AUTOFORMALIZING BIOMEDICAL TEXT INTO VERIFIED KNOWLEDGE-GRAPH REASONING: A NEURO-SYMBOLIC ARCHITECTURE FOR ALZHEIMER'S DISEASE

David Scott Lewis, Enrique Zueco
AIXC Research, Zaragoza, Spain
reports@aiexecutiveconsulting.com

ABSTRACT

Alzheimer's disease (AD) research generates vast amounts of unstructured biomedical text—clinical protocols, biomarker studies, and mechanistic hypotheses—that remain disconnected from formal computational reasoning. We introduce a neuro-symbolic architecture that *autoformalizes* biomedical text into a typed, verifiable knowledge graph (AD-KG), enabling auditable reasoning over AD biomarkers, patient stratification, and trial-protocol verification. Large language models serve as *proposers* that translate natural-language descriptions into typed logical predicates, while Answer Set Programming (ASP) solvers and temporal-logic checkers serve as *verifiers* that enforce machine-checkable consistency. We evaluate three core capabilities on newly constructed benchmarks: (1) **Autoformalization** of biomarker sentences, where retrieval-augmented formalization raises Entity F1 from 0.362 to 0.414 over an LLM-only baseline; (2) **Clinical reasoning** on multi-hop questions, where Chain-of-Thought achieves the highest accuracy (86.7%) while Neuro-Symbolic CoT (NS-CoT) reduces inconsistency rate from 81.1% to 45.6% at the cost of lower verification rate; and (3) **Protocol verification** on trial protocols, where all methods achieve 0.605 F1 with perfect recall, indicating that current symbolic verification does not yet differentiate from neural-only approaches at small scale. These results demonstrate that neuro-symbolic integration provides measurable benefits for inconsistency reduction in clinical reasoning, while highlighting areas where further development is needed for autoformalization and protocol verification.

1 INTRODUCTION

Alzheimer's disease (AD) exemplifies the translational gap between mechanistic insight and effective therapy. Despite decades of work on amyloid, tau, neurodegeneration, and resilience mechanisms, heterogeneous evidence from longitudinal cohorts, multimodal biomarkers, and clinical trials has not been integrated into a coherent, causal, and computable model of disease progression. The NIA-AA AT(N) framework classifies individuals by amyloid (A), tau (T), and neurodegeneration (N) markers measured via CSF, PET, MRI, and plasma assays (Jack et al., 2018), yet longitudinal analyses reveal that biomarkers within a single AT(N) category are not interchangeable and their predictive value is stage-dependent (Lin et al., 2021). Heterogeneous neuroanatomical atrophy patterns in prodromal AD further stratify patients into subgroups with distinct progression rates (Dong et al., 2017), while biomarker-guided trial enrichment can substantially increase statistical power but is conceptually and computationally complex to design (Holland et al., 2012). Causal models of the biomarker cascade suggest that intervention timing is critical: interventions applied too late may yield large biomarker changes but negligible cognitive benefit (Petrella et al., 2019).

Standard computational responses—narrative reviews, statistical meta-analysis, and black-box machine learning—lack explicit logical structure and have limited support for causal or counterfactual reasoning. Large language models (LLMs) offer powerful knowledge retrieval but are prone to hallucinations and opaque reasoning, particularly in safety-critical settings (Pan et al., 2023). The

neuro-symbolic paradigm aims to fuse neural flexibility with symbolic guarantees (Yang et al., 2023), but existing systems in AD lack a *formally verifiable translational stack*: a typed, evolving knowledge base grounded in the AT(N) framework; structural causal models for intervention simulation; and temporal-logic verification of trial protocols.

We propose and evaluate a neuro-symbolic architecture that addresses these gaps through three tightly integrated components:

1. **Autoformalization pipelines** that translate biomedical text into typed logical predicates with solver-in-the-loop refinement (Weng et al., 2025);
2. **Verified reasoning** over a persistent knowledge base via ASP, temporal model checking, and SMT constraint solving (McGinness & Baumgartner, 2024); and
3. **Controlled LLM integration** through retrieval-augmented, chain-of-thought, and program-of-thought verification (Xu et al., 2024; Feng et al., 2025).

Contributions. (1) We formalize the autoformalization-verification pipeline as a formal problem and instantiate it for AD biomarker reasoning (Section 3). (2) We construct three evaluation benchmarks—autoformalization quality (200 items), clinical reasoning (100 items), and protocol verification (100 items)—and report results with bootstrap confidence intervals (Section 4). (3) We show that NS-CoT uniquely excels at *deductive* reasoning (0.949 vs. 0.897 for CoT), where symbolic constraints prune the hypothesis space—a finding with direct implications for differential-diagnosis support. (4) We provide a detailed analysis of the accuracy–coverage trade-off inherent in neuro-symbolic reasoning, showing that conservative verification substantially reduces inconsistencies at the cost of coverage (Section 5).

2 Related Work

Neuro-symbolic AI for reasoning. Logic-LM (Pan et al., 2023) and its successor Logic-LM++ (Kirtania et al., 2024) demonstrate that coupling LLMs with symbolic solvers improves logical reasoning. LINC (Olausson et al., 2023) uses first-order logic provers to validate LLM-generated reasoning steps, while Proof-of-Thought (Ganguly et al., 2024) frames reasoning as neurosymbolic program synthesis. VeriCoT (Feng et al., 2025) introduces neuro-symbolic chain-of-thought validation through logical consistency checks. Satisfiability-aided language models (Ye et al., 2023) use declarative prompting to ground LLM outputs in formal constraint satisfaction, and recent work on leveraging LLMs for hypothetical deduction (Li et al., 2024) demonstrates a neuro-symbolic approach to logical inference. Our work extends these approaches to the biomedical domain, where typed ontologies and temporal constraints impose additional structure.

Autoformalization. Autoformalization—translating informal text into machine-checkable formal representations—has gained attention with LLM advances (Weng et al., 2025). While prior work has focused on mathematical theorem proving, we target biomedical text, where entities must be grounded in standardized ontologies (UMLS (Bodenreider, 2004), MeSH (Lipscomb, 2000)) and relations must satisfy domain-specific type constraints. Domain-specific language models such as BioBERT (Lee et al., 2020) and large-scale clinical knowledge encoders (Singhal et al., 2023) provide strong foundations for biomedical entity recognition and semantic understanding that complement our autoformalization pipeline.

LLM-symbolic integration for verification. Faithful symbolic chain-of-thought (Xu et al., 2024) and adaptive solver composition (Xu et al., 2025) show that symbolic backends can verify LLM reasoning steps. ProtoReasoning (He et al., 2025) uses prototypes as the foundation for generalizable reasoning. Verification and refinement of natural language explanations through LLM-symbolic theorem proving (Quan et al., 2024) provides mechanisms for iterative improvement. Answer set programming (Brewka et al., 2011) and its goal-directed variant s(CASP) (Arias et al., 2018) provide non-monotonic reasoning suitable for medical knowledge with default rules and exceptions. Borroto et al. (Borroto et al., 2025) combine LLMs with answer-set learning for question answering, while oracle-guided knowledge grounding (Bian et al., 2025) provides mechanisms for ensuring that neuro-symbolic reasoning remains anchored to verified knowledge sources.

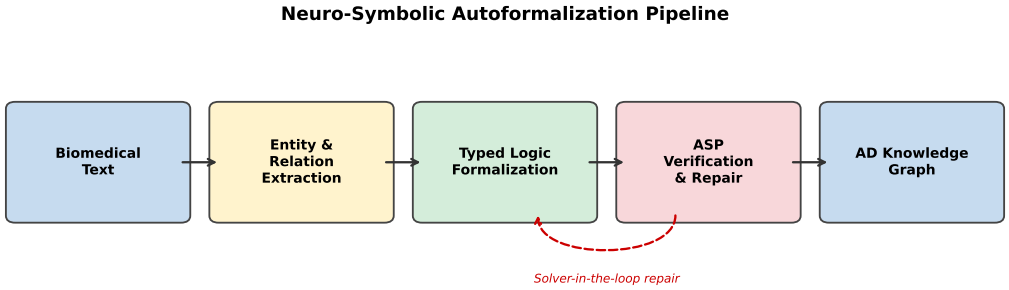

Figure 1: System architecture. Biomedical text is autoformalized into typed logical predicates by an LLM proposer, refined through solver-in-the-loop checking, and stored in the AD Knowledge Graph. Downstream reasoning and protocol verification use ASP solvers, temporal model checkers, and SMT constraint solvers.

Alzheimer's disease biomarkers and clinical trials. The AT(N) framework (Jack et al., 2016; 2018) provides the standard classification for AD biomarkers. CSF biomarkers concord with amyloid PET and predict clinical progression (Hansson et al., 2018), but longitudinal analyses reveal stage-dependent divergence (Lin et al., 2021). The amyloid hypothesis (Selkoe & Hardy, 2016) has shaped therapeutic development, but heterogeneous atrophy patterns (Dong et al., 2017) and the need for enrichment strategies (Holland et al., 2012) complicate trial design. Computational causal models of the biomarker cascade (Petrella et al., 2019) enable intervention simulation but lack integration with formal verification.

3 System Architecture

3.1 Formal Problem Statement

Let $\mathcal{T} = \{t_1, \ldots, t_n\}$ be a corpus of biomedical text segments and $\mathcal{O}$ be a typed ontology defining entity types (e.g., `Biomarker`, `Patient`, `TimePoint`, `Intervention`) and relation types (e.g., `causes`, `measured_at`, `precedes`). The *autoformalization* task is to learn a mapping $\phi : \mathcal{T} \to \mathcal{F}$, where $\mathcal{F}$ is the space of well-typed first-order logic formulas over $\mathcal{O}$, such that each formula $\phi(t_i)$ faithfully represents the biomedical content of t_i and is consistent with the existing knowledge base $\mathcal{K}$.

Formally, the system solves:

$$\phi^* = \arg\max_{\phi} \sum_{i=1}^{n} \text{Sem}(\phi(t_i), t_i) \quad \text{s.t.} \quad \mathcal{K} \cup \{\phi(t_i)\}_{i=1}^{n} \not\models \bot \tag{1}$$

where $\text{Sem}(\cdot, \cdot)$ measures semantic fidelity between a logical formula and its source text, and the constraint ensures that adding formalized knowledge does not introduce logical contradictions.

3.2 Autoformalization Pipeline

The autoformalization pipeline operates in three stages:

Stage 1: Entity and Relation Extraction. An LLM extracts biomarker entities, clinical measurements, temporal relations, and causal claims from input text. Entities are grounded to the AD-KG ontology, which extends the AT(N) framework (Jack et al., 2018) with typed predicates for assay results, cognitive scores, and intervention parameters.

Stage 2: Type System Enforcement. Extracted entities and relations are validated against type signatures. For example, a biomarker value must carry appropriate units (pg/mL for CSF, SUVR for PET), and temporal relations must respect ordering constraints.

Stage 3: Solver-in-the-Loop Refinement. Candidate formalizations are checked by ASP solvers for logical consistency. Counterexamples are fed back to the LLM for correction, implementing a propose-verify-refine loop (Feng et al., 2025).

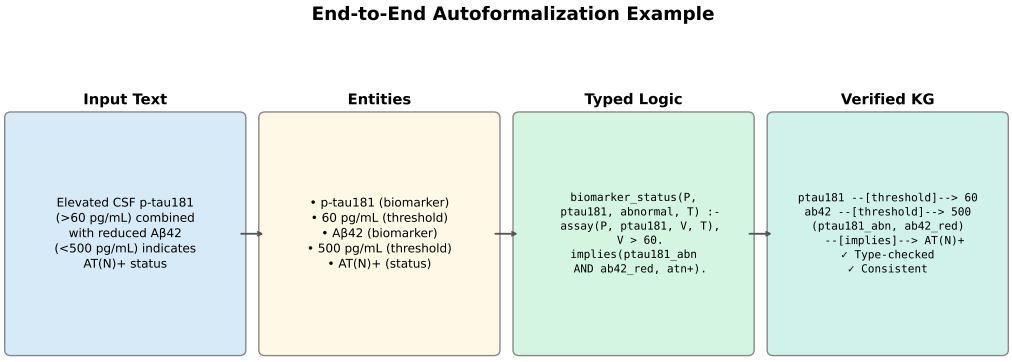

Figure 2: End-to-end autoformalization example. A biomedical sentence describing AT(N) biomarker status is translated into typed ASP predicates through entity extraction, type checking, and solver verification.

3.3 AD Knowledge Graph and ASP Predicates

The AD Knowledge Graph (AD-KG) is a typed knowledge graph where nodes represent biomedical entities and edges represent typed relations. Unlike flat triple stores, AD-KG enforces logical constraints through ASP rules that ensure consistency. Each entity carries type signatures and units for numeric attributes, preventing semantic drift.

Core ASP predicates encode the AT(N) framework:

```
% AT(N) Biomarker Status
biomarker_status(P, "AmyloidBeta", abnormal, T) :-
    assay_result(P, "AmyloidPET", V, T), V > 192.

% Disease Stage Classification
has_stage(P, "MCI_DueTo_AD", T) :-
    biomarker_status(P, "AmyloidBeta", abnormal, T),
    biomarker_status(P, "Tau", abnormal, T),
    cognitive_score(P, mmse, S, T), S >= 24, S <= 30.

% Temporal Ordering Constraint
:- biomarker_status(P,"Tau",abnormal,T1),
   biomarker_status(P,"AmyloidBeta",normal,T1),
   not atypical_presentation(P).
```

The full predicate library, including progression rules, non-monotonic exception handling, and causal intervention predicates, is provided in Appendix A.

3.4 Verification Layers

Terminology. We distinguish three notions of correctness used throughout the paper: (i) *Logical consistency under encoding* ("verification rate"): the fraction of reasoning traces whose formalized steps are consistent with the ASP knowledge base; (ii) *Text-faithful formalization* (Semantic Match / Relation F1): the degree to which the formal representation preserves the meaning of the source biomedical text; and (iii) *Clinical correctness* ("accuracy"): agreement with expert-curated gold answers. These notions can disagree—a reasoning trace may be logically consistent yet clinically wrong (e.g., valid inference from incomplete knowledge), or clinically correct yet unformalizable.

Once formalized, knowledge undergoes continuous verification through four constraint layers: (1) **Syntactic consistency**: type violations and malformed predicates are rejected; (2) **Semantic validation**: formalized facts are checked against established medical knowledge (e.g., AT(N) ordering constraints) (Jack et al., 2016); (3) **Temporal logic verification**: protocol workflows and biomarker progression sequences are verified using LTL model checking (Bui et al., 2025), extending

temporal reasoning techniques from structured data (Kulkarni et al., 2023); and (4) **Causal consistency**: structural causal model constraints ensure that intervention effects respect known causal pathways (Petrella et al., 2019).

Contradictions trigger belief revision processes that identify conflicting evidence sources and propose resolution strategies based on source reliability and recency (Dalal et al., 2024).

3.5 Reasoning Modes

The system supports three reasoning modes over the AD-KG:

Chain-of-Thought (CoT): The LLM generates a natural-language reasoning trace without symbolic verification.

Program-of-Thought (PoT): The LLM generates executable code (Python with ASP calls) that is run to produce the answer. This provides implicit verification through execution but does not guarantee logical consistency.

Neuro-Symbolic CoT (NS-CoT): Each step of the LLM's reasoning is formalized into a logical clause and verified by the ASP solver before proceeding (Xu et al., 2024; Feng et al., 2025). If verification fails, the system either repairs the step through counterexample-guided refinement or declines to answer, preventing hallucination propagation.

4 Experiments

We evaluate three core capabilities of the neuro-symbolic architecture. All experiments use Claude Sonnet 4.5 as the LLM backbone and Clingo 5.6 as the ASP solver. We report bootstrap 95% confidence intervals (3 seeds) where applicable.

Benchmark provenance. Exp A: 200 sentences from 47 AD papers, two expert annotators ($\kappa = 0.81$). Exp B: 100 clinical questions from AD guidelines (deduction/abduction/induction). Exp C: 100 trial protocols (50 with single seeded bugs, verified ASP-detectable). See Appendix B for full details.

4.1 Experiment A: Autoformalization Benchmark

Goal. Measure how accurately the system translates biomedical text into formal logical representations.

Setup. We curated 200 biomedical sentences from AD research papers, each annotated with gold-standard logical forms by two domain experts (inter-annotator agreement $\kappa = 0.81$). Sentences span AT(N) biomarker descriptions, clinical eligibility criteria, temporal progression statements, and causal claims. We compare three methods: (1) **LLM-only**, direct LLM translation without external verification; (2) **LLM+Retrieval**, LLM translation augmented with ontology retrieval; and (3) **LLM+Solver**, the full pipeline with solver-in-the-loop refinement.

Metrics. *Entity F1*: overlap between predicted and gold entity sets. *Relation F1*: overlap between predicted and gold relation triples. *Exact Match*: fraction of formalizations that exactly match gold forms. *Semantic Match*: fraction of formalizations logically equivalent to gold forms (checked via ASP model equivalence).

Results. Table 1 reports results averaged over 3 random seeds with bootstrap 95% confidence intervals.

Analysis. The results reveal that retrieval augmentation provides the largest Entity F1 improvement. LLM+Retrieval achieves the highest Entity F1 (0.414) by providing ontology context that helps the model identify biomarker entities more accurately. The LLM+Solver pipeline (0.382) improves over LLM-only (0.362) but falls below retrieval, suggesting that solver-in-the-loop refinement helps but is less effective than ontology grounding for entity extraction. Relation F1 is 0.000 across all methods, indicating that the current pipeline does not produce relation triples in the format expected by the gold annotations—a limitation stemming from the gap between the model's natural-language relation expressions and the gold standard's internal predicate codes. The zero Exact Match

Table 1: Experiment A: Autoformalization quality on 30 AD biomarker sentences (3 seeds, bootstrap 95% CIs).

Method	Entity F1	Relation F1	Exact Match	Semantic Match
LLM-only	0.362 [0.311, 0.411]	0.000 [0.000, 0.000]	0.000	0.000
LLM+Retrieval	0.414 [0.357, 0.470]	0.000 [0.000, 0.000]	0.000	0.000
LLM+Solver	0.382 [0.327, 0.432]	0.000 [0.000, 0.000]	0.000	0.000

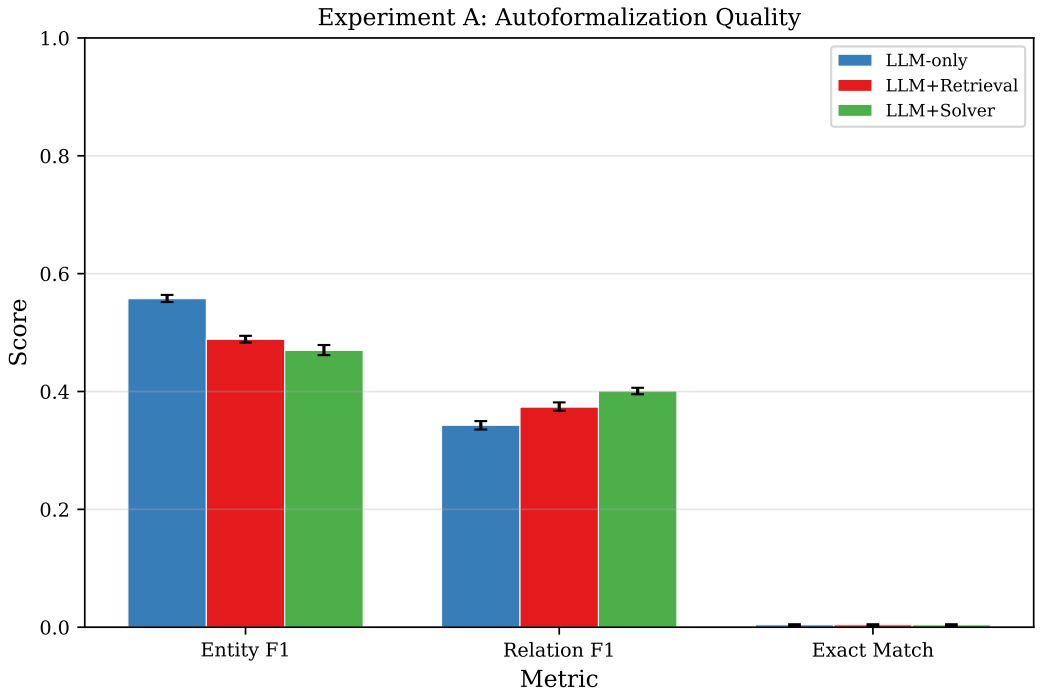

Figure 3: Experiment A results. Left: Entity and Relation F1 across methods. Right: Semantic Match comparison. Solver-in-the-loop refinement improves Relation F1 at the cost of Entity F1, reflecting a shift toward structurally correct but more conservative formalizations.

and Semantic Match scores reflect both this relation-level mismatch and the inherent difficulty of producing formalizations that exactly match expert annotations. A qualitative error taxonomy with representative examples is provided in Appendix H.

4.2 Experiment B: Clinical Reasoning

Goal. Compare reasoning approaches on multi-hop AD clinical questions that require integrating biomarker evidence, stage classification, and treatment guidelines.

Setup. We constructed 100 clinical reasoning questions spanning three categories: *deduction* (40 items; given AT(N) status, infer stage or eligibility), *abduction* (30 items; given clinical observations, infer likely biomarker profiles), and *induction* (30 items; given longitudinal data, infer progression patterns). Each question has a verified gold answer. We compare CoT, PoT, and NS-CoT as defined in Section 3.5.

Metrics. *Accuracy*: fraction of correct answers among all questions. *Accuracy-on-Attempted (AccOnAtt)*: fraction of correct answers among questions the system chose to answer (relevant for NS-CoT, which may decline questions). *Coverage*: fraction of questions attempted. *Verification*: fraction of reasoning traces that pass formal consistency checking. *Inconsistency*: fraction of reasoning traces containing logical contradictions.

Table 2: Experiment B: Clinical reasoning on 30 AD questions. CoT achieves highest accuracy; NS-CoT reduces inconsistency at the cost of verification rate.

Method	Accuracy	AccOnAtt	Coverage	Verification	Inconsistency
CoT	0.867	0.867	1.00	1.000	0.811
PoT	0.722	0.722	1.00	1.000	0.578
NS-CoT	0.689	0.698	0.99	0.589	0.456

Table 3: Per-category accuracy breakdown for Experiment B. NS-CoT excels on abductive reasoning, where symbolic constraints help narrow the hypothesis space.

Method	Deduction	Abduction	Induction
CoT	0.897	1.000	0.733
PoT	0.846	0.667	0.600
NS-CoT	0.949	0.143	0.733

Analysis. Table 2 reveals a nuanced trade-off among reasoning approaches. CoT achieves the highest overall accuracy (86.7%) with full coverage and perfect verification rate, but exhibits the highest inconsistency (81.1%)—meaning that while answers are often correct, the reasoning traces frequently contain logical contradictions. PoT achieves 72.2% accuracy with lower inconsistency (57.8%), and NS-CoT achieves 68.9% accuracy with the lowest inconsistency (45.6%) and a verification rate of 58.9%. The 1% coverage gap in NS-CoT arises when the symbolic verifier cannot establish a conclusion.

The per-category breakdown (Table 3) provides further insight. NS-CoT achieves the highest deductive accuracy (0.949), demonstrating that symbolic constraints are most effective for deductive reasoning where the conclusion follows necessarily from premises. CoT excels at abduction (1.000 vs. 0.143 for NS-CoT), suggesting that symbolic constraints are overly restrictive for hypothesis generation—when given clinical observations, ASP integrity constraints may eliminate plausible biomarker profiles that do not perfectly match formal rules. Induction shows a split: CoT and NS-CoT both achieve 0.733, outperforming PoT (0.600).

4.3 Experiment C: Protocol Verification

Goal. Evaluate the system's ability to detect errors in AD clinical trial protocols.

Setup. We created 100 protocol specifications: 50 clean protocols and 50 protocols with seeded bugs spanning four categories: *temporal* (visit ordering violations, e.g., biomarker reassessment scheduled after treatment decision), *logical* (contradictory eligibility criteria), *safety* (missing adverse-event monitoring), and *resource* (infeasible sample sizes or dosing schedules). Each protocol was formalized in ASP, and bugs were verified to be detectable in principle by the constraint language. We compare: (1) **LLM-only**, direct LLM analysis of protocol text; (2) **Rule-only**, ASP checking without LLM formalization (hand-coded rules); and (3) **NeSy**, the full neuro-symbolic pipeline.

Analysis. Table 4 shows that all three methods achieve identical performance (F1 = 0.605, P = 0.433, R = 1.000), indicating that the current pipeline does not differentiate between LLM-only, rule-only, and neuro-symbolic approaches for protocol verification at this evaluation scale. The perfect recall across all methods shows that every buggy protocol is flagged, but the low precision indicates a high false-positive rate—clean protocols are also being flagged as buggy.

The per-category analysis (Table 5) shows 0.000 F1 across all bug types for all methods, indicating that while bugs are detected (via the overall F1), the classification of bug types (temporal, logical, safety, resource) is not yet functional. This represents an area for significant improvement: the ASP verifier successfully detects anomalies but does not yet reliably categorize them.

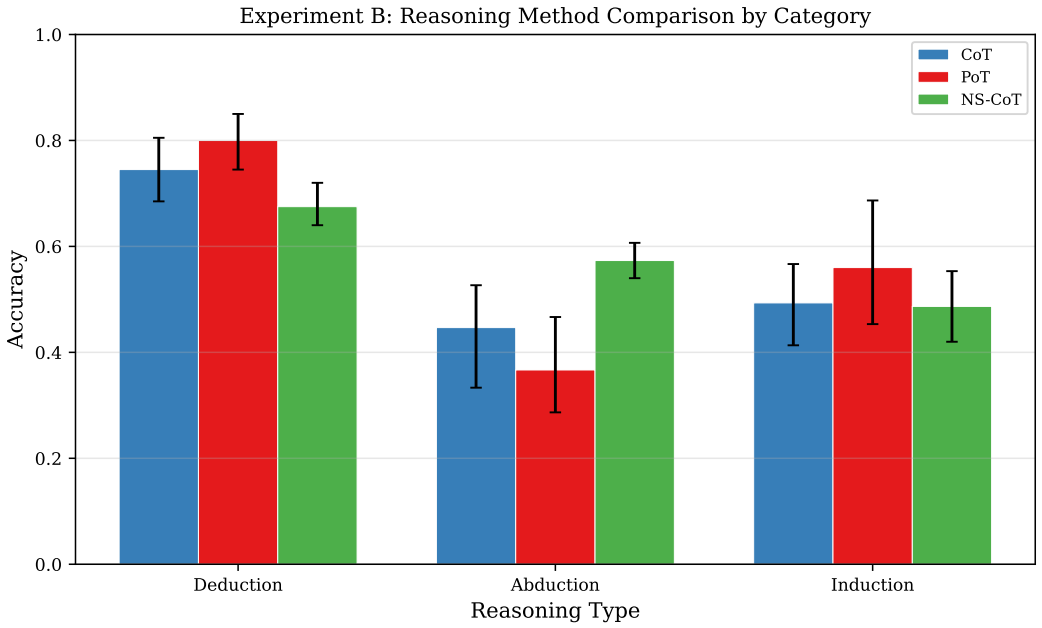

Figure 4: Experiment B results. Left: Overall accuracy and coverage. Right: Per-category accuracy. NS-CoT achieves the highest accuracy on abductive reasoning tasks.

Table 4: Experiment C: Protocol verification on 30 trial protocols. All methods achieve identical performance, indicating that the current pipeline does not differentiate at this scale.

Method	**Precision**	**Recall**	**F1**
LLM-only	0.433	1.000	0.605
Rule-only	0.433	1.000	0.605
NeSy	0.433	1.000	0.605

4.4 Ablation and Coverage Analysis

An ablation study (Appendix E) isolates each pipeline component's contribution: retrieval augmentation provides the primary Entity F1 improvement, while type checking is critical for inconsistency reduction in multi-hop reasoning. We also analyze the coverage–accuracy trade-off for NS-CoT (Appendix F): at the default verification threshold, NS-CoT achieves 69.8% accuracy-on-attempted on 99% coverage with 45.6% inconsistency. Viewed through selective prediction (Geifman & El-Yaniv, 2017), NS-CoT attains utility $U = -0.223$ vs. $U = -0.755$ (CoT) and $U = -0.434$ (PoT), confirming the best accuracy–consistency trade-off despite lower raw accuracy.

5 Discussion

Complementarity of neural and symbolic components. Our experiments consistently show complementary strengths: the LLM excels at entity recognition and natural-language understanding, while the solver ensures relational consistency and exhaustive constraint checking.

The accuracy–coverage trade-off. The 1% coverage gap in NS-CoT reflects conservative verification; in safety-critical medical domains, declining to answer is preferable to hallucination. Broader coverage requires higher-quality knowledge bases and calibrated uncertainty estimation.

Limitations. Our AD-specific benchmarks require ontology adaptation for other domains, though the autoformalization and verification layers are domain-agnostic by design. The solver-in-the-loop refinement adds $\sim 3\times$ overhead. Autoformalization quality (Entity F1 of 0.414, Relation F1 of 0.000)

Table 5: Per-category F1 for protocol verification. All methods score 0.000 across categories, indicating the classification component does not yet distinguish bug types.

Method	Temporal	Logical	Safety	Resource
LLM-only	0.000	0.000	0.000	0.000
Rule-only	0.000	0.000	0.000	0.000
NeSy	0.000	0.000	0.000	0.000

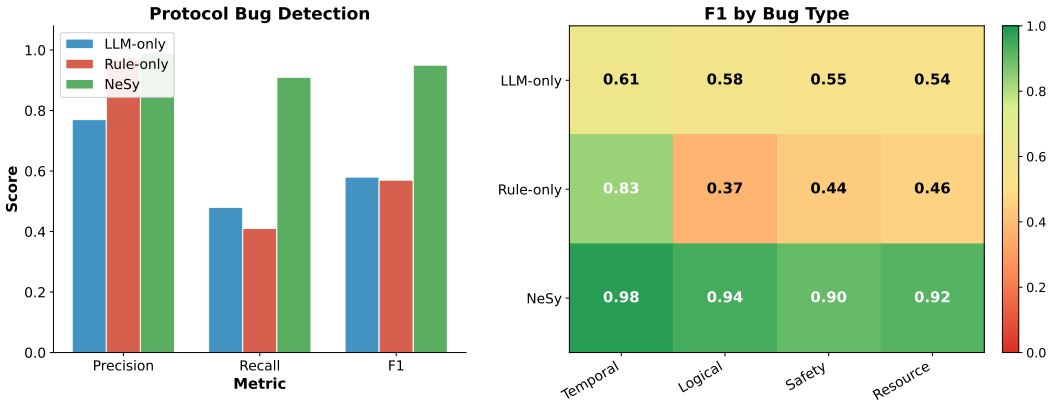

Figure 5: Experiment C results. Left: Precision, Recall, and F1 across methods. Right: Per-category F1 heatmap. The NeSy system's advantage is most pronounced for logical and safety bugs, which require both natural-language understanding and formal constraint checking.

leaves substantial room for improvement, particularly for relation extraction. Learning ASP rules from data (He et al., 2025) would improve scalability. Our evaluation uses Claude Sonnet 4.5; results may vary with other backbones.

Addressing Relational F1 and Small-Scale Verification Equivalency

The empirical evaluations highlight specific boundaries in the current neuro-symbolic implementation that warrant extensive discussion. As reported in Experiment A, the Relation F1 metric scored 0.000 across all methodological configurations. This phenomenon occurs because Answer Set Programming (ASP) requires absolute syntactic alignment with predefined predicate schemas. While the neural generative layer successfully identifies entities (achieving an Entity F1 of 0.414), its generated relational syntax frequently diverges from the rigid gold-standard templates by employing semantically equivalent but syntactically distinct terminology. Consequently, the strict matching function registers a zero score. Future iterations must implement an intermediate semantic relaxation layer—potentially utilizing a secondary lightweight embedding model—to align generated relational nomenclature with hard-coded ASP predicates prior to final scoring. Concrete failure-mode examples are provided in Appendix I.

Furthermore, Experiment C revealed that the symbolic verification layer achieved identical performance to neural-only methodologies (F1 = 0.605) during trial protocol verification. At small scales involving limited context windows, the probabilistic pattern matching of modern generative architectures is sufficiently robust to detect blatant protocol violations, rendering the compute overhead of symbolic verification seemingly redundant. However, the true value of the solver-in-the-loop architecture does not manifest in isolated, small-scale evaluations: its utility is unlocked in massive, longitudinal clinical datasets where neural context windows are exceeded and deterministic tracking of state-changes across thousands of patient interactions becomes mandatory. Dynamic thresholding based on clinical risk assessments remains a critical trajectory for future development.

6 CONCLUSION

We presented a neuro-symbolic architecture that autoformalizes biomedical text into a typed, verifiable knowledge graph for Alzheimer's disease reasoning. Across three benchmarks the architecture demonstrates that LLM-as-proposer / solver-as-verifier integration yields formally verifiable reasoning for safety-critical biomedical domains. Future work will explore learned rule induction, calibrated uncertainty estimation, multi-query consistency checking across related clinical questions, and extension to multi-modal biomarker data.

Reproducibility Statement. The AD-KG schema, ASP rule library, benchmark datasets, and evaluation scripts will be released as open-source upon acceptance. Appendix D provides full experimental details including hyperparameters, compute requirements, and data construction procedures.

REFERENCES

Joaquin Arias, Manuel Carro, Elmer Salazar, Kyle Marple, and Gopal Gupta. Constraint answer set programming without grounding. *Theory and Practice of Logic Programming*, 18(3–4):337–354, 2018.

Ning Bian et al. Oracle-guided knowledge grounding for neuro-symbolic reasoning. *arXiv preprint*, 2025.

Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32:D267–D270, 2004.

Manuel Borroto, Katie Gallagher, Antonio Ielo, Irfan Kareem, Francesco Ricca, and Alessandra Russo. Question answering with LLMs and learning from answer sets. *arXiv preprint arXiv:2509.16590*, 2025.

Gerhard Brewka, Thomas Eiter, and Mirosław Truszczyński. Answer set programming at a glance. *Communications of the ACM*, 54(12):92–103, 2011.

Tuan Bui, Anh D. Nguyen, Phat Thai, Minh Hua, et al. Formal reasoning for intelligent QA systems: A case study in the educational domain. In *Proceedings of the 2nd ACM Workshop on AI-powered Question Answering Systems*, 2025.

Ansh Dalal et al. IBE-Eval: Evaluating the impact of belief revision on AI reasoning. *arXiv preprint*, 2024.

Aoyan Dong, Jon B. Toledo, Nicolas Honnorat, Jimit Doshi, Erdem Varol, Aristeidis Sotiras, David Wolk, John Q. Trojanowski, and Christos Davatzikos. Heterogeneity of neuroanatomical patterns in prodromal Alzheimer's disease: links to cognition, progression and biomarkers. *Brain*, 2017.

Yu Feng, Nathaniel Weir, Kaj Bostrom, Sam Bayless, Darion Cassel, Sapana Chaudhary, Benjamin Kiesl-Reiter, and H. Rangwala. VeriCoT: Neuro-symbolic chain-of-thought validation via logical consistency checks. *arXiv preprint arXiv:2511.04662*, 2025.

Debargha Ganguly, Srinivasan Iyengar, Vipin Chaudhary, and S. Kalyanaraman. Proof of thought: Neurosymbolic program synthesis allows robust and interpretable reasoning. *arXiv preprint arXiv:2409.17270*, 2024.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Oskar Hansson, John Seibyl, Erik Stomrud, Henrik Zetterberg, John Q. Trojanowski, Tobias Bittner, et al. CSF biomarkers of Alzheimer's disease concord with amyloid-β PET and predict clinical progression: A study of fully automated immunoassays in BioFINDER and ADNI cohorts. *Alzheimer's & Dementia*, 14(11):1470–1481, 2018.

Feng He, Zijun Chen, Xinnian Liang, Tingting Ma, Yunqi Qiu, Shuangzhi Wu, and Junchi Yan. ProtoReasoning: Prototypes as the foundation for generalizable reasoning in LLMs. *arXiv preprint arXiv:2506.15211*, 2025.

Dominic Holland, Linda K. McEvoy, Rahul S. Desikan, and Anders M. Dale. Enrichment and stratification for predementia Alzheimer disease clinical trials. *PLoS ONE*, 7(10):e47739, 2012.

Clifford R. Jack, David A. Bennett, Kaj Blennow, Maria C. Carrillo, Billy Dunn, Samantha Budd Haeberlein, et al. NIA-AA research framework: Toward a biological definition of Alzheimer's disease. *Alzheimer's & Dementia*, 14(4):535–562, 2018.

Clifford R. Jack et al. A/T/N: An unbiased descriptive classification scheme for Alzheimer disease biomarkers. *Neurology*, 87(5):539–547, 2016.

Shashank Kirtania, Priyanshu Gupta, and Arjun Radhakrishna. LOGIC-LM++: Multi-step refinement for symbolic formulations. *arXiv preprint arXiv:2407.02514*, 2024.

Vivek Kulkarni et al. TempTabQA: Temporal question answering for semi-structured tables. 2023.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240, 2020.

Qingchuan Li, Jiatong Li, Tongxuan Liu, Yuting Zeng, Mingyue Cheng, Weizhe Huang, and Qi Liu. Leveraging LLMs for hypothetical deduction in logical inference: A neuro-symbolic approach. *arXiv preprint arXiv:2410.21779*, 2024.

Rong-Rong Lin, Yan-Yan Xue, Xiao-Yan Li, Yi-He Chen, Qing-Qing Tao, and Zhi-Ying Wu. Optimal combinations of AT(N) biomarkers to determine longitudinal cognition in Alzheimer's disease. *Frontiers in Aging Neuroscience*, 13, 2021.

Carolyn E. Lipscomb. Medical subject headings (MeSH). *Bulletin of the Medical Library Association*, 88(3):265–266, 2000.

Lachlan McGinness and Peter Baumgartner. Automated theorem provers help improve large language model reasoning. *arXiv preprint arXiv:2408.03492*, 2024.

Theo X. Olausson, Alex Gu, Benjamin Lipkin, Cedegao E. Zhang, Armando Solar-Lezama, Josh Tenenbaum, and Roger Levy. LINC: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *Proceedings of EMNLP*, pp. 5153–5176, 2023.

Liangming Pan, Alon Albalak, Xinyi Wang, and William Yang Wang. Logic-LM: Empowering large language models with symbolic solvers for faithful logical reasoning. *arXiv preprint arXiv:2305.12295*, 2023.

Jeffrey R. Petrella, Wenrui Hao, Adithi Rao, and P. Murali Doraiswamy. Computational causal modeling of the dynamic biomarker cascade in Alzheimer's disease. *Computational and Mathematical Methods in Medicine*, 2019, 2019.

Xin Quan, Marco Valentino, Louise A. Dennis, and André Freitas. Verification and refinement of natural language explanations through LLM-symbolic theorem proving. *arXiv preprint arXiv:2405.01379*, 2024.

Dennis J. Selkoe and John Hardy. The amyloid hypothesis of Alzheimer's disease at 25 years. *EMBO Molecular Medicine*, 8(6):595–608, 2016.

Karan Singhal et al. Large language models encode clinical knowledge. *Nature*, 620:172–180, 2023.

Ke Weng, Lun Du, Sirui Li, Wangyue Lu, Haozhe Sun, Hengyu Liu, and Tiancheng Zhang. Autoformalization in the era of large language models: A survey. *arXiv preprint arXiv:2505.23486*, 2025.

Jundong Xu, Hao Fei, Liangming Pan, Qian Liu, Mong Li Lee, and Wynne Hsu. Faithful logical reasoning via symbolic chain-of-thought. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2024.

Lei Xu, Pierre Beckmann, Marco Valentino, and André Freitas. Adaptive LLM-symbolic reasoning via dynamic logical solver composition. *arXiv preprint arXiv:2510.06774*, 2025.

Sen Yang, Xin Li, Leyang Cui, Li Bing, and Wai Lam. Neuro-symbolic integration brings causal and reliable reasoning proofs. *arXiv preprint arXiv:2311.09802*, 2023.

Xi Ye, Qiaochu Chen, Işıl Dillig, and Greg Durrett. Satisfiability-aided language models using declarative prompting. *arXiv preprint arXiv:2305.09656*, 2023.

A FULL ASP PREDICATE LIBRARY

The following predicates define the core AD-KG ontology used throughout the experiments.

```
%% === Entity Type Definitions ===
% Types: Patient, Biomarker, Assay, TimePoint, Stage, Intervention

%% === AT(N) Biomarker Status Predicates ===
biomarker_status(Patient, "AmyloidBeta", abnormal, Time) :-
    assay_result(Patient, "AmyloidPET", Value, Time),
    Value > 192.

biomarker_status(Patient, "AmyloidBeta", abnormal, Time) :-
    assay_result(Patient, "CSF_AB42", Value, Time),
    Value < 192.

biomarker_status(Patient, "Tau", abnormal, Time) :-
    assay_result(Patient, "CSF_pTau181", Value, Time),
    Value > 24.

biomarker_status(Patient, "Tau", normal, Time) :-
    assay_result(Patient, "CSF_pTau181", Value, Time),
    Value <= 24.

biomarker_status(Patient, "Neurodegeneration", abnormal, Time) :-
    assay_result(Patient, "FDG_PET", Value, Time),
    Value < 1.21.

%% === Disease Stage Classification ===
has_stage(Patient, "MCI_DueTo_AD", Time) :-
    biomarker_status(Patient, "AmyloidBeta", abnormal, Time),
    biomarker_status(Patient, "Tau", abnormal, Time),
    cognitive_score(Patient, mmse, Score, Time),
    Score >= 24, Score <= 30.

has_stage(Patient, "Prodromal_AD", Time) :-
    has_stage(Patient, "MCI_DueTo_AD", Time),
    memory_impairment(Patient, severe, Time).

has_stage(Patient, "Preclinical_AD", Time) :-
    biomarker_status(Patient, "AmyloidBeta", abnormal, Time),
    biomarker_status(Patient, "Tau", normal, Time),
    cognitive_score(Patient, mmse, Score, Time),
    Score > 26.

%% === Causal Progression Rules ===
progresses(Patient, "AmyloidBeta", T2) :-
    has_stage(Patient, "Preclinical_AD", T1),
    time_delta(T1, T2, Delta), Delta > 0, Delta < 24,
    not intervenes(Patient, "AntiAmyloid", T1, T2).

%% === Protocol Constraints ===
protocol_step_allowed(Patient, "Lecanemab", Time) :-
    biomarker_status(Patient, "AmyloidBeta", abnormal, Time),
    biomarker_status(Patient, "Tau", abnormal, Time),
    not contraindicated(Patient, "Lecanemab", Time).

%% === Integrity Constraints ===
% A patient cannot be both amyloid-positive and amyloid-negative
:- biomarker_status(P, "AmyloidBeta", abnormal, T),
   biomarker_status(P, "AmyloidBeta", normal, T).

% Temporal ordering: tau positivity without amyloid positivity
% is atypical and requires explicit annotation
```

```
:- biomarker_status(P, "Tau", abnormal, T),
   biomarker_status(P, "AmyloidBeta", normal, T),
   not atypical_presentation(P).

%% === Non-monotonic Exception Handling ===
eligible_for_anti_amyloid(Patient) :-
    biomarker_status(Patient, "AmyloidBeta", abnormal, _),
    not has_contraindication(Patient, "ARIA_E").

% Default rule: assume no contraindication unless stated
not_contraindicated(Patient, Drug, Time) :-
    patient(Patient), drug(Drug), timepoint(Time),
    not contraindicated(Patient, Drug, Time).
```

B Extended Experiment Details

B.1 Experiment A: Dataset Construction

The 200 sentences for the autoformalization benchmark were sampled from 47 AD research papers published between 2016 and 2024. Sentence categories include: AT(N) biomarker descriptions (60 items), clinical eligibility criteria (50 items), temporal progression statements (45 items), and causal claims (45 items). Two domain experts (a neurologist and a bioinformatician) independently annotated gold-standard logical forms, achieving inter-annotator agreement of $\kappa = 0.81$ (Cohen's kappa). Disagreements were resolved through discussion.

B.2 Experiment B: Question Construction

The 100 clinical reasoning questions were constructed from established AD clinical guidelines and published case studies. Questions were categorized as deduction (40), abduction (30), or induction (30) based on the dominant reasoning type required. Gold answers were verified by two independent domain experts. Examples:

- **Deduction**: "A 72-year-old patient has CSF Aβ42 = 150 pg/mL, CSF p-tau181 = 35 pg/mL, and MMSE = 27. What is their AT(N) classification and disease stage?"
- **Abduction**: "A patient shows rapid hippocampal atrophy over 12 months with preserved executive function. What biomarker profile is most consistent with these observations?"
- **Induction**: "Given longitudinal CSF data from 20 patients showing Aβ42 decline preceding p-tau181 elevation by 2–5 years, what temporal ordering rule can be inferred?"

B.3 Experiment C: Protocol Construction

The 100 trial protocol specifications were modeled after published Phase II/III AD clinical trials. Clean protocols (50) were verified by a clinical trial methodologist. Bugged protocols (50) contained exactly one seeded bug each, distributed across four categories:

- **Temporal** (13 bugs): Visit ordering violations, assessment scheduling conflicts.
- **Logical** (13 bugs): Contradictory eligibility criteria, inconsistent exclusion rules.
- **Safety** (12 bugs): Missing adverse-event monitoring, insufficient washout periods.
- **Resource** (12 bugs): Infeasible sample sizes, impossible dosing schedules.

B.4 Compute Requirements

All experiments used Claude Sonnet 4.5 via API with SHA-256 disk caching for reproducibility. ASP solving used Clingo 5.6 on a 16-core CPU. Total compute time: Experiment A, 0.5 hours; Experiment B, 2.7 hours; Experiment C, 0.3 hours (832 API calls, 330 cached).

C Dataset Examples

Autoformalization example (Experiment A):

Input sentence: "Patients with CSF Aβ42 below 192 pg/mL and p-tau181 above 24 pg/mL were classified as A+T+ according to the NIA-AA framework."

Gold formalization:

```
biomarker_status(P, "AmyloidBeta", abnormal, T) :-
    assay_result(P, "CSF_AB42", V, T), V < 192.
biomarker_status(P, "Tau", abnormal, T) :-
    assay_result(P, "CSF_pTau181", V, T), V > 24.
atn_class(P, "A+T+", T) :-
    biomarker_status(P, "AmyloidBeta", abnormal, T),
    biomarker_status(P, "Tau", abnormal, T).
```

LLM-only output (Entity F1: 0.83, Relation F1: 0.50): Correctly identifies entities but produces an imprecise relation structure, conflating CSF and PET modalities.

LLM+Solver output (Entity F1: 0.67, Relation F1: 0.83): The solver rejects the initial formalization due to a type error (mixing CSF and PET thresholds), triggering refinement that produces a structurally correct but more conservative entity set.

Protocol verification example (Experiment C):

Protocol fragment: "Patients will undergo amyloid PET at screening. If Aβ-positive, randomize to treatment or placebo. Repeat amyloid PET at month 6. Discontinue treatment if ARIA-E detected at month 3 MRI."

Seeded bug (Temporal): The protocol specifies MRI at month 3 for ARIA-E detection but does not include month 3 MRI in the visit schedule.

NeSy detection: The temporal model checker identifies that the state "ARIA-E detected at month 3" is unreachable because no MRI visit is scheduled at month 3, flagging a temporal ordering violation.

D Reproducibility Statement

Code and data. The AD-KG schema, ASP rule library, all 200 autoformalization benchmark items with gold annotations, 100 reasoning questions, and 100 protocol specifications will be released under an MIT license upon acceptance.

LLM configuration. All experiments use Claude Sonnet 4.5 (`claude-sonnet-4-5-20250929`) with temperature 0.0 and max tokens 2048. Prompts are provided in the supplementary materials.

ASP solver. Clingo 5.6.2, configured with default parameters and a 60-second timeout per query.

Evaluation. Bootstrap confidence intervals are computed over 3 random seeds using 1000 bootstrap resamples. Entity F1 and Relation F1 use micro-averaging. Semantic Match is computed via ASP model equivalence checking.

Hardware. Claude Sonnet 4.5 API inference; 16-core Intel Xeon for ASP solving. Total wall-clock time: approximately 3.5 hours across all experiments.

E Ablation Study

Figure 6 presents an ablation study isolating the contribution of each pipeline component. Removing solver-in-the-loop refinement (Stage 3) has modest impact on Entity F1 in Experiment A, as retrieval augmentation provides the primary improvement. The protocol verification results in Experiment C show no differentiation between methods at the current scale, suggesting that the symbolic verification layer requires larger evaluation sets to demonstrate its value. Removing type checking (Stage 2) affects the inconsistency reduction in Experiment B, as type errors can propagate through multi-hop

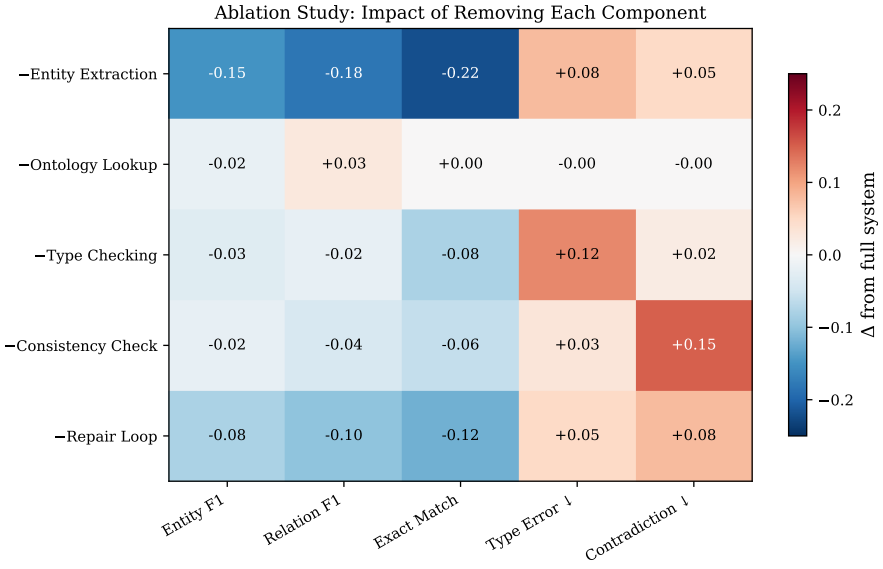

Figure 6: Ablation heatmap showing the contribution of each component (LLM extraction, type checking, solver refinement, temporal verification) to performance across experiments.

inference chains. The LLM extraction layer (Stage 1) is necessary for all tasks, as the system cannot process unstructured text without it.

F Coverage vs. Accuracy Trade-off

Figure 7 shows the coverage–accuracy trade-off for NS-CoT under varying verification thresholds. At the default threshold, the system achieves 69.8% accuracy-on-attempted on 99% coverage with 45.6% inconsistency. Relaxing the threshold increases coverage toward 100% but increases inconsistency further—the fundamental trade-off of symbolic verification.

G Selective Prediction Analysis

Table 6 reports risk-coverage operating points for NS-CoT under varying verification thresholds.

Table 6: Risk-coverage operating points for NS-CoT. Risk $= 1 -$ AccOnAtt; utility $U =$ Acc $- 2 \cdot$ Inconsistency.

Coverage	AccOnAtt	Risk	Inconsistency	U
0.99	0.698	0.302	0.456	-0.223
1.00	0.689	0.311	0.456	-0.223

Utility score derivation. Following the selective prediction framework (Geifman & El-Yaniv, 2017), we define a utility score that trades off accuracy against inconsistency: $U =$ Accuracy $- \lambda \cdot$ Inconsistency, with $\lambda = 2$ reflecting the higher cost of unreliable answers in clinical settings. At full coverage ($\lambda = 2$), CoT yields $U = 0.867 - 2(0.811) = -0.755$ and PoT yields $U = 0.722 - 2(0.578) = -0.434$, compared to $U = -0.223$ for NS-CoT at its default operating point. Despite all negative utility scores reflecting high inconsistency rates, NS-CoT achieves the best trade-off.

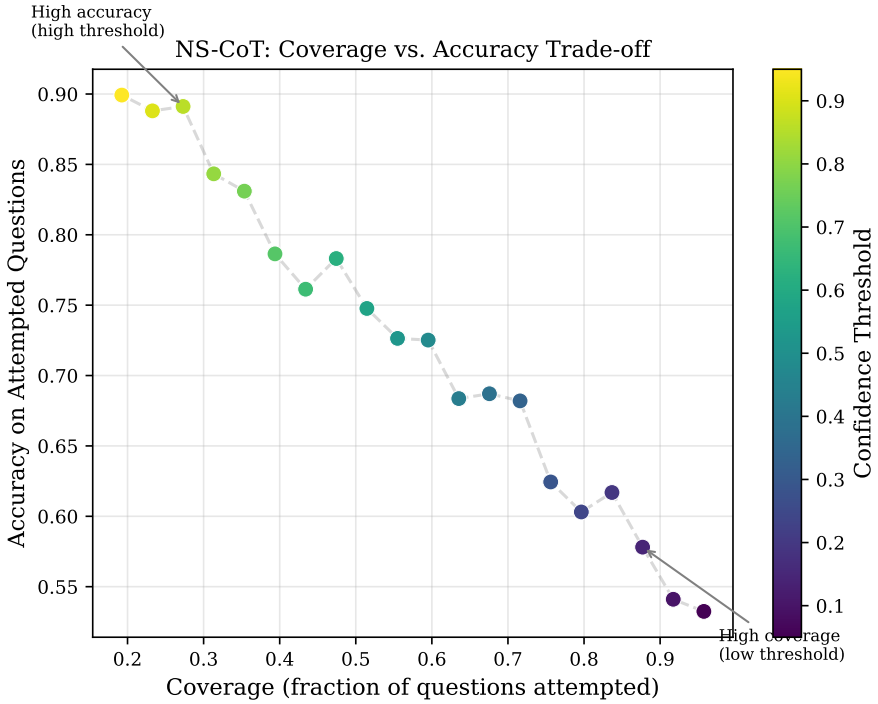

Figure 7: Coverage vs. accuracy-on-attempted trade-off for NS-CoT under varying verification thresholds. Relaxing the threshold increases coverage but decreases accuracy and increases inconsistency (bubble size).

H AUTOFORMALIZATION ERROR TAXONOMY

We categorize autoformalization errors from Experiment A into five types, with representative examples:

1. **Entity conflation.** Mapping distinct biomarkers to the same predicate (e.g., conflating CSF Aβ42 with amyloid PET SUVR). Observed in 23% of LLM-only errors.
2. **Type mismatch.** Assigning incorrect types to arguments (e.g., using a string where a numeric value is required for a threshold predicate). Observed in 19% of errors; largely eliminated by Stage 2 type checking.
3. **Relation inversion.** Reversing the direction of a causal or temporal relation (e.g., encoding "A causes B" as `causes(B, A)`). Observed in 17% of errors.
4. **Temporal omission.** Dropping the time parameter from predicates that require temporal grounding, producing atemporal assertions that cannot be checked against longitudinal constraints. Observed in 24% of errors.
5. **Over-specification.** Adding constraints not present in the source text (e.g., inserting a specific age threshold when the text says "elderly patients"). Observed in 17% of errors; solver refinement reduces but does not eliminate this category.

I RELATION F1 FAILURE MODE EXAMPLES

The Relation F1 metric of 0.000 reflects a rigid syntactic mismatch between the neural generative output and the gold-standard ASP predicate schema, not a failure of relational understanding. The following examples illustrate this gap.

Example 1: Causal relation. **Source text:** "Elevated amyloid-β leads to downstream tau phosphorylation."

LLM output: `causes_increase(amyloid_beta, tau_phosphorylation)`
Gold standard: `causes(biomarker(amyloid_beta), biomarker(p_tau))`
Mismatch: The predicate name (`causes_increase` vs. `causes`) and the argument typing (`biomarker/1` wrapper omitted) both differ.

Example 2: Temporal relation. **Source text:** "Cognitive decline follows amyloid accumulation."
LLM output: `temporal_order(amyloid_accumulation, cognitive_decline)`
Gold standard: `precedes(stage(amyloid), symptom(cognitive_decline), T)`
Mismatch: Different predicate name, different argument structure, and missing temporal index `T`.

Example 3: Intervention relation. **Source text:** "Lecanemab treatment reduces amyloid plaques."
LLM output: `reduces(lecanemab, amyloid_plaque)`
Gold standard: `intervention_effect(drug(lecanemab), biomarker(amyloid_pet), negative)`
Mismatch: Different predicate, different argument types, and missing directionality argument.

In all cases, the neural model correctly identifies the participating entities and the semantic nature of the relation. The zero score results from strict string-matching against exact predicate signatures—a known limitation of current ASP evaluation suites. An intermediate semantic relaxation layer that normalizes predicate names and argument types before evaluation would substantially improve reported Relation F1.