Secondary performance diagnostics. All rows contain 160 cases and 100% coverage. Lower is better for Brier, log loss, and ECE_{10}; higher is better for AUROC.

Cohort and arm	AUROC	Brier	Log loss	ECE_{10}
Admission `F0`	0.8161	0.1361	0.4119	0.0801
Admission submitted `F2`	0.7814	0.1441	0.4419	0.0255
Lockbox historical-prevalence reference	0.5000	0.1886	0.5655	0.0336
Lockbox classical `F0`	0.8360	0.1400	0.4418	0.1103
Lockbox submitted `F2`	0.7544	0.1560	0.4917	0.0448
Lockbox direct judge	0.7680	0.1614	0.4769	0.0650
Lockbox feature-packet judge	0.7739	0.1624	0.4814	0.0808
Permuted-answer control, `F0`	0.4238	0.2007	0.6533	0.0936
Permuted-answer control, submitted `F2`	0.4219	0.1918	0.5745	0.0834

555,254 input tokens (243,968 cached) and 109,207 output tokens, including 60,312 reasoning tokens. This establishes operational completion under batching; it is not a cost-effectiveness result, because subscription CLI billing was unavailable.

A Claude Fable 5 model at maximum effort was planned as a cross-family replication, but the installed CLI reported its weekly quota exhausted at the local freeze. No Claude Lockbox predictions were committed, so the replication is **unrun**. Any later Claude evaluation is a new, post-score replication and cannot be presented as part of this locally frozen run.

5 Corrected and negative evidence

5.1 SCDB/Martin–Quinn: a small, mixed construct check

An earlier analysis used a court ideology summary constructed from the same term's votes. Because those votes include the outcomes being predicted, the feature was contaminated by its target period. The corrected feature, `prior_term_court_median_ideology`, assigns term t only the court summary for term $t-1$ and is unavailable when no prior term exists. A unit-level check ensures that changing term t votes cannot change term t's predictor.

Evaluation uses whole-term temporal folds: a term never straddles training and test, and every training term is strictly earlier than its evaluation block. Across 9,092 usable SCDB cases and 79 terms, the five fold-level AUC effects of adding the lagged summary are shown in Table 9.

Table 9: Corrected SCDB/Martin–Quinn whole-term temporal effects.

Temporal fold	ΔAUC
1	+0.0045
2	−0.0014
3	+0.0135
4	+0.0340
5	−0.0101
Unweighted mean	+0.0081

Mean fold AUC rises from 0.5164 to 0.5245. Two of five folds are negative and the largest positive fold materially influences the mean, so the appropriate verdict is **small, mixed, and non-robust**, not a recovered universal signal. No pooled cross-era AUC or row bootstrap is used as confirmatory evidence.

There is a further temporal caveat. Martin–Quinn ideal points are retrospective dynamic estimates, and the locally joined series is not a set of archived vintage releases recreated at each forecast date. Lagging by term removes direct same-term construction, but a genuinely prospective institutional forecast would require the data vintage actually available then. This analysis is consequently an exploratory construct check and, because the feature was researcher-coded, provides no evidence of LLM feature engineering.

5.2 ECtHR: syntactic failures and chance prediction

The ECtHR session experiment is the only completed legacy path in this draft that uses an LLM to extract features from substantive text. It sampled 160 records and produced usable session artifacts for 150. Across 1,178 committed feature values, 174 had missing or non-verbatim evidence, a **14.77% syntactic grounding-failure rate**. Downstream session AUC is **0.498** (reported interval 0.360–0.719), essentially chance. A rule-based comparator is likewise near chance (AUC 0.490).

These numbers replace the earlier, incorrect claims of zero hallucination and successful grounded extraction. "Syntactic grounding failure" is the measured quantity: it does not distinguish fabrication, formatting error, span mismatch, or an evidence omission. Conversely, a quote that passes exact matching may still be semantically irrelevant. An independent semantic audit is pending, so no faithfulness rate is claimed.

The corpus also illustrates the provenance problem in legal text prediction. ECtHR fact statements are court-authored and may reflect a record assembled after the events or decision process of interest [1, 17]. The session artifacts predate the current section-level firewall. Chance prediction is therefore a useful negative diagnostic, but not a clean prospective forecast and not evidence that the repaired extractor has been validated end to end.

5.3 Other coded replications

Earlier Carp–Manning, Songer, and FJC analyses explored familiar judicial-party, panel-composition, repeat-player, and case-duration constructs. They remain useful for testing whether a schema can express domain hypotheses, but their features were coded by researchers rather than proposed by an LLM. Several also retain design limitations: Carp development and evaluation partitions have substantial judge recurrence; Songer's lead-judge grouping does not exclude other panel-member overlap; and legacy FJC analyses use a narrow disposition proxy and older row-level inference. These results are not used to validate feature engineering, judge calibration, or a general "known-answer" claim. Any future replication should declare an experiment identity, choose the recurring entity before analysis, and use entity-clustered uncertainty.

6 Interpretation and threats to validity

6.1 What the replay shows

Both judge arms have lower Brier loss than the frozen prior in the supplied final-vintage packets, including under the two-comparison multiplicity sensitivity. The feature-packet interval against direct records includes zero and its point estimate is slightly worse. The feature plan does not pass Admission, and its descriptive Lockbox contrast also spans zero around a negative point estimate. The supported claim is therefore narrow: the model extracts predictive signal from an outcome-field-excluded final-vintage record. The run does not establish that the signal existed at forecast time, that Sol discovered a useful representation, that the representation improved its own judgment, that calibration improved, or that either judge should replace the fixed classical model, which had lower observed Brier loss.

6.2 Label and selection limits

`DISP = 13` has construct undercoverage: settlements encoded as voluntary dismissal or consent judgment become false negatives relative to latent settlement. Administrative coding error is also possible. More importantly, the replay conditions on termination by 31 March 2026. It therefore estimates outcomes among cases that terminated in a short post-cutoff window, not settlement propensity among all cases pending or filed by the cutoff. Filing age and case type may predict which cases enter this selected cohort as well as how they are coded within it. It further conditions on the realized disposition not belonging to the exclusion set. Formally, the target is $P(\text{DISP} = 13 \mid \text{final DISP is not excluded, termination is in the window})$, even though neither membership condition was known at cutoff or fully stated in the judge prompt. The design neither identifies causal feature effects nor supports individual legal advice.

6.3 Retrospective, vintage, and model-dependence limits

All outcomes existed before the local freeze, aggregate cohort rates had been inspected, and there is no external timestamp. Hash commitments and retained traces make the run auditable from the local freeze onward, but do not create a prospective registration. More importantly, the final-vintage IDB source does not prove that mutable packet fields had their displayed values at the model cutoff. Excluding explicit outcome fields is necessary but insufficient for an ex-ante test. The same-process controller also deserialized private labels during staging, even though packet computation and provider calls did not use them. Both limitations require architectural repair, not rhetorical qualification alone.

The primary feature engineer and both judge arms use the same model family. Fresh contexts prevent score carryover within the run but do not create independent training histories. Each arm was dispatched in only four batches, with no repeated stochastic runs and no provider snapshot identifier. Cases in one batch may have dependent outputs, while the district bootstrap models only case-sampling dependence; it does not capture provider, batch, or rerun variability. As a diagnostic, direct-arm Brier improvements over the prior by batch were +0.0306, +0.0399, +0.0381, and +0.0005; feature-arm values were +0.0279, +0.0305, +0.0371, and +0.0095. Four context-dependent batches are not replicates, but the spread makes "sustainable" effectiveness an untested claim. The planned Claude replication is unrun because of quota, so cross-family robustness is unknown.

6.4 Statistical limits

The Lockbox contains 160 cases in 57 districts, and no prospective sample-size or minimum-detectable-effect analysis was retained. Effective sample size is smaller under clustering, so intervals can be wide even when point estimates appear favorable. Development selection compares several correlated plans and then permits one feedback round; the separate Admission and Lockbox blocks protect against reusing the same cases for plan promotion, but do not repair field vintage. The registration is ambiguous about whether one or two arm-versus-prior judge tests are primary; the Bonferroni sensitivity addresses that ambiguity after disclosure, not by rewriting the freeze. The implemented contrast-specific seed offsets were also not specified in the frozen text; same-base-seed sensitivity leaves every decision unchanged. Subgroup slices are exploratory and are not powered fairness audits. The frozen protocol listed nature-of-suit and circuit slices, but no slice artifact was produced before this audit; they are reported as omitted rather than reconstructed after seeing the headline. A deterministic sample from one short period does not establish representativeness across years, labels, courts, or domains.

6.5 Artifact availability

The implementation, data bundle, run traces, and freeze artifacts currently reside in a private working repository or local storage. The repository now contains a dependency lock, artifact map, hashes, prompts, prediction traces, and verification commands, but this is local auditability rather than public reproducibility. The source data, private labels, and more than a thousand run files have not been packaged under a release policy, and no independent rerun exists. Accordingly, we make no open-source, public-reproducibility, deployability, or production-readiness claim.

6.6 Decisive prospective test

A follow-up that answers the motivating question should begin with outcomes that do not yet exist. On a public date T, it would archive a point-in-time record for every eligible case then pending, select the Lockbox using only information observable at T, and define the event as `DISP = 13` within a fixed horizon such as 180 days. Cases still pending at the horizon would remain in the estimand as

non-events by that date rather than being excluded using a future disposition or termination status. Each allowed field would have a source-vintage receipt. A packet-building process with no credential or path to the later label store would produce the model inputs; private join keys would remain in a separate scoring environment.

Before model calls, the complete protocol, code identity, packet commitments, provider identifiers, and power simulation would receive an external timestamp. The Arbor-style tree would operate only on historical vintage-matched Development data and would submit one frozen plan to a distinct Admission block. The plan semantics would be declared as additive or replacement in advance. The prospective Lockbox would then compare (1) the admitted engineered classical model with `F0`, (2) a direct-record judge with a fixed non-LLM reference, and (3) the feature-packet judge with the paired direct judge. Primary contrasts, multiplicity control, cluster unit, seed schedule, calibration analyses, slices, abstention rule, and minimum detectable effect would all be fixed before prediction.

To test sustainable rather than one-run performance, each judge condition would be repeated across independently launched contexts and at least two model families under recorded serving versions. All predictions and raw responses would be committed before the horizon closes; only then would a scoring-only process obtain outcomes. Between-run variation would enter the uncertainty analysis instead of being hidden inside a case bootstrap. This prospective experiment—not another reinterpretation of the retained replay—is the test that could support adaptive feature-engineering and judging utility.

7 Related work

LLM-derived features. CHiLL uses language models to instantiate interpretable expert queries, while FeatLLM derives tabular features or rules for downstream few-shot learning [16, 8]. Metis shares the separation between representation and a smaller estimator but makes temporal admission, run identity, and evidence provenance explicit. The outcome-field-excluded FJC replay is needed because an architecture alone does not establish that generated features improve prediction.

Grounding and evaluation. Citation- and claim-level methods such as ALCE, FActScore, and RAGAS evaluate support after generation [6, 18, 3]. Exact-span checking offers a deterministic lower layer, not a substitute for semantic evaluation. LLM-as-judge studies further show position, style, and self-preference biases [19], which motivate frozen metrics and the paired packet experiment.

Temporal forecasting and leakage. Autocast organizes its news corpus by date to simulate the information available to contemporaneous forecasters [28]. ForecastBench instead evaluates questions that are unresolved at submission time [12]. Bench to the Future makes pastcasting repeatable with a hermetic offline corpus tied to each historical question [27]. ExAnte demonstrates that an instruction to honor a cutoff does not reliably suppress future knowledge already internalized by a model [15]. Together these designs distinguish three separate requirements: unresolved outcomes or a defensible pastcast, point-in-time external inputs, and control of model-internal temporal leakage. The current FJC replay satisfies explicit outcome-field exclusion but not the second requirement.

Adaptive agents and hypothesis trees. RUC Hypothesis-Tree Refinement treats research as persistent branching with evidence and backpropagated insights [10]. AMD Arbor frames tree search as a cognition layer coordinating specialists and a critic for systems optimization [22]. Metis adopts the former's experimental-tree logic, adds typed statistical identities and tri-state evidence, and uses the latter only as a conceptual comparison.

Legal prediction and leakage. Work on ECtHR texts reported strong retrospective classification from court-authored facts [1], while later reviews emphasize the distinction between case identification and prospective forecasting [17]. Katz et al. [13] illustrate prediction from structured, temporally available Supreme Court variables. The present protocol treats provenance and time as parts of the estimand rather than post hoc caveats.

Judicial-politics constructs. Attitudinal, ideal-point, repeat-player, and panel-effects research motivates the exploratory schemas [5, 14, 23, 24, 26]. Those literatures supply hypotheses, not labels that certify an adaptive agent. A modern predictor may fail to recover an average historical association for legitimate reasons, while a successful retrospective replication may still be leaked or overfit.

8 Ethics and intended use

Court records describe people and organizations in consequential disputes. Prediction can reproduce historical inequities, encourage automation bias, and convert an administrative code into an unjustified claim about merits or party behavior. The current study is a methods experiment. It does not recommend use in adjudication, bail, sentencing, immigration enforcement, case valuation, or legal representation. Opaque identifiers and restricted packets reduce direct exposure in model prompts but do not eliminate privacy or re-identification risk in source data.

An abstention flag, proper-scoring loss, provenance trace, or fairness veto may make a research artifact easier to inspect; none makes it safe to deploy. Any later application would require a task-specific legal basis, representative data, measurement validation, independent bias and security review, human authority, appeal and monitoring procedures, and evidence that the application is beneficial relative to non-automated practice.

9 Conclusion

Adaptive domain intelligence should be evaluated by whether a frozen procedure survives opportunities to fail. Metis makes feature proposals executable, keeps outcomes outside model packets, records provenance and identity, separates inconclusive evidence from refutation, and reserves a final temporal block for one score. The audit also shows that these controls must extend to source vintage and process-level label isolation.

The result draws a useful boundary. Sol produced fully covered predictions from outcome-field-excluded final-vintage packets, and both arms had lower Brier loss than the prior. Yet neither arm improved calibration, the LLM-engineered plan was not admitted, its Lockbox estimate was negative and inconclusive, and it did not improve the judge over direct fields. Because mutable metadata were not reconstructed from an archived cutoff-time snapshot, the experiment does not answer whether the model predicted a then-unknown outcome from then-available facts. Researcher-coded replications remain separate; the corrected prior-term SCDB effect is small and inconsistent across time; and the ECtHR session run combines 14.77% syntactic grounding failures with chance-level prediction. The next decisive test is a prospectively committed unresolved cohort or a hermetic pending-snapshot pastcast. Until then, Metis has a promising protocol and an auditable outcome-withheld result, not demonstrated adaptive utility.

A Submitted F2 plan and replay artifacts

Plan identity and theory. The selected submitted plan has identifier `F2`, protocol `metis.fjc-postcutoff.plan-selection.v1`, and SHA-256 digest `c78ad2986927d837c235474c39c83517`

0e20d38044619f0047aac68b6b9ef15b. Its frozen hypothesis states that litigant capacity, repeat-player involvement, and remedial structure affect both the feasibility of negotiation and whether a negotiated resolution is administratively recorded as code 13. It anticipates lower explicit code-13 recording for self-represented, fee-waived, prisoner, immigration, and social-security matters, while allowing government roles, civil-rights, labor, and forfeiture matters to follow distinct regimes. The hypothesis is falsified if these distinctions provide no stable separation.

Table 10: The 12 deterministic features in submitted plan F2.

Feature	Operation	Source input(s) or comparison
f2_pro_se_status	copy	pro_se
f2_ifp_status	copy	in_forma_pauperis
f2_pro_se_ifp_joint	boolean_and	pro_se, in_forma_pauperis
f2_prisoner_suit	equals	nature_of_suit = prisoner
f2_immigration_suit	equals	nature_of_suit = immigration
f2_social_security_suit	equals	nature_of_suit = social_security
f2_civil_rights_suit	equals	nature_of_suit = civil_rights
f2_labor_suit	equals	nature_of_suit = labor
f2_forfeiture_suit	equals	nature_of_suit = forfeiture
f2_us_plaintiff_basis	equals	jurisdiction_basis = us_plaintiff
f2_us_defendant_basis	equals	jurisdiction_basis = us_defendant
f2_suit_jurisdiction_pair	categorical_cross	nature_of_suit, jurisdiction_basis

Judge prompt and response contract. Each batch prompt instructs the model to use only the supplied case packet, estimate $P(\text{positive class})$ on $[0, 1]$, avoid outside case-specific knowledge or guessing withheld facts, abstain when the packet does not support a responsible probability, and copy exact packet substrings for the few features that materially change the estimate. It requires exactly 40 independently treated records in schema-conforming JSON. Every response record requires `case_id`, nullable `probability`, `abstain`, nullable `abstain_reason`, and `decisive_features`. The direct arm supplies allowlisted field lines; the feature arm supplies the 12 computed F2 values, descriptions, abstention states, and supporting source lines. Both arms state the positive class, prediction question, historical prevalence 0.2164, historical sample size 5,000, historical snapshot 31 December 2025, and information boundary 16 February 2026. The appendix summarizes rather than reprints the case-bearing prompts; the exact prompt and schema files are retained below.

Repository-relative artifact map.

- Submitted plan and full descriptions: `experiments/fjc_postcutoff_2026/tree/selected_submitted_plan.json`.
- Label-free root prompt and returned plans: `experiments/fjc_postcutoff_2026/root_feature_prompt.md` and `experiments/fjc_postcutoff_2026/runs/feature_tree/root_plans.json`.
- Permitted feedback prompt: `experiments/fjc_postcutoff_2026/tree/refinement_prompt.md`.
- Admission receipt: `experiments/fjc_postcutoff_2026/tree/admission_receipt.json`.
- Eight exact batch prompts and response schemas: `experiments/fjc_postcutoff_2026/lockbox/batches/`.
- Individual canonical case prompts: `experiments/fjc_postcutoff_2026/lockbox/prompts/`.

- Committed 320-record prediction set: `experiments/fjc_postcutoff_2026/lockbox/predictions.json`; commitment SHA-256 `6795808c7fb63fb24f2f973a41c6af057f6d48f0be076043abf7f96159f46cce`.
- Final scores, uncertainty, control, and grounding audit: `experiments/fjc_postcutoff_2026/score/results.json`.

Committed feature-packet shape. A feature-arm record contains an opaque case ID; the fixed Historical prevalence and cutoff date; a list of `{name, value, description, evidence_quotes, confidence, abstained}` objects; and no disposition, termination date, party name, or docket number. Each evidence quote is the exact deterministic `field=value` line used to compute that feature. This representation makes transformations reproducible, but exact quote presence does not resolve the final-vintage problem described in Section 4.3.

References

[1] Aletras, N., Tsarapatsanis, D., Preoţiuc-Pietro, D., and Lampos, V. (2016). Predicting judicial decisions of the European Court of Human Rights: a natural language processing perspective. *PeerJ Computer Science*, 2:e93.

[2] Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3.

[3] Es, S., James, J., Espinosa-Anke, L., and Schockaert, S. (2024). RAGAS: Automated evaluation of retrieval augmented generation. *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*.

[4] Federal Judicial Center. (2019). *The Integrated Database: A Research Guide*. https://www.fjc.gov/sites/default/files/IDB-Research-Guide.pdf.

[5] Galanter, M. (1974). Why the "haves" come out ahead: speculations on the limits of legal change. *Law & Society Review*, 9(1):95–160.

[6] Gao, T., Yen, H., Yu, J., and Chen, D. (2023). Enabling large language models to generate text with citations. *Proceedings of EMNLP 2023*.

[7] Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. *Proceedings of ICML 2017*.

[8] Han, S., Yoon, J., Arik, S. O., and Pfister, T. (2024). Large language models can automatically engineer features for few-shot tabular learning. *arXiv preprint arXiv:2404.09491*.

[9] Huang, J., Chen, X., Mishra, S., Zheng, H. S., Yu, A. W., Song, X., and Zhou, D. (2023). Large language models cannot self-correct reasoning yet. *arXiv preprint arXiv:2310.01798*.

[10] Jin, Q., et al. (2026). Toward generalist autonomous research via hypothesis-tree refinement. *arXiv preprint arXiv:2606.11926*.

[11] Kamoi, R., Zhang, Y., Zhang, N., Han, J., and Zhang, R. (2024). When can LLMs actually correct their own mistakes? A critical survey of self-correction of LLMs. *Transactions of the Association for Computational Linguistics*, 12:1417–1440.

[12] Karger, E., Bastani, H., Yueh-Han, C., Jacobs, Z., Halawi, D., Zhang, F., and Tetlock, P. E. (2025). ForecastBench: a dynamic benchmark of AI forecasting capabilities. *International Conference on Learning Representations*.

[13] Katz, D. M., Bommarito, M. J. II, and Blackman, J. (2017). A general approach for predicting the behavior of the Supreme Court of the United States. *PLOS ONE*, 12(4):e0174698.

[14] Martin, A. D., and Quinn, K. M. (2002). Dynamic ideal point estimation via Markov chain Monte Carlo for the U.S. Supreme Court, 1953–1999. *Political Analysis*, 10(2):134–153.

[15] Liu, Y., Wei, X., Shi, L., Li, X., Zhang, B., Dhillon, P., and Mei, Q. (2026). ExAnte: a benchmark for ex-ante inference in large language models. *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics*, 1551–1571. https://doi.org/10.18653/v1/2026.eacl-long.72.

[16] McInerney, D. J., Young, G., van de Meent, J.-W., and Wallace, B. C. (2023). CHiLL: zero-shot custom interpretable feature extraction from clinical notes with large language models. *Findings of EMNLP 2023*.

[17] Medvedeva, M., Vols, M., and Wieling, M. (2023). Rethinking the field of automatic prediction of court decisions. *Artificial Intelligence and Law*, 31:195–212.

[18] Min, S., Krishna, K., Lyu, X., Lewis, M., Yih, W.-t., Koh, P. W., Iyyer, M., Zettlemoyer, L., and Hajishirzi, H. (2023). FActScore: fine-grained atomic evaluation of factual precision in long form text generation. *Proceedings of EMNLP 2023*.

[19] Panickssery, N., Bowman, S. R., and Feng, S. (2024). LLM evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37.

[20] OpenAI. (2026). *GPT-5.6 Sol model*. https://developers.openai.com/api/docs/models/gpt-5.6-sol. Accessed 24 August 2026.

[21] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[22] Prakriya, N., et al. (2026). Arbor: tree search as a cognition layer for autonomous agents. *arXiv preprint arXiv:2606.12563*.

[23] Segal, J. A., and Spaeth, H. J. (2002). *The Supreme Court and the Attitudinal Model Revisited*. Cambridge University Press.

[24] Songer, D. R., Kuersten, A., and Kaheny, E. B. (1999). Why the haves don't always come out ahead: repeat players meet amici curiae for the disadvantaged. *Political Research Quarterly*, 52(3):537–556.

[25] Stechly, K., Valmeekam, K., and Kambhampati, S. (2024). On the self-verification limitations of large language models on reasoning and planning tasks. *arXiv preprint arXiv:2402.08115*.

[26] Sunstein, C. R., Schkade, D., and Ellman, L. M. (2004). Ideological voting on federal courts of appeals: a preliminary investigation. *Virginia Law Review*, 90(1):301–354.

[27] FutureSearch, Wildman, J., Bosse, N. I., Hnyk, D., Mühlbacher, P., Hambly, F., Evans, J., Schwarz, D., and Phillips, L. (2025). Bench to the Future: a pastcasting benchmark for forecasting agents. *arXiv preprint arXiv:2506.21558*.

[28] Zou, A., Xiao, T., Jia, R., Kwon, J., Mazeika, M., Li, R., Song, D., Steinhardt, J., Evans, O., and Hendrycks, D. (2022). Forecasting future world events with neural networks. *Advances in Neural Information Processing Systems*, 35.