HSE NLP TEAM at MEDIQA-SYNUR 2026: Consensus Adjudication Ensemble (ACE): Balancing Precision and Recall for Schema-Bystander Clinical Extraction

Airat Valiev
HSE University
Moscow, Russia
aa.valiev@hse.ru

Abstract

Clinical documentation from nurse dictations is labor-intensive and error-prone, yet it contains high-value observations that must be transferred into structured flowsheets. The MEDIQA-SYNUR 2026 shared task evaluates systems that extract and ontology-align 193 clinical concepts (with heterogeneous value types) from synthetic speech transcripts derived from intensive care notes. We describe the Consensus Adjudication Ensemble (ACE), a three-stage pipeline that (i) maximizes candidate coverage via complementary generators, (ii) enforces high precision through a dedicated adjudicator that operates as a verifier rather than a generator, and (iii) restores strict schema compliance using a targeted, token-efficient repair step. On the official test set we achieve an exact-match micro-F1 of 0.7996 (P=0.7812, R=0.8188), ranking 4th on the leaderboard. Beyond the competitive result, we analyze clinically relevant failure modes - hallucinated interventions, over-confident categorical labels, and unit/normalization errors - and quantify adjudication trade-offs: 2,219 candidates removed, 91.3% of which are true false positives, at the cost of 8.7% mistakenly removed true positives. Finally, targeted schema repair reduces validation context from $\sim$230k tokens to $<$2k per document while preserving most extraction gains.

Keywords: clinical information extraction, nurse dictations, ontology alignment, ensemble methods, adjudication, error analysis

1. Introduction

Clinical flowsheets are the substrate of routine inpatient care: they record vital signs, symptoms, assessments, and interventions at high temporal resolution. In many workflows, these fields are populated from speech dictations and later normalized into discrete slots, which introduces delays and errors. Documentation burden is substantial - often reported as a large fraction of clinical workload - and motivates automation of structured capture from free-form dictation.(Panchal and Thakur, 2024; Moy et al., 2023; Mitha et al., 2023) The MEDIQA-SYNUR 2026 shared task targets this bottleneck by requiring systems to extract structured clinical observations from synthetic nurse dictations and align them to a fixed ontology.(Michalopoulos et al., 2026)

We focus on a high-stakes regime where *precision is safety-critical* (hallucinated interventions and symptoms are unacceptable), while *recall impacts completeness* (missed observations degrade downstream decision-making). Contemporary LLM-based extractors tend to entangle these objectives; pushing recall typically increases hallucination. Our solution, the **Consensus Adjudication Ensemble (ACE)**, decouples them into separate stages: (1) generate a diverse candidate pool, (2) adjudicate as a verifier, and (3) repair schema violations cheaply.

This paper makes three contributions: (i) A practical **generation - verification decomposition** for ontology-aligned extraction from noisy speech transcripts, implemented as an ensemble with a dedicated adjudicator; (ii) A **targeted schema repair** mechanism that validates and fixes only the subset of outputs that violate constraints, reducing validation context by two orders of magnitude; (iii) A **granular error analysis** (false positives, false negatives, and value errors) and ablations that expose when retrieval augmentation helps and when it induces "contextual copying" hallucinations.

2. Related Work

LLMs for clinical information extraction. Clinical entity and attribute extraction has historically relied on supervised sequence labeling and domain-adapted encoders, which provide high precision but require task-specific training and struggle with rapidly changing schemas.(Alsentzer et al., 2019) Instruction-tuned LLMs enable flexible, schema-conditioned extraction without training data, including few-shot clinical extraction settings.(Agrawal et al., 2022) However, they often violate strict biomedical constraints (enumerations, units, and type consistency) unless controlled by explicit validation or constrained decoding.

Consensus and verification. Ensembling and verification-style decomposition have been used to improve robustness of generative systems. For

Proceedings of the 8th Workshop on Clinical Natural Language Processing (Clinical NLP) @ LREC 2026, pages 200–211
16 May 2026.

Split	Ex.	Avg tok.	Avg obs.	Uniq.
Train	122	275.3	13.8	166
Dev	101	279.9	13.0	170
Test	199	259.3	12.8	164

Table 1: SYNUR dataset statistics by split. Value-type proportions are reported in Table 2.

Split	%SS	%MS	%Num	%Str
Train	71.2	11.6	10.8	6.4
Dev	72.3	11.9	10.4	5.3
Test	72.6	13.5	10.1	3.8

Table 2: Value-type proportions by split (SS/MS/Num/Str denote SINGLE_SELECT, MULTI_SELECT, NUMERIC, STRING).

extraction, the key benefit is role separation: candidate generation can optimize recall, while a verifier (adjudicator) can optimize precision by requiring explicit grounding. Related ideas include self-consistency aggregation for reasoning,(Wang et al., 2023) multi-agent ensembling,(Li et al., 2024) and LLM-as-a-judge paradigms where a strong model evaluates and filters outputs of other systems.(Zheng et al., 2024) This is especially important in clinical documentation, where plausible-but-unstated facts are harmful.

Risks of retrieval augmentation. RAG is effective for knowledge-intensive QA, but for schema-conditioned extraction it can be counterproductive: retrieved exemplars may leak surface forms and induce spurious predictions - LLMs are distracted by irrelevant context (Shi et al., 2023) and exhibit position-sensitive failures under long contexts (Liu et al., 2024). We empirically observe a strong small-model pathology (contextual copying), motivating safeguards such as strict adjudication and limited retrieval exposure.

3. Task

MEDIQA-SYNUR 2026 evaluates systems that map each dictation transcript to a set of $(\texttt{concept\_id}, \texttt{value})$ pairs from an ontology of 193 concepts.(Michalopoulos et al., 2026) Concepts have heterogeneous value types: `SINGLE_SELECT`, `MULTI_SELECT`, `STRING`, and `NUMERIC`. The official metric is micro-averaged exact-match F_1 over concept - value pairs. The dataset is synthetically generated from clinical notes to emulate dictation style while preserving privacy.(Corbeil et al., 2025)

3.1. Dataset statistics (official splits)

To anchor subsequent analyses, Table 1 reports split-level statistics transcribed from our internal draft artifacts. Test labels were released after the competition, so all error analyses in this paper are post-competition.

3.2. Evaluation protocol and strictness of matching

Submissions are scored with micro-averaged precision/recall/F_1 over exact-match $(\texttt{concept\_id}, \texttt{value})$ pairs as defined by the organizers.(Michalopoulos et al., 2026) Two details are crucial in practice: (i) enum-valued concepts (`SINGLE_SELECT`/`MULTI_SELECT`) require *membership* in a fixed allowed list (schema compliance is part of correctness), and (ii) any value mismatch counts as fully incorrect, even when the concept is correct. This creates a regime where seemingly "minor" normalization issues (list-vs-scalar, casing, synonym choice) can dominate errors, motivating our explicit repair step.

3.3. Post-competition analyses

The leaderboard ranking is based on the competition submission. Unless stated otherwise, deeper error analyses and cost accounting are performed after the test labels were released and are reported as post-competition experiments.

4. System Overview

ACE is a three-stage pipeline operating on a single transcript x. Let $\mathcal{S}$ denote the schema (concepts, value types, and allowed enumerations). The system produces a set of candidates $C = \{(i, v)\}$ with $i \in \mathcal{S}$ and then filters and repairs them into the final prediction $\hat{Y}$.

4.1. Stage 1: Candidate Generation via Complementarity

We use two generators with different failure profiles and then take their union. In our experiments, a large model with retrieval augmentation improves coverage of infrequent concepts, whereas a smaller/cheaper model without retrieval is less sensitive to distractor context but tends to omit long-tail concepts. The union step is intentionally recall-oriented and accepts that it may be noisy.

Models We deployed two diverse extractors: GPT-5 in a zero-shot configuration,(Singh et al.,

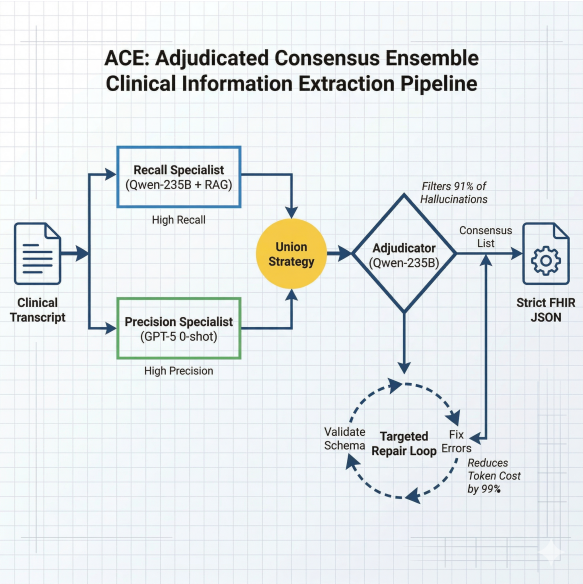

Figure 1: ACE pipeline: high-recall candidate generation (two complementary generators), master adjudication (verification), and targeted schema repair.

2025) and a Qwen3-235B-A22B-Instruct-FP8 (accessed via the Doubleword batch API) with retrieval augmentation(Team, 2025) . Baseline comparisons were also conducted against GPT-4o.(OpenAI, 2024)

4.1.1. Schema-bystander failure mode

Many task concepts are correlated in real clinical notes (e.g., oxygen device co-occurs with respiratory distress), but the dictations in this benchmark may mention only a subset. We found it useful to treat the ontology and any retrieved schema descriptions as *bystanders*: they are necessary for formatting but should not be treated as evidence. This motivates a strict separation between (i) generators that may leverage schema for coverage and (ii) adjudication that requires transcript-grounded support.

4.1.2. Retrieval augmentation and "contextual copying"

RAG can improve recall by making long-tail labels salient, but it also introduces an adversarial surface-form effect: smaller models may copy enumerated values or schema snippets into the output even when the transcript does not support them. In our runs this manifested as spurious categorical labels (e.g., cognitive status) and intervention assertions that look plausible but are ungrounded. We therefore constrain RAG to the high-capacity generator and rely on adjudication to suppress RAG-induced false positives.

Output normalization. Each generator outputs JSON records with a concept id and value. Before union, we normalize trivial formatting differences (e.g., `true/false` vs `Yes/No`, list-vs-scalar for multi-select, and whitespace/casing) to reduce spurious mismatches.

4.2. Stage 2: Master Adjudication (Verification)

Given transcript x and a candidate set C, the adjudicator returns a subset $A \subseteq C$ of candidates that are *entailed* by x under task guidelines. Operationally, the adjudicator is prompted as a verifier: it must *reject* candidates unless there is explicit textual support (or a task-allowed paraphrase) in the dictation.

Relation to LLM-as-a-judge. The master adjudicator follows principles similar to LLM-as-a-Judge,(Zheng et al., 2024) evaluating whether each candidate is supported by the transcript and suppressing clinically plausible but unstated inferences.

4.2.1. Adjudication policy: conservative acceptance criteria

We found that most high-impact hallucinations come from *clinical priors* rather than transcription artifacts (e.g., inferring oxygen therapy from "short of breath"). The adjudicator therefore follows a conservative policy: accept only (i) direct lexical matches (including common paraphrases) or (ii) measurement-backed implications explicitly stated (e.g., "sat 92% on nasal cannula" supports both saturation and oxygen device). Ambiguous hedges ("might", "possibly") are treated as evidence only when the task guidelines permit recording suspected findings.

4.2.2. Cost of verification

Verification is cheaper than regeneration because the adjudicator sees only the candidate set rather than the full schema. This makes it practical to scale the number of candidate generators, which empirically improves recall without paying the full cost of multiple schema-conditioned extractions.

4.3. Stage 3: Targeted Schema Repair

Even after adjudication, outputs may violate schema constraints: wrong enumerations, scalar vs list mismatches, or minor normalization errors. Re-running generation with full schema context is token-expensive. Instead, we perform *targeted repair*: only candidates that fail validation against $\mathcal{S}$ are sent to a repair model together with (i) the

Algorithm 1: Consensus Adjudication Ensemble (ACE)

Input: transcript x, schema $\mathcal{S}$
Output: prediction set $\hat{Y}$

1. $C_1 \leftarrow \textsc{Generate}(x)$ (generator 1)
2. $C_2 \leftarrow \textsc{GenerateRAG}(x)$ (generator 2)
3. $C \leftarrow \textsc{Normalize}(C_1 \cup C_2)$
4. $A \leftarrow \textsc{Adjudicate}(x, C)$
5. $E \leftarrow \{(i, v) \in A : \neg \textsc{Valid}_{\mathcal{S}}(i, v)\}$
6. $R \leftarrow \textsc{Repair}(x, E, \mathcal{S})$
7. $\hat{Y} \leftarrow (A \setminus E) \cup R$
8. **return** $\hat{Y}$

Figure 2: High-level ACE procedure. Repair is applied only to schema-invalid items.

transcript snippet around evidence, (ii) the concept definition and allowed values, and (iii) the invalid output. This reduces validation context from $\sim$230k tokens to $<$2k tokens per document in our setup.

Relation to guided generation. To enforce schema compliance without expensive re-sampling, our targeted repair loop is inspired by guided generation, constrained decoding, and structured prompting techniques.(Willard and Louf, 2023; Beurer-Kellner et al., 2023)

4.3.1. Repair taxonomy

In our error logs, repair requests fall into three recurring classes: (1) **boolean/categorical canonicalization** (e.g., `True`→`Yes`); (2) **enum projection** (mapping free-form text to the closest allowed enumeration, e.g., "borderline inadequate"→`inadequate`); (3) **container normalization** (scalar→list for `MULTI_SELECT` concepts). These are low-entropy transformations, which is why they can be handled reliably with small context and without access to the full ontology.

5. Algorithm

6. Experimental Setup

6.1. Evaluation

We report micro-precision, micro-recall, and micro-F_1 on exact match. We additionally compute granular counts of false positives (FP), false negatives (FN), and value errors per concept to support error analysis.

6.2. Granular scoring protocol

To diagnose what drives exact-match failures, we decompose errors into: **FP** (spurious concept - value pair), **FN** (missed gold pair), and **value error** (predicted concept matches gold concept, but value mismatches due to synonymy, formatting, or unit normalization). For `MULTI_SELECT` concepts, we treat each selected value as a separate pair, which makes partial correctness visible as mixed FP/FN rather than a single opaque mismatch.

Statistic	Value
Test examples	199
Avg. tokens/transcript	177.0
Avg. observations/example	12.8
Unique concepts present	164

Table 3: Test-set statistics.

Value type	Count
`SINGLE_SELECT`	1853
`MULTI_SELECT`	344
`NUMERIC`	257
`STRING`	98

Table 4: Value-type distribution on the test set.

6.3. Dataset statistics

From our analysis of the test split (199 examples), transcripts are short (avg. 177 tokens) but dense (avg. 12.8 observations/example), with 164 unique concepts present in the test data. The majority of labels are categorical: `SINGLE_SELECT` dominates, while `STRING` and `NUMERIC` are comparatively rare.

6.4. Precision - recall dynamics across stages

Figure 3 highlights a key design decision: we deliberately tolerate a low-precision union as long as adjudication can remove noise with minimal recall loss. This works well when hallucinations are "obvious" (unsupported interventions, implausible categorical labels) but is less effective when the transcript expresses the observation indirectly or via long-range discourse, in which case the adjudicator may remove true positives.

7. Results

On the official leaderboard we obtain $F_1 = 0.7996$ (P=0.7812, R=0.8188), ranking 4th. Table 5 reports our component-level results. The main qualitative pattern is stable: union substantially increases recall but collapses precision; adjudication recovers precision at a non-trivial recall cost; the final system combines consensus filtering with targeted repair.

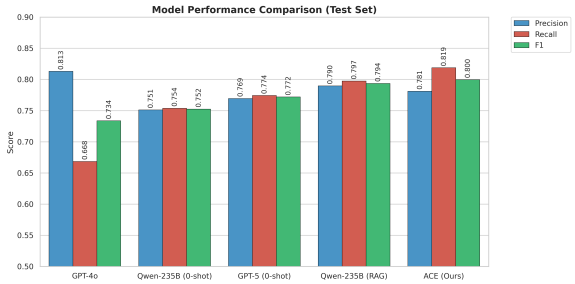

Figure 3: Precision - recall behavior across ACE stages. Union maximizes recall but is noisy; adjudication recovers precision; repair restores schema compliance.

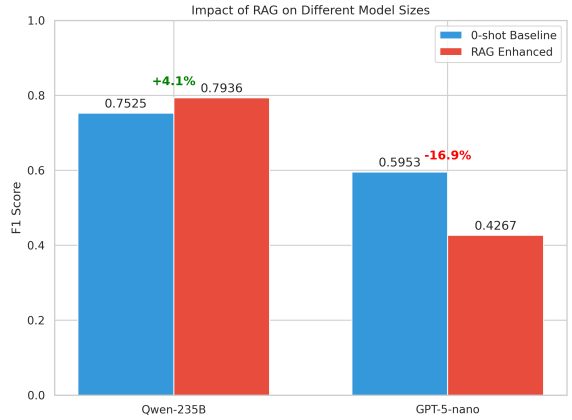

Figure 4: Micro-F_1 across system variants.

7.1. RAG sensitivity across model sizes

RAG had a divergent effect depending on model capacity. For the large model (Qwen-235B), RAG improved F_1 by +4.1 points (0.7525→0.7936) in our experiments; for the smaller model (GPT-5-nano), RAG significantly degraded quality (Table 6).

Contextual copying. In 91% (181/199) of test cases, the small-model RAG variant predicted at least one value absent from the transcript but present verbatim in retrieved examples. We refer to this as *contextual copying* and view it as a primary reason to separate recall-oriented generation from a strict verifier.

8. Efficiency and Cost

Beyond accuracy, we analyze token usage and estimated cost, since practical deployment of large-model pipelines depends on throughput. Table 7 summarizes stage-wise token usage and cost estimates from our internal runs. The key observation is that targeted repair is cheap relative to extraction and thus a favorable place to enforce schema compliance.

Component	P	R	F_1
Ens. union	0.4618	0.7449	0.5702
Adj. consensus	0.7352	0.6582	0.6946
ACE	**0.7812**	**0.8188**	**0.7996**

Table 5: Core component results. "ACE" corresponds to the official submission (ensemble + adjudication + targeted repair). "Ens. union" is the recall base; "Adj. consensus" is the verifier-style filter.

Model setting	P	R	F_1
0-shot	0.7126	0.5112	0.5953
RAG	0.3807	0.4855	0.4267

Table 6: GPT-5-nano RAG effect analysis: retrieval degrades F_1 by 16.9 points due to hallucination/leakage.

9. Error Analysis

We categorize errors into: (i) **FP**: predicted concept - value absent in gold; (ii) **FN**: missed gold item; and (iii) **Value error**: correct concept but mismatched value (often casing, synonyms, list-vs-scalar, or unit normalization).

9.1. Error type distribution

Across the most problematic concepts, we observe a characteristic pattern: high FN counts for implicitly stated activities (e.g., breathing exercises) and high value-error counts for appearance-like multi-select attributes where the ontology enumerations are restrictive (e.g., urine appearance, skin condition). This suggests that improving performance beyond our rank will likely require (i) better discourse-sensitive extraction for implicitly referenced items and (ii) a more principled lexical mapping layer for constrained enumerations.

9.2. Top error types

Table 8 reports the dominant error drivers by type (computed after test labels became available). Respiratory interventions (concept `3`) account for a large share of misses, while urinary symptoms (`177`) are prominent in both hallucinations (FP) and value mismatches, indicating that the ontology constraints are both hard to satisfy and easy to over-predict.

Two mechanisms explain these patterns. First, care actions such as respiratory exercises are often expressed as brief imperatives ("deep breathe", "incentive spirometer") and may be separated from other respiratory descriptors, which increases false negatives under exact-match scoring. Second,

Stage	Model	In tok.	Out tok.	Cost ($)
Extract	Qwen-235B	801,893	33,593	0.09
Extract	GPT-5	503,393	33,593	0.48
Adjud.	Qwen-235B	255,234	33,593	0.04
Repair	Qwen-235B	138,000	3,450	0.02
Total		1,698,520	104,229	0.63

Table 7: Token usage and cost estimation (internal accounting). Stages are extraction, adjudication, and targeted repair; percentages are omitted for compactness.

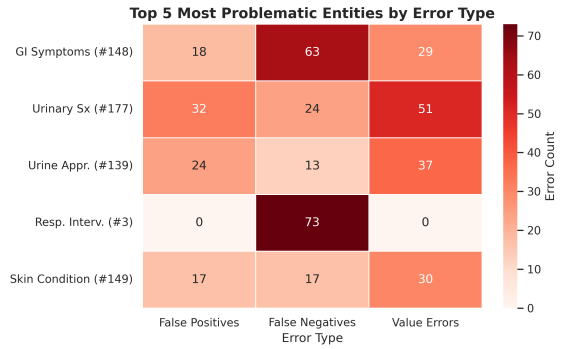

Figure 5: Heatmap of errors by concept. We observe clusters of value normalization errors for multi-select appearance/skin concepts and high FN rates for breathing exercises.

symptom inventories (e.g., urinary symptoms) are high-cardinality multi-select fields; generators tend to over-complete them from clinical priors, and even when the concept is correct, mapping free-form phrasing to the restricted enum set yields value errors.

9.3. Most problematic concepts

From granular error logs, the most frequent *misses and value errors* concentrate in a small subset of concepts (Table 9). Many correspond to implicit or discourse-level information, or to highly variant lexical realizations.

9.4. Hallucination profile and adjudication impact

Adjudication removes 2,219 candidate observations, of which 2,025 (91.3%) are correctly removed false positives. The remaining 194 (8.7%) removed items are true positives, representing a dominant source of recall loss in the ACE pipeline.

Precision - recall impact ratio. This trade-off is strongly favorable for precision: for every true positive removed by the adjudicator, it removes $2025/194 \approx 10.4$ hallucinated observations. We

Type	C_ID	Count	Description
FN	3	73	Respiratory interventions
FN	148	63	Gastrointestinal symptoms
Value	177	51	Urinary symptoms (W/A)
FP	177	32	Urinary symptoms (halluc.)

Table 8: Top error types on the test set (post-competition analysis). W/A means Wrong Answer by value, and halluc. is a hallucinated entity.

ID	Concept name (schema)	#FN
148	Gastrointestinal symptoms	92
177	Urinary symptoms	75
3	Respiratory interventions	73
139	Urine appearance	50
149	Skin condition	47
84	Abdomen assessment	39
165	Breath sounds	34
13	Breathing pattern	32
0	Broset violence checklist - confusion	23
130	Cognitive status	20

Table 9: Top 10 most problematic concepts (false negatives + value errors) from the test-set analysis logs, with official schema names.

interpret this as a clinically meaningful operating point, because the removed false positives are frequently high-risk categories (fabricated interventions and cognitive-state labels), consistent with self-verification approaches for hallucination detection.(Manakul et al., 2023)

Where adjudication fails. The dominant failure mode is *indirect evidence*: dictations may describe an observation using a dispersed set of cues (e.g., symptoms spread across narrative) or with hedges and revisions. In such cases, generators may propose a correct item, but the verifier may not find sufficient explicit support to accept it.

Most hallucinated concepts. The same few concepts dominate false positives, often reflecting clinical priors (e.g., inferring comorbidities) or schema copying (multi-select fields). In our logs, `177` (urinary symptoms) and `31` are the top FP drivers, followed by breathing pattern (`13`) and respiratory exercises (`3`).

9.5. Qualitative examples

Adjudication is particularly effective at rejecting clinically plausible but unsupported interventions and cognitive-state labels; repair fixes minor value canonicalization without re-running extraction.

ID	Concept name (schema)	FP
177	Urinary symptoms	32
31	Diabetes history	31
13	Breathing pattern	29
3	Respiratory interventions	25
139	Urine appearance	24
26	Cognitive disturbances	22
148	Gastrointestinal symptoms	18
130	Cognitive status	18
165	Breath sounds	18
0	Broset violence checklist - confusion	18

Table 10: Top 10 most hallucinated concepts (FP) from the test-set analysis logs, with official schema names.

Quantity	Count	Share
Removed by adjudication	2219	100.0%
Correctly removed (FP)	2025	91.3%
Incorrectly removed (TP)	194	8.7%

Table 11: Adjudication effect from our analysis logs.

Adjudication removal. In a representative case, the union stage proposes multiple unsupported items (e.g., cognitive status labels and interventions) that are clinically plausible but not stated; the adjudicator removes them, improving precision.

Repair canonicalization. In separate cases, repair fixes low-entropy mismatches such as boolean-to-enum mapping (`True`→`Yes`), near-miss enum projection ("borderline inadequate"→`inadequate`), and scalar-to-list normalization for `MULTI_SELECT` fields.

10. Implementation Details

We provide implementation details to support replication. All steps operate on a single transcript and an immutable schema $\mathcal{S}$.

10.1. Prediction representation

We represent predictions as a set of JSON objects with keys: `id` (string concept id) and `value`. Values are typed by the concept's `value_type` in the schema. For `MULTI_SELECT`, we normalize values to a JSON list and score each selected value as a separate $(\texttt{id}, \texttt{value})$ pair.

10.2. Schema validation

Before and after adjudication we validate each candidate against $\mathcal{S}$: (i) concept id must exist; (ii) the predicted value must match the declared value type; (iii) for `SINGLE_SELECT`/`MULTI_SELECT`, each value must appear in `value_enum`; (iv) for `NUMERIC`, we require a parseable numeric value and canonicalize trivial formatting (e.g., stripping units when the schema expects a bare number).

Stage	Model	Temp.	Max tok.	Top-p
Extract.A	Qwen3-235B	0.0	2048	1.0
Extract.B	GPT-5	0.0	2048	1.0
Adjudic.	Qwen3-235B	0.0	2048	1.0
Repair	Qwen3-235B	0.0	512	1.0

Table 12: Inference parameters used in each ACE stage.

10.3. Normalization rules

We apply lightweight normalization to reduce avoidable mismatches: **booleans** (`true/false`, `yes/no`) are canonicalized to the schema enum when applicable; **container** normalization maps scalar→list for `MULTI_SELECT`; **casing/whitespace** are normalized for string-like fields. We avoid aggressive synonym mapping at this stage because it can introduce false positives; instead, enum projection is handled by the targeted repair step with explicit access to allowed values.

10.4. Targeted repair prompts (high level)

For each invalid item (i, v), the repair model is given: (a) transcript x (or a short evidence window), (b) concept name and value type, (c) the allowed enumeration if present, and (d) the invalid value. The model must output a single corrected value or an explicit `null` to drop the item. This design bounds context length and makes repair a low-entropy transformation rather than re-extraction.

10.5. Hyperparameters

All stages use greedy decoding (temperature 0.0) to ensure deterministic outputs; parameters are summarized in Table 12.

11. Discussion

11.1. Why verification helps more than better prompting

In this task, many clinically "reasonable" facts are *not stated* in the transcript. Extractor prompts that encourage completeness tend to over-infer. By reframing the second stage as an entailment-style verifier, we obtain a controllable knob for precision without forcing generators to become conservative.

Risk	Error mode	Examples
High	Fabricated interventions / status	Oxygen delivery device; suctioning; "disoriented" cognitive labels
Medium	Omitted symptoms / care actions	Missed GI symptoms; missed respiratory exercises
Low	Normalization / schema mismatch	list-vs-scalar for MULTI_SELECT; enum projection; unit/casing variants

Table 13: Risk-oriented error taxonomy used in our analysis.

11.2. What limits recall after adjudication

The main failure case for a strict verifier is *distributed evidence*: the transcript may mention an observation in a fragmented way (disfluencies, self-corrections, or references like "same as before"), so neither the candidate nor the verifier sees a crisp support span. Future work could use evidence selection (span extraction) as an intermediate representation, allowing the adjudicator to operate on explicit spans rather than raw text.

11.3. Clinical safety lens

We found it useful to interpret errors not only by F_1 but by clinical risk. High-risk FPs include fabricated interventions (oxygen delivery device, suctioning) and misrepresented cognitive status (e.g., disorientation). Medium-risk errors include symptom omissions that may delay escalation. Low-risk errors include unit formatting and mild synonym mismatches. This framing matches why adjudication is valuable: it preferentially removes high-risk hallucinations even when it slightly harms recall.

This taxonomy is supported by our empirical error distributions: high-risk items are over-represented among frequent false positives (Table 10), while medium-risk omissions align with the most frequent false negatives (Table 8 and Table 9). Low-risk issues largely correspond to value and formatting mismatches that are addressed by schema-aware repair.

11.4. Safety implications by model component

We observe a consistent safety pattern across variants: (i) the recall-oriented union stage is the primary source of high-risk hallucinations; (ii) adjudication removes most of these, effectively acting as a safety filter; and (iii) targeted repair mainly reduces low-risk schema mismatches (container and enum canonicalization) without meaningfully affecting the hallucination rate. This supports a design where safety-critical filtering is placed before normalization-heavy post-processing.

11.5. When retrieval augmentation helps (and when it hurts)

We observe that retrieval augmentation is beneficial for large models by improving coverage of long-tail concepts, but can be harmful for smaller models: they exhibit contextual copying behavior, producing surface forms from retrieved schema snippets even when unsupported by the transcript. This motivates keeping RAG in the high-capacity generator and relying on adjudication to suppress spurious candidates.

12. Limitations and Ethics

The dataset is synthetic and derived from ICU notes; dictation style and concept prevalence may differ in other departments or spoken workflows. Verifier-style adjudication can remove true positives when evidence is indirect or distributed across the transcript. Finally, while we emphasize precision to reduce hallucinated observations, our system is not a clinical device and must be validated prospectively before any deployment. Comparisons with additional small and mid-range models (e.g., Qwen3-4B) remain as future work due to competition timeline constraints.

13. Data and Code Availability

Our code is available in an anonymized repository. The repository contains inference scripts and prompt. [1]

14. Bibliographical References

Monica Agrawal, Stefan Hegselmann, Hunter Hunter, and David Sontag. 2022. Large language models are few-shot clinical information extractors. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1997–2022.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical bert embeddings. In *Proceedings of the*

[1] https://anonymous.4open.science/r/MEDIQA-SYNUR-HSE_NLP-8C2B/

2nd Clinical Natural Language Processing Workshop, pages 72–78. Association for Computational Linguistics.

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. 2023. Prompting is programming: A query language for large language models. In *Proceedings of the 44th ACM SIGPLAN Conference on Programming Language Design and Implementation*, pages 1826–1850.

Jean-Philippe Corbeil, Asma Ben Abacha, George Michalopoulos, Phillip Swazinna, Miguel Del-Agua, Jerome Tremblay, Akila Jeeson Daniel, Cari Bader, Kevin Cho, Pooja Krishnan, Nathan Bodenstab, Thomas Lin, Wenxuan Teng, Francois Beaulieu, and Paul Vozila. 2025. Empowering healthcare practitioners with language models: Structuring speech transcripts in two real-world clinical applications. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, Suzhou (China). Association for Computational Linguistics.

Junyou Li, Qi Zhang, and Philip S Yu. 2024. More agents is all you need. *arXiv preprint arXiv:2402.05120*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

George Michalopoulos, Jean-Philippe Corbeil, Cari Bader, Nate Bodenstab, and Asma Ben Abacha. 2026. Overview of the mediqa-synur 2026 shared task on observation extraction from nurse dictations. In *Proceedings of the 8th Clinical Natural Language Processing Workshop, ClinicalNLP@LREC 2026, Palma, Mallorca, Spain, May 16, 2026*. Association for Computational Linguistics.

Shazia Mitha, Jessica Schwartz, Mollie Hobensack, Kenrick Cato, Kyungmi Woo, Arlene Smaldone, and Maxim Topaz. 2023. Natural language processing of nursing notes: an integrative review. *CIN: Computers, Informatics, Nursing*, 41(6):377–384.

AJ Moy, J Adler-Milstein, et al. 2023. Measurement of clinical documentation burden among physicians and nurses using electronic health record audit logs. *Journal of the American Medical Informatics Association*, 30(4):643–652.

OpenAI. 2024. Gpt-4o system card.

S Panchal and P Thakur. 2024. Harnessing the power of natural language processing in nursing services. *International Journal of Advanced Research*, 12(5):154–156.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, Akshay Nathan, Alan Luo, Alec Helyar, Aleksander Madry, Aleksandr Efremov, Aleksandra Spyra, Alex Baker-Whitcomb, Alex Beutel, Alex Karpenko, Alex Makelov, Alex Neitz, Alex Wei, Alexandra Barr, Alexandre Kirchmeyer, Alexey Ivanov, Alexi Christakis, Alistair Gillespie, Allison Tam, Ally Bennett, Alvin Wan, Alyssa Huang, Amy McDonald Sandjideh, Amy Yang, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrei Gheorghe, Andres Garcia Garcia, Andrew Braunstein, Andrew Liu, Andrew Schmidt, Andrey Mereskin, Andrey Mishchenko, Andy Applebaum, Andy Rogerson, Ann Rajan, Annie Wei, Anoop Kotha, Anubha Srivastava, Anushree Agrawal, Arun Vijayvergiya, Ashley Tyra, Ashvin Nair, Avi Nayak, Ben Eggers, Bessie Ji, Beth Hoover, Bill Chen, Blair Chen, Boaz Barak, Borys Minaiev, Botao Hao, Bowen Baker, Brad Lightcap, Brandon McKinzie, Brandon Wang, Brendan Quinn, Brian Fioca, Brian Hsu, Brian Yang, Brian Yu, Brian Zhang, Brittany Brenner, Callie Riggins Zetino, Cameron Raymond, Camillo Lugaresi, Carolina Paz, Cary Hudson, Cedric Whitney, Chak Li, Charles Chen, Charlotte Cole, Chelsea Voss, Chen Ding, Chen Shen, Chengdu Huang, Chris Colby, Chris Hallacy, Chris Koch, Chris Lu, Christina Kaplan, Christina Kim, CJ Minott-Henriques, Cliff Frey, Cody Yu, Coley Czarnecki, Colin Reid, Colin Wei, Cory Decareaux, Cristina Scheau, Cyril Zhang, Cyrus Forbes, Da Tang, Dakota Goldberg, Dan Roberts, Dana Palmie, Daniel Kappler, Daniel Levine, Daniel Wright, Dave Leo, David Lin, David Robinson, Declan Grabb, Derek Chen, Derek Lim, Derek Salama, Dibya Bhattacharjee, Dimitris Tsipras, Dinghua Li, Dingli Yu, DJ Strouse, Drew Williams, Dylan Hunn, Ed Bayes, Edwin Arbus, Ekin Akyurek, Elaine Ya Le, Elana Widmann, Eli Yani, Elizabeth Proehl, Enis Sert, Enoch Cheung, Eri Schwartz, Eric Han, Eric Jiang,

Eric Mitchell, Eric Sigler, Eric Wallace, Erik Ritter, Erin Kavanaugh, Evan Mays, Evgenii Nikishin, Fangyuan Li, Felipe Petroski Such, Filipe de Avila Belbute Peres, Filippo Raso, Florent Bekerman, Foivos Tsimpourlas, Fotis Chantzis, Francis Song, Francis Zhang, Gaby Raila, Garrett McGrath, Gary Briggs, Gary Yang, Giambattista Parascandolo, Gildas Chabot, Grace Kim, Grace Zhao, Gregory Valiant, Guillaume Leclerc, Hadi Salman, Hanson Wang, Hao Sheng, Haoming Jiang, Haoyu Wang, Haozhun Jin, Harshit Sikchi, Heather Schmidt, Henry Aspegren, Honglin Chen, Huida Qiu, Hunter Lightman, Ian Covert, Ian Kivlichan, Ian Silber, Ian Sohl, Ibrahim Hammoud, Ignasi Clavera, Ikai Lan, Ilge Akkaya, Ilya Kostrikov, Irina Kofman, Isak Etinger, Ishaan Singal, Jackie Hehir, Jacob Huh, Jacqueline Pan, Jake Wilczynski, Jakub Pachocki, James Lee, James Quinn, Jamie Kiros, Janvi Kalra, Jasmyn Samaroo, Jason Wang, Jason Wolfe, Jay Chen, Jay Wang, Jean Harb, Jeffrey Han, Jeffrey Wang, Jennifer Zhao, Jeremy Chen, Jerene Yang, Jerry Tworek, Jesse Chand, Jessica Landon, Jessica Liang, Ji Lin, Jiancheng Liu, Jianfeng Wang, Jie Tang, Jihan Yin, Joanne Jang, Joel Morris, Joey Flynn, Johannes Ferstad, Johannes Heidecke, John Fishbein, John Hallman, Jonah Grant, Jonathan Chien, Jonathan Gordon, Jongsoo Park, Jordan Liss, Jos Kraaijeveld, Joseph Guay, Joseph Mo, Josh Lawson, Josh McGrath, Joshua Vendrow, Joy Jiao, Julian Lee, Julie Steele, Julie Wang, Junhua Mao, Kai Chen, Kai Hayashi, Kai Xiao, Kamyar Salahi, Kan Wu, Karan Sekhri, Karan Sharma, Karan Singhal, Karen Li, Kenny Nguyen, Keren Gu-Lemberg, Kevin King, Kevin Liu, Kevin Stone, Kevin Yu, Kristen Ying, Kristian Georgiev, Kristie Lim, Kushal Tirumala, Kyle Miller, Lama Ahmad, Larry Lv, Laura Clare, Laurance Fauconnet, Lauren Itow, Lauren Yang, Laurentia Romaniuk, Leah Anise, Lee Byron, Leher Pathak, Leon Maksin, Leyan Lo, Leyton Ho, Li Jing, Liang Wu, Liang Xiong, Lien Mamitsuka, Lin Yang, Lindsay McCallum, Lindsey Held, Liz Bourgeois, Logan Engstrom, Lorenz Kuhn, Louis Feuvrier, Lu Zhang, Lucas Switzer, Lukas Kondraciuk, Lukasz Kaiser, Manas Joglekar, Mandeep Singh, Mandip Shah, Manuka Stratta, Marcus Williams, Mark Chen, Mark Sun, Marselus Cayton, Martin Li, Marvin Zhang, Marwan Aljubeh, Matt Nichols, Matthew Haines, Max Schwarzer, Mayank Gupta, Meghan Shah, Melody Huang, Meng Dong, Mengqing Wang, Mia Glaese, Micah Carroll, Michael Lampe, Michael Malek, Michael Sharman, Michael Zhang, Michele Wang, Michelle Pokrass, Mihai Florian, Mikhail Pavlov, Miles Wang, Ming Chen, Mingxuan Wang, Minnia Feng, Mo Bavarian, Molly Lin, Moose Abdool, Mostafa Rohaninejad, Nacho Soto, Natalie Staudacher, Natan LaFontaine, Nathan Marwell, Nelson Liu, Nick Preston, Nick Turley, Nicklas Ansman, Nicole Blades, Nikil Pancha, Nikita Mikhaylin, Niko Felix, Nikunj Handa, Nishant Rai, Nitish Keskar, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Oona Gleeson, Pamela Mishkin, Patryk Lesiewicz, Paul Baltescu, Pavel Belov, Peter Zhokhov, Philip Pronin, Phillip Guo, Phoebe Thacker, Qi Liu, Qiming Yuan, Qinghua Liu, Rachel Dias, Rachel Puckett, Rahul Arora, Ravi Teja Mullapudi, Raz Gaon, Reah Miyara, Rennie Song, Rishabh Aggarwal, RJ Marsan, Robel Yemiru, Robert Xiong, Rohan Kshirsagar, Rohan Nuttall, Roman Tsiupa, Ronen Eldan, Rose Wang, Roshan James, Roy Ziv, Rui Shu, Ruslan Nigmatullin, Saachi Jain, Saam Talaie, Sam Altman, Sam Arnesen, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Sarah Yoo, Savannah Heon, Scott Ethersmith, Sean Grove, Sean Taylor, Sebastien Bubeck, Sever Banesiu, Shaokyi Amdo, Shengjia Zhao, Sherwin Wu, Shibani Santurkar, Shiyu Zhao, Shraman Ray Chaudhuri, Shreyas Krishnaswamy, Shuaiqi, Xia, Shuyang Cheng, Shyamal Anadkat, Simón Posada Fishman, Simon Tobin, Siyuan Fu, Somay Jain, Song Mei, Sonya Egoian, Spencer Kim, Spug Golden, SQ Mah, Steph Lin, Stephen Imm, Steve Sharpe, Steve Yadlowsky, Sulman Choudhry, Sungwon Eum, Suvansh Sanjeev, Tabarak Khan, Tal Stramer, Tao Wang, Tao Xin, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Degry, Thomas Shadwell, Tianfu Fu, Tianshi Gao, Timur Garipov, Tina Sriskandarajah, Toki Sherbakov, Tomer Kaftan, Tomo Hiratsuka, Tongzhou Wang, Tony Song, Tony Zhao, Troy Peterson, Val Kharitonov, Victoria Chernova, Vineet Kosaraju, Vishal Kuo, Vitchyr Pong, Vivek Verma, Vlad Petrov, Wanning Jiang, Weixing Zhang, Wenda Zhou, Wenlei Xie, Wenting Zhan, Wes McCabe, Will DePue, Will Ellsworth, Wulfie Bain, Wyatt Thompson, Xiangning Chen, Xiangyu Qi, Xin Xiang, Xinwei Shi, Yann Dubois, Yaodong Yu, Yara Khakbaz, Yifan Wu, Yilei Qian, Yin Tat Lee, Yinbo Chen, Yizhen Zhang, Yizhong Xiong, Yonglong Tian, Young Cha, Yu Bai, Yu Yang, Yuan Yuan, Yuanzhi Li, Yufeng Zhang, Yuguang Yang, Yujia Jin, Yun Jiang, Yunyun Wang, Yushi Wang, Yutian Liu, Zach Stubenvoll, Zehao Dou, Zheng Wu, and Zhigang Wang. 2025. Openai gpt-5 system card.

Qwen Team. 2025. Qwen3 technical report.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency

improves chain of thought reasoning in language models. In *ICLR 2023*.

Brandon T Willard and Rémi Louf. 2023. Efficient guided generation for large language models. *arXiv preprint arXiv:2307.09702*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36.

A. Prompts

To ensure reproducibility, we summarize the prompt templates used in each ACE stage. Prompts follow a strict role–task–constraint–output structure.

A.1. Extraction prompt (core)

The system prompt for both RAG and zero-shot extractors enforces strict JSON schema adherence.

> **ROLE**: You are a clinical documentation expert extracting structured observations from nurse dictations.
>
> **TASK**: Extract all clinical observations mentioned in the transcript and map them to the given ontology.
>
> **OUTPUT FORMAT**: Return a valid JSON object with an `observations` array.
>
> **RULES**:
>
> 1. Extract only observations explicitly mentioned in the transcript.
> 2. Match observation names and IDs to the ontology exactly.
> 3. For `SINGLE_SELECT`, the value must be an exact member of `value_enum`.
> 4. Do not hallucinate observations or values not supported by the transcript.
>
> **ONTOLOGY**: `{schema_str}`

A.2. Adjudication prompt (consensus)

The adjudicator acts as a verifier that resolves disagreements between two candidate drafts.

> **TRANSCRIPT**: `{transcript}`
>
> Below are two candidate extractions from high-performing models.
>
> **DRAFT A** (Model: `{expert_a_name}`, Est. F1: `{expert_a_f1}`): `{draft_a_json}`
>
> **DRAFT B** (Model: `{expert_b_name}`, Est. F1: `{expert_b_f1}`): `{draft_b_json}`
>
> **YOUR TASK**:
>
> 1. Synthesize a single high-fidelity extraction.
> 2. If drafts agree on ID but differ in value, use the transcript to resolve.
> 3. If an observation appears in only one draft, keep it only when explicitly supported by the transcript.
> 4. Ensure all IDs map to valid schema IDs.
>
> Output only the final JSON object.

A.3. Targeted repair prompt

Instead of re-sending the full ontology, the repair prompt provides localized schema-validation feedback to minimize token usage.

> **TRANSCRIPT**: `{transcript}`
>
> **PREVIOUS (INVALID) OBSERVATIONS**: `{invalid_subset_json}`
>
> **ERRORS FOUND**: Observation ID `{id1}`: Value `{val1}` does not match type `SINGLE_SELECT`.
> Observation ID `{id2}`: ID not found in ontology.
>
> Please correct these observations based on transcript evidence and the ontology, and return the full list of valid observations.

B. Additional qualitative cases

We include short qualitative cases to illustrate (i) hallucination removal by adjudication and (ii) value canonicalization by targeted repair. Full transcripts are omitted for space and anonymization.

B.1. Adjudication removes clinically plausible hallucinations

Case A.1 (document ID 0). The dictation describes seizure history and fall-risk awareness, but it does not state several specific structured items. The union stage proposes multiple clinically plausible observations that are unsupported by the transcript; the adjudicator correctly removes them:

- **History of falls** (concept `78`): `Yes` (unsupported),
- **Ambulatory aid** (concept `21`): `walker` (unsupported),

- **Mobility** (concept 89): `Mildly impaired` (unsupported),
- **Cognitive status** (concept 130): `Disoriented to time` (unsupported),
- **Fall risk identification** (concept 107): `High` (unsupported).

This example illustrates why a verifier-style stage is valuable: these items are plausible extrapolations (fall history, mobility limitations, disorientation) but would be unsafe to record as structured observations without explicit evidence in the dictation.

B.2. Targeted repair fixes low-entropy schema/value mismatches

Case A.2 (document ID 5; Use of accessory muscles, concept 38). Before repair, the system outputs `True` for a `SINGLE_SELECT` concept whose allowed values are `Yes/No`. Targeted repair canonicalizes `True`→`Yes`, matching gold.

Case A.3 (document ID 14; Nutrition status, concept 25). Repair projects a near-miss enum phrase "borderline inadequate"→`inadequate`, matching gold.

Case A.4 (document ID 20; Breathing pattern, concept 13). Repair normalizes container type for a `MULTI_SELECT` field: `labored`→`['labored']`, matching gold.

These cases demonstrate that repair is most effective for constrained transformations (boolean/enum canonicalization and list-vs-scalar normalization) rather than re-extraction.