Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Faithful Structured Representation from Clinical Notes via Iterative Feedback–Refinement

Eslam Ahmed Mohamed

i.mohamed@deakin.edu.au

Deakin University

Irini Logothetis

Deakin University

Adrian Bingham

Deakin University

Kon Mouzakis

Deakin University

Method Article

Keywords: Large Language Models, Clinical Natural Language Processing, Knowledge Representation, Summarization, Schema Induction

Posted Date: July 30th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10249995/v2

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Faithful Structured Representation from Clinical Notes via Iterative Feedback–Refinement

Eslam Ahmed Mohamed[1*], Irini Logothetis[1], Adrian Bingham[1], Kon Mouzakis[1]

[1*]Deakin Applied Artificial Intelligence Initiative, Deakin University, Burwood, Victoria, Australia.

*Corresponding author(s). E-mail(s): i.mohamed@deakin.edu.au;
Contributing authors: rena.logothetis@deakin.edu.au;
adrian.bingham@deakin.edu.au; kon.mouzakis@deakin.edu.au;

Abstract

Faithfully converting unstructured text into structured representations remains a fundamental challenge. Existing approaches either generate free-text summaries, which are prone to hallucination and omission, or populate fixed, manually engineered schemas that generalize poorly across heterogeneous documentation styles. In both cases, the target representation is fixed in advance, so deficiencies in the representation propagate to every extraction. In this work, we present an iterative feedback–refinement framework for adaptive schema induction using large language models (LLMs), which makes the schema itself the object of optimization. A dual-LLM loop iteratively evaluates the schema against source documents and incorporates aggregated feedback into successive schema revisions, progressively improving a shared schema without labeled data or task-specific fine-tuning. To assess reliability, we adopt a faithfulness measure that decomposes quality into precision (absence of hallucination) and recall (absence of omission). Experiments on the public MTSamples clinical corpus with 8 open-weight LLMs compare the framework against extractive, abstractive, and in-context-learning baselines. Iterative schema refinement achieves the highest mean faithfulness across all three LLM-judge metrics while reducing document length to 34.5% of the source. The improvement is driven primarily by higher recall while maintaining comparable or higher precision, and mid-sized models match or outperform larger ones. These findings indicate that optimizing the schema, rather than individual outputs, provides an effective, model-agnostic approach for faithful structured representation.

Keywords: Large Language Models, Clinical Natural Language Processing, Knowledge Representation, Summarization, Schema Induction

1 Introduction

Transforming unstructured text into faithful structured representations is a foundational problem in knowledge and data engineering, underpinning information extraction, knowledge graph construction, interoperability, and decision support [1]. The central challenge is heterogeneity: source documents vary widely in layout, narrative style, and section coverage across domains and institutions, limiting the ability of representations designed for one document type to generalize to others [2, 3]. Recent advances in large language models (LLMs) have substantially improved the transformation of unstructured text into structured representations. In practice, LLM-based systems typically follow one of two paradigms: generating condensed free-text summaries or mapping text onto a predefined, manually engineered schema. Both assume a fixed target representation and optimize only the generated instances that conform to it. Consequently, deficiencies in the representation propagate to every extraction, while the representation itself remains unchanged. We instead argue that the representation, an explicitly typed schema, should itself be induced and iteratively refined from the data.

We study this problem in the domain of clinical documentation, an especially demanding instance of unstructured-text structuring. Clinical notes are the primary repository of patient-specific knowledge in electronic health records (EHRs), encoding symptoms, diagnoses, treatment decisions, and longitudinal communication between providers [1]. They combine high structural heterogeneity across specialties and institutions [2] with low tolerance for error: even human-authored notes contain, on average, at least one error and four omissions [4]. These failure modes are severe for automated systems as well; studies of GPT-4 and GPT-3.5-turbo on emergency-department notes report hallucinations in 42% of summaries and clinically relevant omissions in 47% of cases [5]. Automated methods must therefore be evaluated not only on how compactly they represent a document but on *faithfulness*: the extent to which the representation is supported by, and preserves the content of, its source [6, 7]. Under this definition, hallucinations (asserting content absent from or inconsistent with the source) and omissions (dropping information present in the source) are complementary failure modes [8]. This combination of heterogeneity, scale, and stringent faithfulness requirements makes clinical notes a rigorous testbed for adaptive schema induction while preserving the domain generality of the proposed method.

Existing approaches reveal two main limitations. (1) Summarization- and schema-population methods optimize outputs against a target representation that is fixed in advance, so any deficiency in that representation persists across all generated instances; iterative-refinement methods mitigate this only at the level of individual outputs, refining the generated text rather than the underlying representation (§2). (2) The few

methods that treat the schema as mutable either rely on expert-guided, human-in-the-loop refinement outside the clinical domain [9], or optimize the schema for extraction accuracy rather than faithfulness [10], while clinical systems continue to populate fixed, manually engineered schemas [11, 12]. Because these schemas are constructed once and subsequently held fixed, deficiencies introduced during their design propagate to every downstream extraction, whereas output-level refinement corrects only individual instances. Automated, faithfulness-driven refinement of the schema itself therefore remains unexplored. We address this gap by making the schema the object of optimization, so that each refinement propagates to all subsequent extractions, and driving that refinement autonomously from per-document evidence of hallucination and omission against the source, without human intervention, labeled data, or task-specific fine-tuning.

We reframe structured clinical documentation as a problem of adaptive schema induction rather than instance generation from a predefined template. Instead of optimizing generated outputs, we optimize the schema to improve faithfulness by reducing hallucinations (raising precision) and omissions (raising recall). Clinical notes are first embedded and clustered so that refinement is exposed to diverse structural patterns rather than a random sample. A dual-LLM loop then alternates between an evaluator, which assesses the current schema against the source notes and produces categorical feedback on missing fields, value-representation errors, and structural deficiencies, and a refiner, which incorporates aggregated feedback into a revised schema. Optimizing the schema rather than individual outputs improves faithfulness at the representation level by reducing hallucinations and omissions. Precision improves as unreliable fields that cannot be consistently grounded in the source are removed, while recall improves as fields are introduced for information that is consistently present but previously unrepresented.

Our main contributions are summarized as follows:

- We propose an iterative feedback–refinement framework that reframes structured representation as adaptive schema induction, making a single shared schema, rather than the generated instance, the object of optimization.
- We design a decoupled dual-LLM evaluator–refiner loop that induces a faithful, explicitly typed schema from unlabeled notes, grounding every schema edit in per-document feedback on missing fields, value-representation errors, and structural deficiencies, and requiring no task-specific fine-tuning.
- We develop a faithfulness-driven evaluation protocol that decomposes quality into hallucination (precision) and omission (recall) and instantiates it with four mechanistically distinct judges to reduce single-metric bias.
- We evaluate the framework on the public MTSamples corpus across 8 open-weight LLMs (1–109B parameters), demonstrating higher faithfulness at greater compression than extractive, abstractive, and in-context-learning (ICL) baselines, while showing that mid-sized models can match or exceed substantially larger models.

Table 1 Positioning against representative approaches to structured representation. Adaptive: the target schema is induced or refined rather than fixed; Autonomous: refinement requires no human in the loop; Faithfulness-driven: schema edits are guided by explicit hallucination and omission evidence: a single schema is optimized and reused across all documents.

Approach	Adaptive	Autonomous	Faithful.-driven
Fixed-schema population [11, 12]	No	Yes	No
Free-text summarization [15]	No	Yes	No
Output-level refinement [16, 17]	No	Yes	Partial
Human-in-the-loop schema discovery [9]	Yes	No	No
Schema optimization for extraction [10]	Yes	Yes	No
Adaptive KG construction [13, 14]	Yes	Yes	No
Our work	Yes	Yes	Yes

2 Related Work

2.1 Schema Induction and Adaptive Representation

Converting unstructured text into structured, explicitly typed representations is a long-standing goal in knowledge and data engineering, and a recurring question is whether the target representation is fixed in advance or induced from the data. Most deployed systems assume a fixed schema and optimize only the mapping of text onto it, which limits generalization when documents vary in structure. A growing line of work instead treats the schema itself as adaptable. AutoSchemaKG [13] performs autonomous knowledge-graph construction by inducing entity and event schemas directly from web-scale corpora, and Extract-Define-Canonicalize (EDC) [14] constructs a self-generated schema when no predefined one is available before canonicalizing extracted triples. Most closely related to our setting, LLMs4SchemaDiscovery [9] generates and refines scientific schemas through an iterative workflow, and PARSE [10] treats JSON schemas as optimizable interfaces and autonomously rewrites them to improve entity-extraction reliability. These methods share our premise that the schema should be induced rather than fixed, but differ in objective and setting. AutoSchemaKG and EDC target entity, event, and relation types for knowledge-graph triples rather than a shared representation for faithful text-to-JSON conversion; LLMs4SchemaDiscovery depends on expert, human-in-the-loop refinement; and PARSE optimizes schemas for extraction accuracy on web and dialogue data rather than for faithfulness on clinical text. In contrast, our framework refines a single shared schema autonomously, using an LLM evaluator rather than a human, and drives each edit from explicit per-document evidence of hallucination and omission, so that improvements accumulate at the representation level and propagate to every subsequent extraction. Table 1 summarizes this comparison.

2.2 LLM-Based Structured Extraction and Refinement

A complementary body of work uses LLMs to populate predefined structured representations, spanning closed, schema-bound extraction and open, schema-agnostic

extraction [18]. Clinical instances include structured note templates [19–22], JSON-based formats [11], FHIR-compliant outputs [23], retrieval-augmented pipelines [24], generation conditioned on retrieved case reports and biomedical knowledge graphs [25], and fine-tuned models for structured extraction [12, 26, 27]. These approaches assume a fixed, manually engineered schema and optimize only the instances conforming to it, so limitations in the schema persist across every extraction.

A separate line of work shows that iterative refinement outperforms single-pass generation. Chain-of-thought (CoT) prompting decomposes a task into intermediate reasoning stages [28] but does not revise its own outputs. SELF-REFINE [16] improves outputs through repeated generation, self-critique, and revision, and the Detect-Critique-Refine framework of Wadhwa et al. [17] shows that decomposing correction into error detection, fine-grained critique, and targeted revision yields greater factual consistency than end-to-end correction. Self-reflection loops similarly reduce hallucinations in medical question answering by repeatedly evaluating both background knowledge and generated answers [29]. However, intrinsic self-correction is unreliable without external grounding: LLMs often fail to correct their own reasoning, and performance can degrade after self-revision [30], and refinement benefits from external, grounded feedback rather than intrinsic self-critique [31, 32], motivating role-separated designs such as Reflexion [33]. Crucially, all of this work refines the generated output. Our framework instead applies a decoupled evaluator–refiner loop to the shared schema, grounded in external faithfulness evidence, so that improvements accumulate at the representation level rather than being recomputed for each instance.

2.3 Faithfulness Evaluation

In the natural language processing literature, faithfulness is the extent to which a generated output is supported by its source document [6, 7], with hallucination and omission as complementary failure modes [8]. Such failures persist in clinical generation even after prompt refinement, with a substantial fraction of hallucinations classified as major clinical errors [34], and they are not reliably captured by reference-overlap metrics such as ROUGE [35], which measure lexical overlap rather than factual alignment. Existing mitigation strategies largely operate after generation, including faithfulness-aware decoding [36], factuality metrics better correlated with human judgment [37], and post-hoc correction frameworks such as MEDAL [38], and therefore do not resolve deficiencies originating in the underlying representation. For evaluation, LLM-as-judge methods such as G-Eval [39] and reference-free consistency checks such as SelfCheck-GPT [40] and MQAG [41] have become standard, but judges may exhibit systematic preference for their own or structured generations [42]. We address this concern by combining four mechanistically distinct judges (§5.2) rather than relying on any single scoring mechanism.

2.4 Structured Representation in the Clinical Domain

Clinical text summarization has been the dominant route to improving information accessibility in EHRs, evolving from early rule-based systems built on textual patterns and medical ontologies [43], through transformer models such as BERT and BART [44],

to modern LLMs [45] that perform strongly across radiology reports, progress notes, and doctor–patient dialogue [15]. Summarization nonetheless requires oversight, as fluent outputs can deviate from source documentation or omit critical details [46], and free text lacks the structural constraints needed for reuse, interoperability, and faithful representation [47]. This motivates structured representations, although existing clinical structuring systems continue to rely on fixed schemas. Our framework extends this line of work by introducing autonomous, faithfulness-driven schema induction to this setting, using clinical notes as a high-stakes, high-heterogeneity testbed for a method that is otherwise domain-general.

3 Dataset and Preprocessing

3.1 MTSamples Clinical Notes Corpus

This study utilizes the MTSamples dataset[1], a publicly available corpus of 5,000 transcribed clinical notes spanning 40 medical categories including 'Surgery', 'Cardiovascular/Pulmonary', 'Gastroenterology', and 'Consult - History and Physicals'. The number of reports per category ranges from 6 to 1,105, yielding an imbalanced but clinically diverse corpus. Clinical Notes are semi-structured free text, typically organized under section headings such as Chief Complaint, History of Present Illness, Medications, Physical Examination, and Assessment and Plan; the presence, ordering, and content of these sections vary substantially across specialties and documentation styles, reflecting real-world documentation heterogeneity.

We focus on the 'Consult - History and Physicals' category, comprising 515 reports, because its notes contain comprehensive patient information and consistently include the sections relevant to clinical decision-making, providing a suitable benchmark for schema induction while remaining representative of real-world documentation. Figure 1 shows an example report, illustrating the semi-structured, section-based format of the corpus.

3.2 Preprocessing Pipeline

The preprocessing pipeline consists of four sequential stages: data cleaning, length-based outlier removal, embedding-based thematic clustering, and stratified data partitioning. Figure 2 summarizes the pipeline and the data-flow statistics at each stage.

3.2.1 Data Cleaning and Outlier Removal

We first standardized the raw notes to remove boilerplate headers, transcription artefacts, and formatting inconsistencies including erroneous whitespace, inconsistent line breaks, and delimiter variability. This cleaning preserves clinical content while standardizing document structure. After standardizing the notes, we prepared the corpus for efficient processing by accounting for the memory requirements of long-context inference. Because the memory required by the key–value (KV) cache grows linearly

[1] https://www.mtsamples.com/

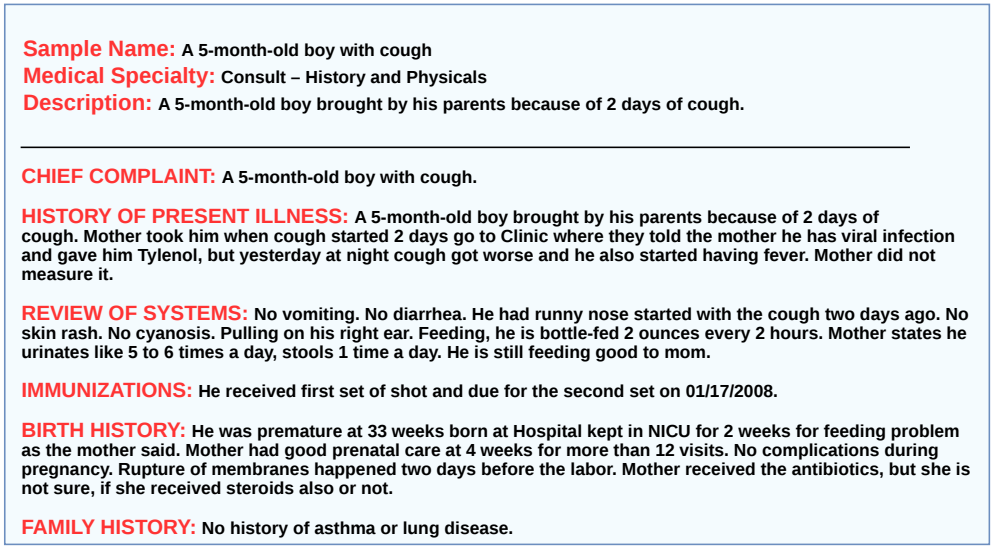


Fig. 1 A representation of the medical consult note.

with the number of input tokens, processing the full corpus at once exceeds available Graphics Processing Unit (GPU) memory. For example, the Llama-4-Scout model used in our experiments requires approximately 109 GB of memory for its weights, while the key–value (KV) cache requires an additional 0.19 MB per token at BF16 precision[2]. Processing the full corpus (520,678 tokens) therefore requires roughly 95 GB of KV-cache memory in addition to the weights, about 204 GB in total, which exceeds the 141 GB of a single NVIDIA H200 GPU and motivates memory-aware filtering.

We tokenized all 515 documents and retained those satisfying

$$\left|\frac{x-\mu}{\sigma}\right| < \tau \tag{1}$$

where x is a document's token count, μ and σ are the mean and standard deviation of token counts across the corpus, and τ is a z-score threshold. We evaluated several values of τ (Table 2) and selected $\tau = 0.7$, the largest threshold whose combined weight and KV-cache memory requirements fit within the available GPU memory of two NVIDIA H200 GPUs (141 GB of memory each; 282 GB total). This retained 337 of 515 documents and reduced the total token volume by 39.5% (520,678 to 314,792 tokens), removing long notes that exceed the KV-cache budget and very short notes that carry insufficient information for reliable schema evaluation, without the large sample-size loss incurred at smaller thresholds (e.g., $\tau = 0.5$). Figure 3 shows the token distributions before and after filtering.

[2] $2 \times L \times H_{kv} \times d_{\text{head}} \times b = 2 \times 48 \times 8 \times 128 \times 2$ bytes per token, where L is the number of layers, H_{kv} is the number of key–value heads, d_{head} is the attention head dimension, and $b = 2$ bytes for BF16 precision.

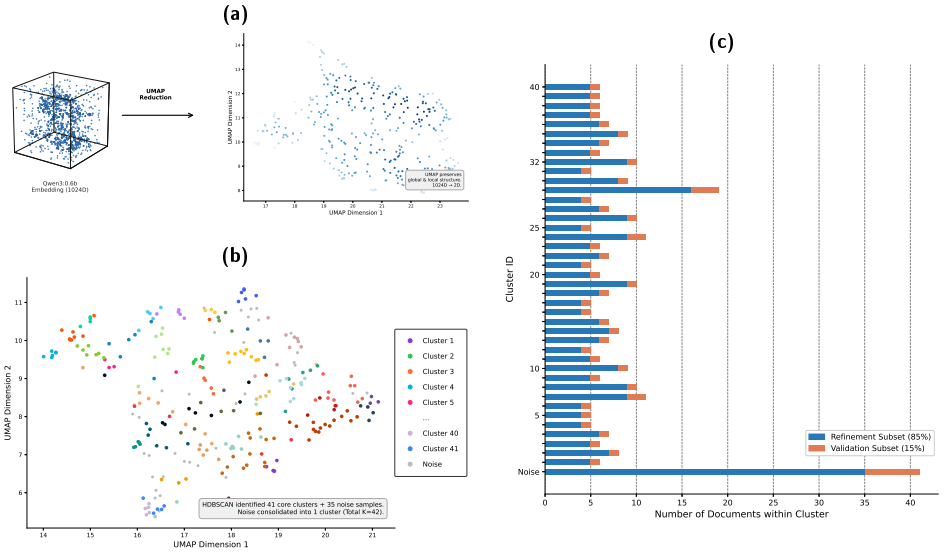


Fig. 2 Preprocessing pipeline overview. **(a)** 1024-dimensional document embeddings generated by `Qwen3:0.6b` are projected to 2D via UMAP, revealing latent thematic structure in the corpus. **(b)** HDBSCAN clustering identifies 41 thematic clusters plus 41 noise samples (consolidated into one), yielding a total of 42 clusters ($K = 42$). **(c)** Cluster-stratified 85%/15% partitioning allocates documents into refinement ($N_{\text{refine}} = 286$) and validation ($N_{\text{val}} = 51$) subsets, preserving proportional cluster representation across both splits.

Table 2 Effect of the z-score threshold τ on retained documents and total token volume. Reduction is relative to the full corpus (520,678 tokens). We select $\tau = 0.7$.

τ	Documents	Total tokens	Reduction
3.0	507	489,877	5.9%
2.0	493	457,654	12.1%
1.0	424	384,803	26.1%
0.9	398	361,925	30.5%
0.7	**337**	**314,792**	**39.5%**
0.5	246	233,703	55.1%

3.2.2 Embedding-Based Thematic Clustering

To expose the iterative feedback–refinement process to diverse note structures while maintaining computational tractability, we partitioned the filtered corpus into thematic clusters using an unsupervised embedding-based pipeline. This cluster-organized approach ensures that schema refinement encounters structurally distinct note types in a controlled sequence rather than through random sampling, which could result in redundant evaluations or premature convergence.

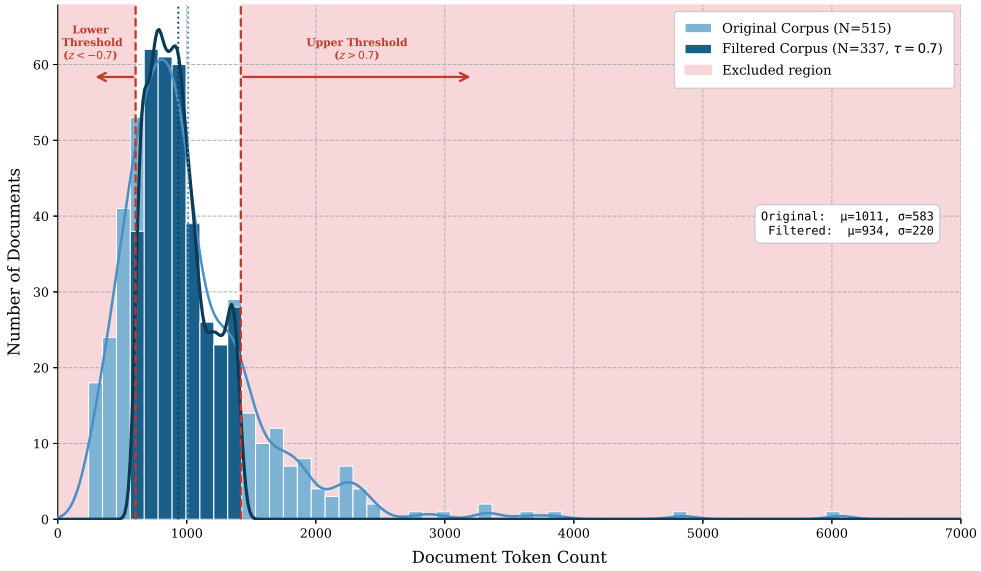


Fig. 3 Document length distribution before and after length-based filtering. The original corpus ($N = 515$, light blue) exhibits high variance with prominent long tails. Applying a z-score threshold of $\tau = 0.7$ (dashed red lines) trims extreme outliers, resulting in a more uniform filtered corpus ($N = 337$, dark blue).

Embedding Generation.

We employed `Qwen3:0.6b` as the embedding model because it provides competitive performance on English biomedical retrieval tasks in the MTEB (Massive Text Embedding Benchmark) benchmark [48] while remaining computationally efficient for embedding the complete 337-document corpus. Each note was encoded into a 1024-dimensional dense vector space $\mathbf{e}_i \in \mathbb{R}^{1024}$, capturing both semantic content (diagnoses, symptoms, medications) along with structural patterns (section organization, information density).

Dimensionality Reduction.

We applied UMAP (Uniform Manifold Approximation and Projection) [49] to project dimensionality from 1024 to 2 dimensions (Figure 2A) prior to clustering. UMAP was selected over alternatives (PCA, t-SNE) because it preserves both local neighborhood relationships and global topology while effectively modelling the nonlinear structure of high-dimensional text embeddings [50, 51]. These properties are particularly important for clinical documentation, where notes may exhibit similarity at multiple levels of granularity, such as shared presenting complaints despite differing diagnoses, management plans, or document organization [52]. The resulting low-dimensional representation facilitates clustering while preserving clinically meaningful relationships among documents.

Density-Based Clustering.

For clustering, we employed HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) [53], which detects clusters of varying density, determines the cluster count automatically, and leaves structurally unique notes unassigned rather than forcing them into ill-fitting clusters. HDBSCAN identified 41 thematic clusters and 41 noise samples (notes not belonging to any dense region); to avoid singleton batches during refinement, we merged all noise samples into one additional cluster, giving $K = 42$ (Figure 2B). The 41 core clusters averaged 7.2 notes each (approximately 6,672 tokens), and manual inspection confirmed that assignments corresponded to clinically meaningful groupings by chief complaint (e.g., cardiovascular, neurological, or musculoskeletal), note complexity, and documentation style.

3.2.3 Data Partitioning for Schema Refinement and Validation

To enable systematic evaluation, we constructed a cluster-stratified partition of the 337 filtered documents into refinement and validation subsets. This design preserves proportional representation of all 42 clusters across both subsets (Figure 2C), ensuring exposure to the full diversity of clinical note types.

We applied an 85%/15% split:

- **Refinement subset** (85%, $N_{\text{refine}} = 286$): used for in-context learning pattern analysis, schema generation, and iterative feedback-based refinement across cluster batches.
- **Validation subset** (15%, $N_{\text{val}} = 51$): reserved for schema filling using the finalized schema, and for baseline summarization comparisons.

Cluster assignments were directly used as the batching mechanism for iterative schema refinement. This ensures each iteration processes structurally coherent notes while progressively exposing the schema to the full diversity of clinical documentation patterns.

4 Methodology

We study the transformation of unstructured clinical notes into structured and condensed representations through a comparative framework encompassing five approaches. Our framework is organized into two paradigms: **(1) Summarization** (extractive and abstractive), which produces condensed text representations, and **(2) Schema-Based Structuring** (ICL and iterative feedback–refinement), which produces machine-readable JSON documents conforming to formal schemas. The summarization approaches and ICL serve as baselines; our primary contribution is the iterative feedback–refinement. Figure 4 provides a high-level overview of our proposed framework.

4.1 Problem Formulation

Let $\mathcal{N} = \{n_1, \ldots, n_N\}$ be a corpus of N free-text clinical notes, where n_i denotes an individual note. For the summarization task, the goal is to produce a condensed

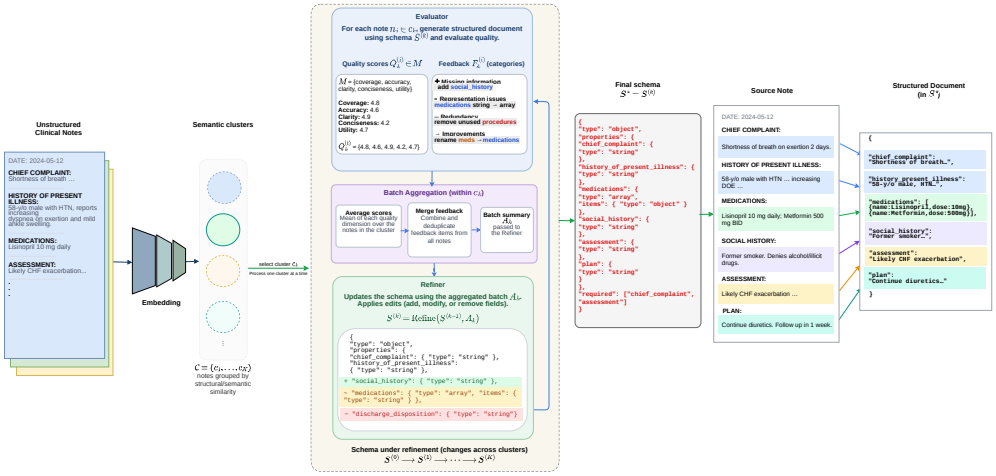


Fig. 4 Overview of the proposed framework. Unstructured clinical notes are embedded and grouped into semantic clusters. Each cluster is processed in turn by a dual-LLM loop in which an evaluator scores the current schema against the cluster's notes and generates structured feedback, and a refiner edits the schema accordingly (adding, modifying, or removing fields). The refined schema carries forward across clusters to yield a final schema $\mathcal{S}^*$, which maps source-note spans into a condensed, reorganized structured document.

text representation $\hat{s}_i$ of each note n_i that preserves clinically salient information. For schema-based structuring, we seek a schema $\mathcal{S}^*$ that, through a fixed mapping, converts each note n_i into a structured document $\hat{d}_i$ conforming to $\mathcal{S}^*$ while minimizing omissions and hallucinations. The schema is a hierarchical key-value structure in which every field is explicitly typed using JSON Schema type definitions, and it remains fixed once finalized, ensuring structural consistency across all structured documents.

The refinement subset is partitioned into K disjoint clusters $\{c_1, \ldots, c_K\}$ derived from the UMAP–HDBSCAN pipeline (§3.2.2), where each cluster c_k groups notes sharing structural and linguistic patterns. This partition is used exclusively for the iterative schema refinement approach.

4.2 Baseline Approaches

4.2.1 Baseline 1: Extractive Summarization (BERT-Centroid)

Extractive summarization selects salient sentences from a note verbatim rather than generating new text, which preserves the original clinical phrasing and avoids hallucination by construction [54]. The goal is identifying which sentences are most representative of the note as a whole. We adopt a centroid-based approach in which the note is reduced to a single vector that captures its overall meaning, and each sentence is ranked by its cosine similarity to that vector, so that sentences reflecting the note's central content rank above peripheral or off-topic ones. The top-ranked sentences are returned in their original document order, forming a coherent excerpt of the source.

Formally, each note n_i contains a set of T sentences, $\{s_1, \ldots, s_T\}$, where s_t is the t^{th} sentence. We obtained sentence embeddings $\mathbf{e}_t \in \mathbb{R}^{768}$ from Bio_ClinicalBERT[3], a BERT model pre-trained on MIMIC-III clinical text [55], for each sentence s_t using mean pooling over non-padding token hidden states. The document-level centroid $\mathbf{c}_i$ is the mean of these embeddings and approximates the note's overall semantic content,

$$\mathbf{c}_i = \frac{1}{T} \sum_{t=1}^{T} \mathbf{e}_t. \tag{2}$$

Each sentence is scored by its cosine similarity to $\mathbf{c}_i$, and the top M are retained, where M is the number of sentences to keep, set adaptively from note length so that short notes are not over-compressed and long notes do not produce unwieldy summaries. Given a target compression ratio $\rho = 0.50$, mean sentence length L in characters, and a note of length $|n_i|$,

$$M = \max\Big(M_{\min},\ \min\Big(\Big\lfloor \tfrac{\rho \cdot |n_i|}{L} \Big\rfloor,\ M_{\max},\ T\Big)\Big) \tag{3}$$

with $M_{\min} = 3$ and $M_{\max} = 15$ as the lower and upper bounds. The selected sentences are returned in their original document order, yielding a summary that is a verbatim excerpt of the source note rather than a paraphrase.

4.2.2 Baseline 2: Standard Abstractive Summarization

The standard abstractive approach employs a single-turn instruction-following paradigm. The LLM receives the full clinical note and is instructed to produce a concise, medically faithful summary. No structural scaffolding or intermediate reasoning steps are imposed, testing the model's intrinsic clinical summarization capacity without prompt-engineering enhancements.

4.2.3 Baseline 3: Chain-of-Thought Abstractive Summarization

The CoT approach decomposes summarization into two sequential LLM calls that share a persistent conversation context, motivated by prior findings that intermediate reasoning steps improve output quality on knowledge-intensive tasks [28]. In the first call, the model performs an open-ended analysis of the note, identifying clinically salient information and events without being given any prescriptive section labels such as "chief complaint" or "assessment", so that it can surface relevant content without structural assumptions that may not generalize across note types. In the second call, conditioned on the analysis produced in the first, the model synthesizes its observations into a concise clinical summary. This decoupling of comprehension from compression is intended to reduce the risk of missing information located in secondary sections, such as buried medication lists or incidental findings, which single-pass abstractive summarization tends to overlook [5, 56, 57].

[3] https://huggingface.co/emilyalsentzer/Bio_ClinicalBERT

4.2.4 Baseline 4: In-Context Learning Schema Extraction

The ICL approach derives a schema from the refinement subset in two LLM passes without any iterative critique, serving as a strong non-iterative baseline for schema generation. In the first pass, the full refinement subset of N_{refine} clinical notes is provided to the LLM in a single prompt, exploiting the extended context window of `Llama-4-Scout`[4]; the model is asked to analyze structural and linguistic patterns across the corpus, including sectioning conventions, content organization, and recurrent medical terminology. Seeing the entire refinement corpus at once, without an explicit critique-and-revise loop, is what distinguishes ICL from the iterative approach. In the second pass, the LLM uses the retained analysis together with the original notes to produce a comprehensive JSON schema along with one example structured document that guides subsequent extraction; the schema output is constrained at decoding time using a JSON Schema specification. The resulting schema is then applied to each note in the validation subset using a dedicated stateless filler client instructed to populate all applicable fields.

4.3 Proposed Approach: Iterative Feedback–Refinement

Our proposed iterative feedback–refinement framework progressively improves a JSON schema through cluster-organized LLM evaluation, structured feedback aggregation, and iterative schema updates. The method consists of a schema-initialization stage followed by three stages, namely evaluation, aggregation, and refinement, executed sequentially across the K cluster batches. In addition, a memory-management mechanism governs how conversational state is shared within and across batches to prevent cross-batch contamination.

4.3.1 Stage 1: Schema Initialization

An initial schema $\mathcal{S}^{(0)}$ is generated by a zero-shot prompt to the evaluator LLM. The prompt asks for a JSON schema covering the section types commonly seen in clinical notes (demographics, chief complaint, history of present illness, examination, assessment, and plan), but treats this list as guidance rather than a fixed skeleton, the model is free to add, omit, or restructure sections. All fields are declared optional and typed using standard JSON Schema scalars and composites. A deterministic post-processor then strips any `required` constraints the model may have introduced and normalizes union types, so that all subsequent refinement operates on a consistent representation. $\mathcal{S}^{(0)}$ is intentionally minimal; substantial structural change is expected during refinement.

4.3.2 Stage 2: Per-Note Evaluation

For each note n_i in cluster c_k, the evaluator performs an extraction step using the current schema $\mathcal{S}^{(k-1)}$, followed by a two-part critique comprising quality scoring and feedback generation. The first step produces a structured document $\hat{d}_i$ from the note, and the second critiques the schema in light of that document. Separating extraction

[4] https://ai.meta.com/blog/llama-4-multimodal-intelligence/

from critique avoids overloading a single prompt and yields more focused feedback, in line with the decoupled-evaluator pattern of Wadhwa et al. [17]. The two critique steps share a batch-level conversation history so that critique remains consistent across notes within a cluster.

(1) Structured Document Generation.

A stateless extraction call populates all fields of $\mathcal{S}^{(k-1)}$ that are supported by the note, producing a structured document $\hat{d}_i$. Fields with no corresponding information in the source are omitted rather than populated with `null`, and no inference, guessing, or out-of-schema fields are permitted. The output is constrained to a JSON object at inference time.

(2) Quality Scoring.

For each note n_i, the evaluator scores the schema across five quality dimensions indexed by $j \in \{1, \ldots, 5\}$, each using a 6-point ordinal scale $q_{i,j} \in \{0, \ldots, 5\}$ (0 = lowest, 5 = highest). The dimensions and their operational definitions are as follows:

- Coverage: How well the schema captures the relevant clinical information contained in the note.
- Accuracy: How faithfully the resulting structured document $\hat{d}_i$ represents the source note.
- Clarity: How logical and easy to interpret the schema structure is, including the appropriateness of hierarchical nesting.
- Conciseness: How free from redundant, overlapping, or low-utility fields the schema is.
- Utility: How clinically practical and useful the schema is for downstream workflows.

(3) Feedback Generation.

The evaluator generates structured feedback following four categories indexed by $r \in \{1, \ldots, 4\}$ for each note n_i. The generated feedback is therefore denoted as $f_{i,r}$. The following are the feedback categories used by the evaluator:

- Missing information: Clinically important content present in the note that has no corresponding schema field.
- Representation issues: Fields that exist in the schema but whose types or structures are mismatched with the documented format (e.g., blood pressure stored as a string rather than a structured object).
- Redundancy: Duplicate, overlapping, or unnecessary fields that add noise without clinical utility.
- Improvements: Concrete, actionable suggestions for the next schema version.

The output of Stage 2 is constrained to a JSON object containing the scores, the feedback, and the structured document to ensure that the evaluation response is always parseable and conforms to the expected structure.

4.3.3 Stage 3: Batch Aggregation

After evaluating all notes in cluster c_k, the per-note results are aggregated to identify systematic schema deficiencies prior to refinement. For each quality dimension j, we compute the arithmetic mean score across the cluster:

$$\bar{q}_k^{(j)} = \frac{1}{|c_k|} \sum_{n_i \in c_k} q_{i,j}, \quad j \in \{1, \ldots, 5\} \tag{4}$$

where $|c_k|$ denotes the number of notes in cluster c_k.

We use the mean rather than the median for two reasons. Batch sizes are small (around 7 notes per cluster), and atypical notes that expose genuine schema gaps should influence the aggregated signal rather than be discarded as outliers. In a feedback-driven refinement setting, sensitivity to a single note that exposes a missing or mis-typed field is a feature rather than a source of noise.

For each feedback category r, we aggregate and deduplicate all items generated within the batch:

$$F_k^{(r)} = \text{deduplicate}\left(\bigcup_{n_i \in c_k} f_{i,r} \right), \quad r \in \{1, \ldots, 4\} \tag{5}$$

where deduplicate$(\cdot)$ removes duplicate strings while preserving first-occurrence order. Near-duplicate items are deliberately retained, since their repetition is itself a useful frequency signal for the refiner. The final batch aggregation is the tuple $\mathcal{A}_k = \left(\{\bar{q}_k^{(j)}\}_{j=1}^{5}, \{F_k^{(r)}\}_{r=1}^{4}, |c_k| \right)$, which serves as the input to schema refinement.

4.3.4 Stage 4: Schema Refinement

Given the current schema $\mathcal{S}^{(k-1)}$ and the batch-level aggregation $\mathcal{A}_k$, a separate stateless refiner LLM produces an updated schema $\mathcal{S}^{(k)}$. The refinement prompt contains: (1) the complete current schema, (2) the five mean quality scores, and (3) the deduplicated feedback items across the four categories. The refiner is also given a structured chain-of-thought scaffold asking it to identify the two or three most important issues raised by the feedback, decide for each whether to add, remove, rename, merge, or restructure fields, and judge whether each change is likely to generalize to other clinical notes or only address a single observed case.

To promote stable schema evolution, the refinement LLM is instructed to address the most frequent feedback items and follow a conservative refinement strategy in which extensions (adding fields or refining existing types) are preferred over deletions, since removed fields may remain relevant to note types not yet observed. Field deletion is permitted only when performed as part of a principled restructuring, such as replacing an underspecified field with a more expressive or semantically precise representation. All revisions are validated against JSON Schema Draft 2020-12 and are constrained to use scalar types (string, number, integer, boolean) or composite types (object, array) with explicit per-field type annotations.

A low temperature of 0.1 is used to encourage deterministic and stable schema updates. The refinement client returns a structured output consisting of (1) the revised schema $\mathcal{S}^{(k)}$, (2) an explicit list of modifications relative to $\mathcal{S}^{(k-1)}$, and (3) a natural-language rationale describing the refinement decisions.

The structural delta between consecutive versions is tracked using JSON structural diffing:

$$\Delta^{(k)} = \mathrm{diff}\Big(\mathcal{S}^{(k-1)}, \mathcal{S}^{(k)}\Big) \tag{6}$$

For each cluster c_k, $k = 1, \ldots, K$, the schema is updated as:

$$\mathcal{S}^{(k)} = \textsc{Refine}\Big(\mathcal{S}^{(k-1)},\ \textsc{Aggregate}\Big(\{\textsc{Evaluate}(n_i, \mathcal{S}^{(k-1)})\}_{n_i \in c_k}\Big)\Big), \tag{7}$$

starting from the initial schema $\mathcal{S}^{(0)}$ and yielding the final schema $\mathcal{S}^* = \mathcal{S}^{(K)}$.

4.3.5 Memory Management

Across the stages above, memory management is structured to prevent contamination between clusters while supporting consistent critique within a cluster. Within cluster c_k, the evaluator's quality-scoring and feedback-generation calls share a batch-level conversation history, which allows the evaluator to maintain a consistent critique style across the notes in a cluster and to recognize recurring patterns. The evaluator's schema-filling sub-call is stateless and does not share this history, ensuring that extraction from one note cannot leak into extraction from another. At each cluster boundary $k \to k+1$, the batch-level memory is explicitly reset:

$$\mathcal{B}^{(k)} \leftarrow \emptyset \quad \forall\, k \in \{1, \ldots, K\}, \tag{8}$$

where $\mathcal{B}^{(k)}$ denotes the evaluator's batch-level conversation buffer prior to processing cluster c_k. The refiner client is fully stateless: each refinement call receives only the current schema and the batch aggregation, with no persistent conversational context. This separation of stateful evaluation within a batch from stateless refinement across batches reflects the assumption that batch-internal consistency is valuable while cross-batch independence is essential for stable schema evolution.

4.3.6 Final Schema Application

After the final cluster has been processed, the resulting schema $\mathcal{S}^* = \mathcal{S}^{(K)}$ is held fixed and applied to the validation subset ($N_{\text{val}} = 51$, §3.2.3). A separate stateless filler client extracts a structured document from each validation note conditioned on $\mathcal{S}^*$ and a single in-context reference example drawn from the highest-scoring structured document produced during refinement. The filler operates under the same extraction rules as §4.3.2, extracting only information explicitly stated in the note, omitting fields with no source information, and disallowing inference. The resulting structured documents are the artefacts evaluated in §6 against the summarization and ICL baselines.

5 Experimental Setup

5.1 Models and Inference

All experiments were implemented in Python 3.12 and executed on a workstation equipped with 2 NVIDIA H200 GPUs. LLM inference was performed using SGLang [58], which provides high-throughput batched generation and supports JSON-Schema-constrained decoding, eliminating the need for post-hoc output repair. We evaluate 8 open-weight models ranging from 1 to 109 billion parameters (Table 3). This range enables a controlled analysis of quality–efficiency trade-off across the proposed framework and all baselines.

The extractive baseline uses Bio_ClinicalBERT with GPU inference. The ICL baseline is run exclusively with Llama-4-Scout, selected for its extended context window, enabling the complete refinement subset together with the schema-generation prompt to fit within a single inference context. All abstractive and iterative-refinement experiments are repeated across the remaining models in Table 3.

Table 3 Open-weight LLMs evaluated in this study. Model sizes are reported in billions of parameters (B). Owing to its extended context window, Llama-4-Scout is used only for the ICL baseline. All other models are evaluated on both abstractive summarization and iterative schema refinement.

Model	Params (B)	Use
Gemma-3-1B	1	Abstractive, Iterative
Qwen3-4B	4	Abstractive, Iterative
Llama-3.1-8B	8	Abstractive, Iterative
Qwen3.5-9B	9	Abstractive, Iterative
Gemma-3-27B	27	Abstractive, Iterative
Qwen3.5-27B	27	Abstractive, Iterative
Llama-3.3-70B	70	Abstractive, Iterative
Llama-4-Scout	109	ICL, Iterative

Decoding temperature was set to 0.1 for all extraction, schema-filling, and refinement calls to encourage deterministic outputs. For abstractive summarization, each model's default decoding temperature was used to preserve fluency. All schema-constrained generations are validated against a JSON Schema specification by SGLang at decoding time. The validation subset ($N_{\text{val}} = 51$, §3.2.3) is held out across every approach to ensure cross-method comparability.

5.2 Evaluation Methods

Our goal is to determine whether a filled structured document preserve the clinical content of a source note at least as well as free-text summaries while achieving comparable compression. Treating the source note as ground truth, we define faithfulness as

agreement between a generated representation and the source. Faithfulness is decomposed into precision, which measures the absence of hallucinations, and recall, which measures the absence of clinically important omissions.

We evaluate each representation in terms of compression, precision, and recall using both a reference-based evaluator (SelfCheckGPT) and an LLM judge (DeepEval). Because these metrics compare generated content against the source note, each representation is converted to a common textual format before evaluation. Free-text summaries are sentence-segmented, whereas JSON instances are recursively flattened into `key > subkey: value` statements. This conversion preserves the clinical content and schema hierarchy of each extracted field without imposing a prose format, enabling direct comparison of structured representations and free-text summaries against the source note. The evaluation therefore measures the fidelity of the represented clinical information rather than the fluency of its expression.

5.2.1 Compression

A faithful representation is only useful if it is also substantially shorter than the source, thereby reducing the time required for clinicians to review and analyze clinical documentation [59, 60]. This is a key motivation for both automated summarization and structured clinical representation. Brevity and faithfulness, however, pull against each other. A representation can be made perfectly faithful by copying the note verbatim, or maximally brief by discarding clinical content, so compression is informative only when read alongside the faithfulness metrics. We therefore report the word-level compression ratio,

$$\rho_i = \frac{w(\hat{r}_i)}{w(n_i)}, \tag{9}$$

where $w(\cdot)$ is the number of whitespace-delimited words, n_i is the source note, and $\hat{r}_i$ is its generated representation, either a summary $\hat{s}_i$ or a structured document $\hat{d}_i$. Lower ρ_i indicates greater condensation.

5.2.2 SelfCheckGPT-Based Faithfulness

We adopt the SelfCheckGPT framework [40] with the source clinical note as the single reference, applying two complementary scorers.

BERTScore consistency.

For each unit in the flattened output (a sentence or a field statement), the SelfCheck-BERTScore scorer assigns an *inconsistency* score $b_m \in [0, 1]$, computed from the highest BERTScore similarity [61] between the selected unit and each sentence of the source note. A unit that closely matches many parts of the source receives a low b_m; an unsupported unit receives a high b_m. Thus *lower is more faithful*. Following the original work, a unit is *consistent* if $b_m < 0.5$. For a note with M units we report a document-level score:

$$\bar{b}_i = \frac{1}{M} \sum_{m=1}^{M} b_m \tag{10}$$

Given this scorer relies on embedding similarity to source *sentences*, it implicitly favours sentence-like units. Flattened schema statements are telegraphic rather than prose, and therefore tend to score less similar even when the extracted value is correct. We therefore use BERTScore consistency for comparison *within* a paradigm (e.g., across models or across summarization strategies), and rely on the meaning-based metrics below (MQAG and the DeepEval judges) for comparison *between* the summarization and schema paradigms, as these assess factual agreement rather than surface similarity and are robust to this representational difference.

MQAG question-answering consistency.

The Multiple-choice Question Answering and Generation (MQAG) method [41] assesses faithfulness by generating Q multiple-choice questions from the candidate passage and comparing answer probability distributions between the candidate and the reference document. Let P_{cand} and P_{ref} denote the answer distributions induced by the candidate and reference, respectively. We report four distributional divergence metrics:

- **KL-Divergence** (D_{KL}): asymmetric divergence from candidate to reference; unbounded, more sensitive to probability mass shifts.
- **Counting** (D_{cnt}): proportion of questions where the modal answer differs between candidate and reference; $\in [0, 1]$.
- **Hellinger Distance** (D_{H}): symmetric, bounded divergence measure; $\in [0, 1]$.
- **Total Variation** (D_{TV}): maximum pointwise probability difference; $\in [0, 1]$.

All four metrics are negatively oriented, so lower values indicate greater consistency between the generated representation and the source note. We set $Q = 15$ questions per document. Given that MQAG generates its questions from the candidate, it cannot probe content the candidate has omitted; it is therefore a document-level, primarily precision-oriented consistency signal that complements the unit-level BERTScore measure.

5.2.3 DeepEval LLM-as-Judge Metrics

Three judge-based metric families are computed via DeepEval[5] using gpt-5-mini as the evaluator. Each metric returns a score in $[0, 1]$, where higher values indicate better performance and 1 represents the optimum score. The metrics are designed to assess one or both of the failure modes targeted in this study: **hallucination**, referring to claims unsupported by the source (precision-oriented errors), and **omission**, referring to source information absent from the generated representation (recall-oriented errors).

SummarizationMetric.

We use a modified version of DeepEval's SummarizationMetric. As in the original, an *alignment* (precision) score is the fraction of claims extracted from the output that are supported by the source note, so a low value signals **hallucination**. We replace the original's simple closed-question coverage check with a stronger recall measure:

[5] https://github.com/confident-ai/deepeval

the judge generates importance-weighted, open-ended questions from the source note, each with a reference answer, then answers them from the output alone and grades each answer against the reference; the mean grade is the *coverage* (recall) score, so a low value signals **omission**. For structured-document inputs, claim extraction and question generation are instructed to treat the output as a structured record, so empty fields are not counted as claims and surface prose is not assessed. The final summarization score is the harmonic mean (F1).

G-Eval.

G-Eval [39] is a chain-of-thought LLM-as-judge metric configured with explicit evaluation steps rather than a free-form criterion. We instantiate two independent G-Eval metrics, one for *precision* (absence of hallucination, i.e., whether every clinical claim in the output is grounded in the source note) and one for *recall* (absence of omission, i.e., whether every clinically important fact in the source note is retained in the output). Both metrics use an identical four-level rubric structure over $[0, 10]$ (linearly normalised to $[0, 1]$), differing only in whether the graded failure is unsupported or omitted content, so the two scores are directly comparable; the omission check assumes no fixed section structure.

DAGMetric.

DAGMetric scores an output by routing it through a fixed decision tree (a directed acyclic graph) of discrete judgement steps, so that for a given judge and input the scoring path and outcome are deterministic and reproducible. We define two parallel branches that mirror our precision and recall properties and average their scores. The precision branch proceeds in three steps: a `TaskNode` extracts all clinical claims from the output, a `BinaryJudgementNode` checks whether any extracted claim asserts information not supported by the source note, in which case the branch scores 0; otherwise a `NonBinaryJudgementNode` grades how well the claims are grounded in the source on a three-level scale (10, 6, or 2 for strong, partial, or weak grounding). The recall branch proceeds in two steps: a `TaskNode` enumerates the key clinical facts in the source note without assuming any fixed section structure, and a `NonBinaryJudgementNode` then grades the proportion of those facts captured by the output on the same three-level scale (10, 6, or 2 for most, some, or few facts captured). The final DAGMetric score is the mean of the two branch scores, normalised to $[0, 1]$.

5.2.4 Metric Complementarity

The metrics above jointly cover the two faithfulness properties introduced earlier, precision and recall, through complementary ways, namely question answering (SummarizationMetric), rubric-anchored chain-of-thought reasoning (G-Eval), and a constrained decision tree (DAGMetric). Because all three use the same judge model over the same documents, they are not statistically independent; their differing mechanisms nonetheless reduce the methodological bias of any single judge. They are complemented by two reference-based SelfCheck signals: BERTScore consistency, a unit-level precision-oriented measure, and MQAG, a document-level, primarily

Table 4 Evaluation metrics and the faithfulness property each targets. ↑ higher is better, ↓ lower is better.

Metric	Targets	Better
SelfCheck BERTScore	Precision	↓
SelfCheck MQAG	Precision	↓
SummarizationMetric	Precision + Recall	↑
G-Eval	Precision + Recall	↑
DAGMetric	Precision + Recall	↑

precision-oriented measure of answer-distribution agreement. Table 4 summarises this mapping.

All evaluations are run on the separated subset of N_{val} validation notes. Per-document and per-unit scores persist in structured CSV files for downstream statistical analysis and cross-method comparison.

6 Results

We compare the proposed Iterative Schema approach against four baselines on the validation subset N_{val}. Std. Abstractive, CoT Abstractive, and Iterative Schema are evaluated using the same 8 LLMs, enabling direct comparison across model scales. Extractive (BERT-Centroid) and ICL Schema (Llama-4-Scout) serve as single-model reference baselines. Throughout the remainder of this paper, we refer to the approaches using the abbreviated names Extractive, Std. Abstractive, CoT Abstractive, ICL Schema, and Iterative Schema, and identify models by family and parameter size. Table 5 reports the complete results for every evaluated approach–model combination, with inter-judge agreement reported in Table 6.

6.1 Overall Faithfulness Comparison

Iterative Schema achieves the highest mean F1 across all three judge-based metrics (Figure 5, Table 5), with model-averaged scores of 0.939 for SummarizationMetric, 0.926 for G-Eval, and 0.910 for DAGMetric. The strongest summarization baseline (Std. Abstractive) achieves corresponding scores of 0.876, 0.860, and 0.862, while CoT Abstractive reaches 0.768, 0.825, and 0.822. Consequently, Iterative Schema improves upon the strongest summarization baseline by +0.063, +0.066, and +0.047 F1, respectively.

The two single-model reference points fall between the summarization baselines and Iterative Schema. ICL Schema, evaluated on Llama-4-Scout, attains F1 scores of 0.877, 0.817, and 0.876, indicating that structured extraction without iterative refinement is already competitive. The Extractive baseline, evaluated on BERT-Centroid, attains 0.715, 0.814, and 0.872, indicating that an extractive approach can achieve competitive judge-based scores when its precision is high.

Table 5 Per-approach, per-model evaluation results (mean ± standard deviation across notes). F1 columns are higher-is-better (↑); BERTScore (mean unit-level inconsistency) and the four MQAG distributional divergences (D_{KL} KL-divergence, D_{cnt} counting, D_{H} Hellinger, D_{TV} total variation) are lower-is-better (↓). Per-cell best value per column is in **bold**; each multi-model approach block ends with an *(mean)* row averaging that approach's cells.

Approach	Model	Summ. F1↑	G-Eval F1↑	DAG F1↑	BERTScore↓	MQAG D_{KL} ↓	MQAG D_{cnt} ↓	MQAG D_{H} ↓	MQAG D_{TV} ↓
Extractive[1]	BERT-Centroid	0.715 ± .042	0.814 ± .048	0.872 ± .045	**0.127 ± .040**	0.276 ± .123	0.280 ± .118	0.099 ± .021	0.221 ± .029
Std. Abstractive[2]	Llama-3.1-8B	0.853 ± .047	0.869 ± .029	0.861 ± .033	0.592 ± .059	0.432 ± .277	0.306 ± .125	0.134 ± .032	0.231 ± .036
	Llama-3.3-70B	0.898 ± .035	0.848 ± .053	0.819 ± .024	0.517 ± .033	0.268 ± .108	0.237 ± .092	0.096 ± .032	0.188 ± .037
	Llama-4-Scout	0.895 ± .038	0.857 ± .058	0.925 ± .021	0.539 ± .028	0.288 ± .138	0.256 ± .116	0.101 ± .022	0.199 ± .029
	Qwen3-4B	0.875 ± .041	0.866 ± .021	0.823 ± .037	0.583 ± .036	0.289 ± .132	0.259 ± .109	0.101 ± .040	0.197 ± .023
	Qwen3.5-9B	0.935 ± .026	0.858 ± .043	0.935 ± .028	0.658 ± .032	0.218 ± .104	0.224 ± .093	0.074 ± .032	0.171 ± .022
	Qwen3.5-27B	0.940 ± .023	0.907 ± .030	0.837 ± .050	0.607 ± .029	0.271 ± .123	0.239 ± .097	0.094 ± .038	0.190 ± .026
	Gemma-3-1B	0.701 ± .033	0.829 ± .051	0.892 ± .039	0.615 ± .037	0.326 ± .131	0.266 ± .095	0.114 ± .021	0.210 ± .026
	Gemma-3-27B	0.910 ± .037	0.844 ± .039	0.804 ± .023	0.604 ± .024	0.256 ± .103	0.242 ± .070	0.090 ± .034	0.186 ± .020
(mean)		**0.876 ± .035**	**0.860 ± .040**	**0.862 ± .032**	0.589 ± .035	0.294 ± .140	0.254 ± .100	0.101 ± .031	0.196 ± .027
CoT Abstractive[3]	Llama-3.1-8B	0.690 ± .019	0.817 ± .020	0.827 ± .032	0.633 ± .058	0.607 ± .313	0.442 ± .168	0.184 ± .037	0.329 ± .042
	Llama-3.3-70B	0.884 ± .038	0.865 ± .050	0.875 ± .040	0.571 ± .029	0.440 ± .197	0.354 ± .134	0.154 ± .030	0.281 ± .036
	Llama-4-Scout	0.880 ± .043	0.807 ± .012	0.742 ± .029	0.558 ± .029	0.405 ± .167	0.344 ± .123	0.146 ± .028	0.280 ± .034
	Qwen3-4B	0.860 ± .037	0.839 ± .015	0.832 ± .021	0.599 ± .028	0.450 ± .184	0.367 ± .141	0.157 ± .029	0.297 ± .034
	Qwen3.5-9B	0.847 ± .042	0.829 ± .059	0.855 ± .031	0.633 ± .026	0.456 ± .176	0.369 ± .117	0.159 ± .028	0.298 ± .031
	Qwen3.5-27B	0.840 ± .021	0.838 ± .024	0.826 ± .033	0.645 ± .029	0.467 ± .148	0.358 ± .123	0.162 ± .023	0.299 ± .028
	Gemma-3-1B	0.558 ± .036	0.799 ± .038	0.811 ± .025	0.629 ± .027	0.496 ± .204	0.397 ± .108	0.171 ± .030	0.304 ± .034
	Gemma-3-27B	0.585 ± .032	0.804 ± .049	0.811 ± .051	0.633 ± .023	0.562 ± .270	0.400 ± .145	0.193 ± .040	0.329 ± .045
(mean)		0.768 ± .034	0.825 ± .033	0.822 ± .033	0.613 ± .031	0.485 ± .208	0.379 ± .132	0.166 ± .031	0.302 ± .035
ICL Schema[4]	Llama-4-Scout	0.877 ± .025	0.817 ± .037	0.876 ± .033	0.628 ± .055	0.199 ± .121	0.265 ± .123	**0.069 ± .037**	0.190 ± .027
Iterative Schema[5]	Llama-3.1-8B	0.940 ± .039	0.869 ± .053	0.895 ± .037	0.894 ± .027	0.185 ± .082	0.198 ± .066	0.061 ± .024	0.135 ± .033
	Llama-3.3-70B	0.972 ± .043	0.949 ± .031	0.879 ± .022	0.869 ± .021	0.178 ± .068	0.191 ± .071	0.059 ± .021	0.134 ± .029
	Llama-4-Scout	0.980 ± .043	0.957 ± .032	0.886 ± .022	0.833 ± .021	0.171 ± .069	0.183 ± .072	0.057 ± .021	0.129 ± .029
	Qwen3-4B	0.975 ± .033	0.916 ± .036	**0.972 ± .053**	0.759 ± .030	**0.107 ± .046**	**0.144 ± .066**	0.038 ± .017	**0.104 ± .028**
	Qwen3.5-9B	0.942 ± .027	0.915 ± .017	0.868 ± .033	0.880 ± .021	0.400 ± .122	0.321 ± .078	0.101 ± .025	0.189 ± .030
	Qwen3.5-27B	**0.989 ± .036**	0.964 ± .039	0.908 ± .048	0.911 ± .033	0.231 ± .091	0.220 ± .071	0.065 ± .022	0.147 ± .028
	Gemma-3-1B	0.784 ± .129	**0.967 ± .148**	0.938 ± .063	0.875 ± .025	0.440 ± .201	0.261 ± .076	0.138 ± .027	0.203 ± .025
	Gemma-3-27B	0.927 ± .035	0.870 ± .024	0.932 ± .021	0.816 ± .029	0.150 ± .055	0.174 ± .066	0.051 ± .017	0.123 ± .025
(mean)		**0.939 ± .048**	**0.926 ± .047**	**0.910 ± .037**	0.855 ± .026	0.233 ± .092	0.211 ± .071	0.071 ± .022	0.145 ± .028

[1]Baseline 1: Extractive summarization (BERT-Centroid).

[2]Baseline 2: Standard abstractive summarization.

[3]Baseline 3: Chain-of-thought abstractive summarization.

[4]Baseline 4: In-context learning schema extraction.

[5]Proposed iterative feedback–refinement.

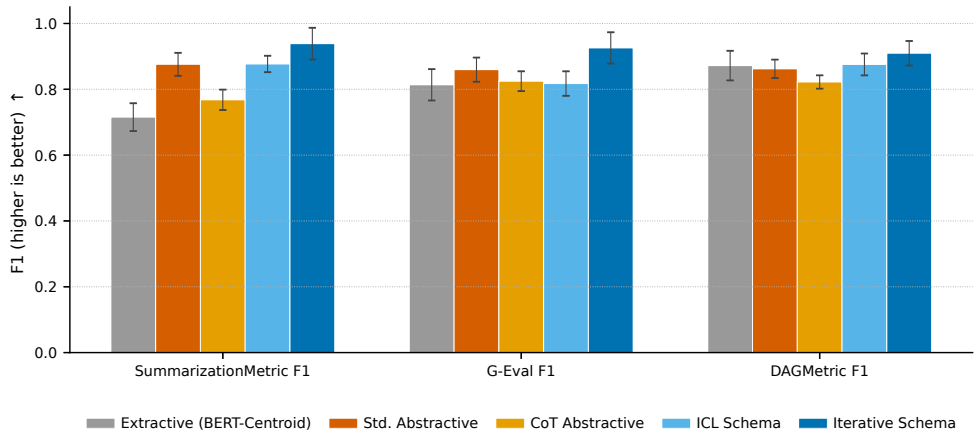


Fig. 5 Mean F1 by approach across the three judge-based metrics (higher is better). Error bars are one standard deviation across the model cells contributing to each approach mean. Iterative Schema approach leads on all three metrics; BERT-Centroid extractive and ICL schema reference points are single-model results.

6.2 Precision and Recall Decomposed

We decompose each F1 into precision, the absence of hallucination, and recall, the absence of omission. This decomposition indicates that Iterative Schema achieves high precision and recall simultaneously rather than from a favourable trade-off between them. Across the three judges, its precision (0.942/0.922/0.915 on SummarizationMetric, G-Eval, and DAGMetric) and recall (0.762/0.942/0.896) both remain high, and it is the only approach to retain this dual strength under every judge. The Extractive baseline makes the underlying trade-off explicit: because it copies source sentences verbatim, it cannot hallucinate and so attains the highest precision of any approach (0.994/0.988/0.950), yet verbatim selection offers no guarantee of completeness, leaving it with the lowest recall (0.581/0.809/0.808). High precision alone is thus insufficient, as a representation may avoid hallucination entirely while still omitting much of the note. Iterative Schema, by contrast, combines precision comparable to the strongest summarization baselines with the highest recall on the two rubric-based judges (G-Eval, 0.942 against 0.839 for Std. Abstractive and 0.837 for CoT; DAGMetric, 0.896 against 0.856 and 0.849). Consequently, its superior F1 scores are primarily driven by improved recall without sacrificing precision. This pattern is evident per judge in Figure 6, and the G-Eval precision/recall plane (Figure 7) locates each (approach, model) cell on iso-F_1 contours, with the schema cells reaching the upper-right region by sustaining precision while improving recall.

6.3 Consistency Across Models

A model-averaged result could conceal an advantage concentrated in one or two strong models. To test for this, we compute the per-model difference $\Delta = \text{IterSchema} - \max(\text{CoT}, \text{Std.})$ on SummarizationMetric F1 for every model that produced both an

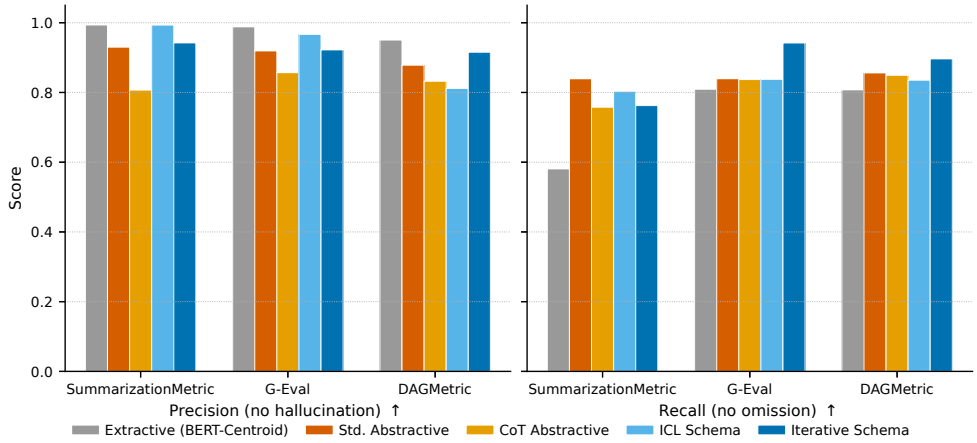


Fig. 6 Per-judge precision and recall scores by approach. The recall gain of Iterative Schema is clearest on G-Eval and DAGMetric; on the QA-based SummarizationMetric the schema leads on precision but not recall.

Iterative Schema and a summarization result (Figure 8). Iterative Schema exceeds the stronger summarization baseline on every model, with Δ ranging from 0.007 on Qwen3.5-9B to 0.100 on Qwen3-4B.

The advantage holds across models, although faithfulness did not increase monotonically with model size. Qwen3-4B attains a SummarizationMetric F1 of 0.975 on Iterative Schema, which exceeds the much larger Llama-3.3-70B at 0.972 and approaches Qwen3.5-27B at 0.989, and Llama-3.1-8B attains 0.940. These results indicate that mid-sized models can match or exceed larger ones under the right inference approach. The smallest model, Gemma-3-1B, is the one exception. It produces a positive difference of 0.083, near the across-model median, yet produces the lowest absolute scores of any model across all approaches and judges, for example a CoT SummarizationMetric F1 of 0.558 and a Std. Abstractive SummarizationMetric F1 of 0.701 (Table 5). Because Gemma-3-1B is the only model whose absolute scores consistently lag, these findings suggest that limited model capacity may contribute more to the lower performance than parameter count alone.

6.4 Compression

The faithfulness gains do not come at the expense of brevity. The mean word compression ratio of Iterative Schema is 0.345, approximately 35% of the source length, against 0.635 for Std. Abstractive and 0.665 for CoT Abstractive. On this validation set, the structured representation is therefore both more compact and more faithful than the summarization baselines. Figure 9 shows this result, with the schema cells clustering in the upper-left region of the compression–faithfulness plane, where faithfulness is high and length is low.

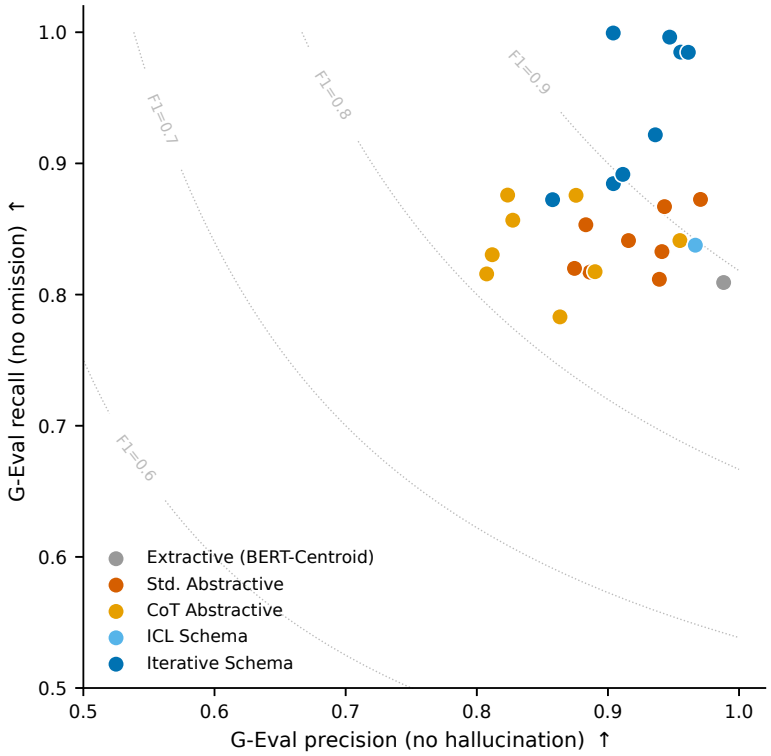


Fig. 7 G-Eval precision (no hallucination) against G-Eval recall (no omission), with one point per (approach, model) cell. Dotted lines are iso-F_1 contours. Iterative Schema occupies the upper-right corner; summarization points lie predominantly on lower-recall iso-curves.

6.5 Embedding-Based Faithfulness (MQAG)

The MQAG measures agree with the judge-based metrics. All four divergences are negatively oriented (lower indicates higher consistency), and Iterative Schema attains the lowest mean Hellinger distance (0.071) and total variation (0.145) of any approach, against 0.166/0.302 for CoT and 0.101/0.196 for Std. Abstractive (Table 5). KL divergence and counting disagreement, though on different scales, rank the approaches identically, again placing Iterative Schema first. The single-model ICL Schema point records the lowest divergences overall (0.069 Hellinger, 0.199 KL), consistent with schema-based representations preserving answer-distribution agreement with the source note better than free-text summarization. Within the schema paradigm, ICL Schema's edge suggests a single in-context exemplar already recovers most of the distributional consistency, so iterative refinement contributes mainly to the judge-based F1 gains.

Moreover, SelfCheckBERTScore shows the schema with higher mean unit-level inconsistency than the summarization baselines (0.858 vs 0.589/0.613). As discussed in Section 5.2, this similarity is computed against source sentences and so penalises telegraphic field statements. We therefore interpret BERTScore only within paradigms

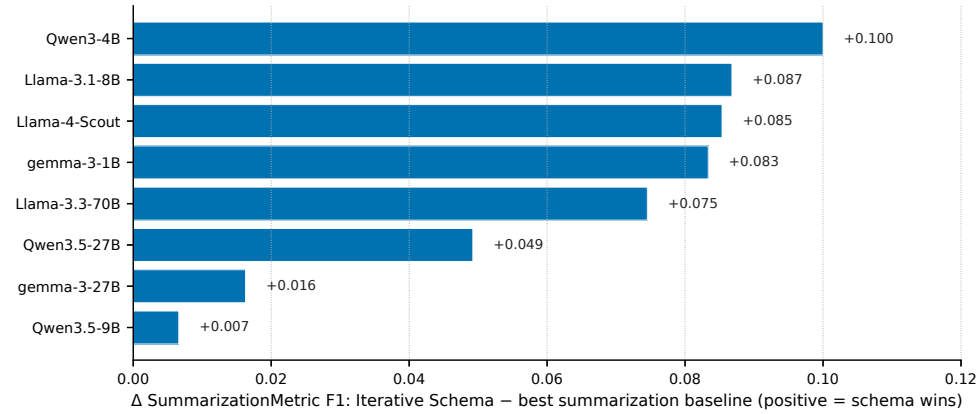


Fig. 8 Per-model advantage of Iterative Schema over the stronger of the two summarization baselines, measured as the difference in SummarizationMetric F1. All deltas are positive across 8 paired models.

(across models or summarization strategies) and use MQAG for the schema-versus-summary comparison.

6.6 Agreement Between Judges and Reference-Based Metrics

Table 6 reports Spearman rank correlations (r_s) between the three judge-based F1 scores and the two main MQAG divergences, computed across all approach–model cells. The three F1 scores are positively correlated with each other ($r_s \in [0.42, 0.72]$), and negatively correlated with the MQAG divergences ($r_s \in [-0.82, -0.53]$), as expected given that MQAG is negatively oriented. The strongest among-judge correlations are SummarizationMetric F1 with G-Eval F1 ($r_s = 0.72$) and SummarizationMetric F1 with MQAG Hellinger ($r_s = -0.82$); the weakest is SummarizationMetric F1 with DAGMetric F1 ($r_s = 0.42$), which is consistent with the DAG metric's discrete decision-node structure producing more coarse-grained scores than the continuous rubric and QA-based metrics. The two MQAG divergences are near-perfectly correlated ($r_s = 0.99$), as expected for two normalised distances over the same answer distributions. Together these correlations indicate the schema advantage is not an artefact of one judging mechanism, since four metrics with four different computational structures agree on its direction.

7 Discussion

The observed improvements are consistent with a representation-centric refinement strategy in which the schema, rather than the generated output, is the object of optimization. Unlike summarization methods that optimize each output independently, the proposed framework iteratively refines a shared schema that guides all subsequent extractions, so improvements accumulate throughout refinement and propagate to all subsequent extractions. The feedback–refinement architecture grounds these updates in structured feedback derived directly from the source notes, reducing reliance on

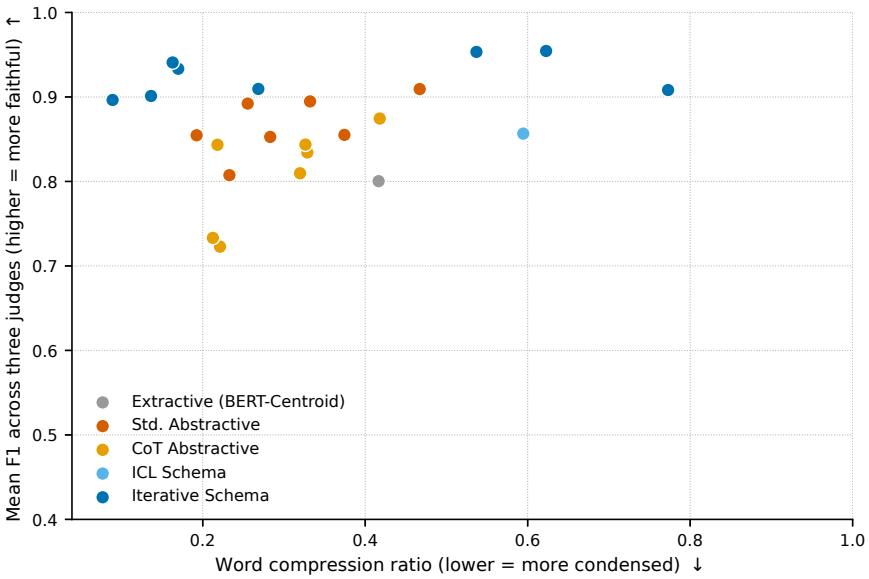


Fig. 9 Faithfulness–compression plane. Each point is one (approach, model) cell. The vertical axis is the mean of the three judge-based F1 scores; the horizontal axis is the word compression ratio (lower = more condensed). The schema points (deep blue) form a band at higher faithfulness than the summarization points while remaining at comparable or greater compression.

Table 6 Spearman rank correlation (r_s) between metrics across all approaches and models. MQAG divergences are negatively oriented, so their correlations with the F1 scores are expected to be negative. Strong agreement across mechanisms (question-answering, rubric, decision tree, and distributional) supports the robustness of the headline result.

	Summ. F1	G-Eval F1	DAG F1	MQAG D_{H}	MQAG D_{KL}
Summarization F1	—	0.72	0.42	−0.82	−0.78
G-Eval F1	0.72	—	0.62	−0.61	−0.53
DAGMetric F1	0.42	0.62	—	−0.57	−0.55
MQAG D_{H}	−0.82	−0.61	−0.57	—	0.99
MQAG D_{KL}	−0.78	−0.53	−0.55	0.99	—

Values are Spearman rank correlation coefficients (r_s). D_{H}: Hellinger distance; D_{KL}: KL divergence. MQAG divergences are negatively oriented, so negative correlations with the F1 scores are expected.

unguided self-correction. This design aligns with evidence that intrinsic self-correction is unreliable without external grounding [31, 32] and with the role-separation strategy used in Reflexion and related agentic frameworks [33].

The gains can be attributed to complementary changes in precision and recall. Across the abstractive baselines, Iterative Schema maintains comparable precision while improving recall (Figure 7), indicating that faithfulness improves through reduced omission rather than increased hallucination. This behavior reflects a property of structured representations: a schema explicitly defines the information fields

to be populated, encouraging systematic coverage, whereas free-text summarization balances completeness against fluency and brevity and thereby increases the risk of omission. The baseline comparisons further clarify this behavior. The extractive baseline attains the highest precision (Figure 7) by copying source text but records the lowest recall, showing that avoiding hallucination alone is insufficient when clinically relevant information is omitted; the non-iterative ICL baseline (Figure 5) suggests that much of the benefit already arises from the structured representation itself, with iterative refinement contributing further gains in faithfulness. Prior work similarly identifies omission as a persistent limitation of clinical summarization, even when outputs are fluent and factually plausible [34, 62], and existing methods address it mainly through faithfulness-aware decoding or post-generation correction [36, 38]; our results suggest that refining the representation itself offers an alternative, reducing omission during extraction rather than correcting it afterwards.

Comparisons across the evaluated baselines provide further insight into where the observed gains originate and indicate that these gains are not incidental. First, they are consistent across model scales, suggesting that the refinement strategy contributes more than model size. Mid-sized models in the 4B–9B range perform comparably to larger models, consistent with prior findings that appropriately adapted smaller models can match stronger systems on clinical tasks [15]. Second, the advantage holds across evaluation metrics with different mechanisms, so it is unlikely to be an artefact of any single scoring method, although the judges are not statistically independent (§5.2.4).

Beyond clinical documentation, these results point to the value of optimizing the representation itself rather than only the generated outputs. From a knowledge-engineering perspective, the representation is an important optimization target alongside model architecture and prompting strategy: iteratively improving a shared schema allows refinements to transfer to all future extractions, yielding a reusable representation that is largely independent of the underlying foundation model. Because the induced schema is an explicitly typed, machine-readable structure, the approach applies to other text-to-structure tasks, such as knowledge-graph construction and ontology population, where faithful conversion of unstructured text is required [63]. In the clinical domain specifically, more faithful structured representations may reduce omission errors that propagate into downstream decision-support systems and other secondary uses of clinical data, and the resulting outputs are naturally compatible with interoperability standards and common data models.

Several limitations should be considered when interpreting these findings. All three judge-based metrics relied on a single evaluation model (gpt-5-mini), which ensured a consistent evaluator across approaches while varying the scoring mechanism through question-answering, rubric-based, and decision-tree frameworks; nonetheless, LLM judges may exhibit systematic preference for structured or model-generated text [39, 42], and although agreement across metrics mitigates this risk, validation with mechanistically distinct evaluators and human experts remains necessary. The evaluation was also reference-free, measuring faithfulness against source clinical notes rather than clinician-authored references, which reflects real-world constraints but may encode a particular interpretation of clinical relevance. In addition, experiments were

conducted on a single corpus of English transcribed clinical notes, which may not capture the structural variability, abbreviation density, and documentation practices of native EHR data or of other domains, limiting generalizability; evaluating the framework on additional corpora, including non-clinical text, is an important direction for future work. Hardware constraints further introduced a selection bias by excluding very short and very long notes, restricting evaluation to moderate-length documents, which motivates future work on processing arbitrarily long notes without exclusion, for example through hierarchical strategies that structure each section independently before combining them into a note-level output [24]. Finally, although the framework substantially reduces hallucination, it does not eliminate it, and clinical deployment should therefore include appropriate oversight. Future work should incorporate field-level uncertainty estimation and provenance tracking to source sentences [64], enabling more reliable interpretation and verification of extracted clinical information.

8 Conclusion

This work reframes structured documentation as a problem of adaptive schema induction rather than instance generation from a fixed template or document summarization, demonstrating that refining the schema is an effective alternative to refining generated text. By making the representation the target of optimization, the proposed framework produces structured documents that are more faithful and more compact than free-text summarization, a result that holds across 8 open-weight models and several evaluation metrics with different underlying mechanisms. The framework attains precision equal to or better than the strongest summarization baseline while recovering more of the source content, so it reduces omission without admitting more hallucination. As the schema evolves independently of any single model and requires neither labeled data nor task-specific fine-tuning, the approach provides a scalable and model-agnostic route to faithful structured representation. Although demonstrated on clinical notes as a high-stakes, high-heterogeneity testbed, the method is domain-general, and its induced machine-readable schemas are directly applicable to other text-to-structure tasks. Overall, the results indicate that the principal leverage in structured extraction lies less in iterative prompting, output refinement, or free-text summarization, and more in treating the schema itself as the object of faithful representation.

9 Data Availability

The data used in this research are derived from the publicly available MTSamples dataset. Details of the data sources, preprocessing steps, and experimental settings are provided in the manuscript. To ensure reproducibility, the code and datasets are publicly available at: https://github.com/Eslam21/Iterative-Feedback-Refinement.

10 Ethics declarations

10.1 Competing Interests

The authors declare no competing interests.

10.2 Clinical Trial Number

Not applicable.

10.3 Ethics Approval and Consent to Participate

Not applicable.

10.4 Consent for Publication

Not applicable.

References

[1] Rosenbloom, S.T., Denny, J.C., Xu, H., Lorenzi, N., Stead, W.W., Johnson, K.B.: Data from clinical notes: a perspective on the tension between structure and flexible documentation. Journal of the American Medical Informatics Association **18**(2), 181–186 (2011) https://doi.org/10.1136/jamia.2010.007237 . Accessed 2026-04-14

[2] Cohen, G.R., Friedman, C.P., Ryan, A.M., Richardson, C.R., Adler-Milstein, J.: Variation in Physicians' Electronic Health Record Documentation and Potential Patient Harm from That Variation. Journal of General Internal Medicine **34**(11), 2355–2367 (2019) https://doi.org/10.1007/s11606-019-05025-3

[3] Li, T., Zhang, G., Do, Q.D., Yue, X., Chen, W.: Long-context LLMs Struggle with Long In-context Learning. arXiv. arXiv:2404.02060 [cs] (2024). https://doi.org/10.48550/arXiv.2404.02060 . http://arxiv.org/abs/2404.02060 Accessed 2025-11-19

[4] Moramarco, F., Papadopoulos Korfiatis, A., Perera, M., Juric, D., Flann, J., Reiter, E., Belz, A., Savkov, A.: Human Evaluation and Correlation with Automatic Metrics in Consultation Note Generation. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.) Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 5739–5754. Association for Computational Linguistics, Dublin, Ireland (2022). https://doi.org/10.18653/v1/2022.acl-long.394 . https://aclanthology.org/2022.acl-long.394/ Accessed 2026-05-11

[5] Williams, C.Y.K., Bains, J., Tang, T., Patel, K., Lucas, A.N., Chen, F., Miao, B.Y., Butte, A.J., Kornblith, A.E.: Evaluating Large Language Models for Drafting Emergency Department Discharge Summaries. medRxiv. Pages: 2024.04.03.24305088 (2024). https://doi.org/10.1101/2024.04.03.24305088 . https://www.medrxiv.org/content/10.1101/2024.04.03.24305088v1 Accessed 2025-09-27

[6] Maynez, J., Narayan, S., Bohnet, B., McDonald, R.: On Faithfulness and Factuality in Abstractive Summarization. In: Jurafsky, D., Chai, J., Schluter, N.,

Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 1906–1919. Association for Computational Linguistics, Online (2020). https://doi.org/10.18653/v1/2020.acl-main.173 . https://aclanthology.org/2020.acl-main.173/ Accessed 2025-11-18

[7] Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y.J., Madotto, A., Fung, P.: Survey of Hallucination in Natural Language Generation. ACM Computing Surveys **55**(12), 248–124838 (2023) https://doi.org/10.1145/3571730 . Accessed 2026-06-10

[8] Oukelmoun, A., Semmar, N., Chalendar, G., Cormi, C., Oukelmoun, M., Vibert, E., Allard, M.-A.: Detecting Omissions in LLM-Generated Medical Summaries. In: Potdar, S., Rojas-Barahona, L., Montella, S. (eds.) Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track, pp. 325–337. Association for Computational Linguistics, Suzhou (China) (2025). https://doi.org/10.18653/v1/2025.emnlp-industry.22 . https://aclanthology.org/2025.emnlp-industry.22/ Accessed 2025-11-18

[9] Sadruddin, S., D'Souza, J., Poupaki, E., Watkins, A., Babaei Giglou, H., Rula, A., Karasulu, B., Auer, S., Mackus, A., Kessels, E.: LLMs4SchemaDiscovery: A Human-in-the-Loop Workflow for Scientific Schema Mining with Large Language Models. In: Curry, E., Acosta, M., Poveda-Villalón, M., Erp, M., Ojo, A., Hose, K., Shimizu, C., Lisena, P. (eds.) The Semantic Web, pp. 244–261. Springer, Cham (2025). https://doi.org/10.1007/978-3-031-94578-6_14

[10] Shrimal, A., Jain, A., Chowdhury, S., Yenigalla, P.: PARSE: LLM driven schema optimization for reliable entity extraction. In: Potdar, S., Rojas-Barahona, L., Montella, S. (eds.) Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track, pp. 2749–2763. Association for Computational Linguistics, Suzhou (China) (2025). https://doi.org/10.18653/v1/2025.emnlp-industry.184 . https://aclanthology.org/2025.emnlp-industry.184/

[11] Hier, D.B., Carrithers, M.D., Do, T.S., Obafemi-Ajayi, T.: Efficient Standardization of Clinical Notes using Large Language Models. arXiv. arXiv:2501.00644 [cs] (2024). https://doi.org/10.48550/arXiv.2501.00644 . http://arxiv.org/abs/2501.00644 Accessed 2025-09-28

[12] Huang, J., Yang, D.M., Rong, R., Nezafati, K., Treager, C., Chi, Z., Wang, S., Cheng, X., Guo, Y., Klesse, L.J., Xiao, G., Peterson, E.D., Zhan, X., Xie, Y.: A critical assessment of using ChatGPT for extracting structured data from clinical notes. npj Digital Medicine **7**(1), 106 (2024) https://doi.org/10.1038/s41746-024-01079-8 . Accessed 2026-05-11

[13] Bai, J., Fan, W., Hu, Q., Zong, Q., Li, C., Tsang, H.T., Luo, H., Yim, Y., Huang, H., Zhou, X., Qin, F., Zheng, T., Peng, X., Yao, X., Yang, H., Wu, L., Yi, J., Zhang, G., Chen, R., Song, Y.: AutoSchemaKG: Autonomous knowledge graph construction through dynamic schema induction from web-scale corpora.

In: Liakata, M., Moreira, V.P., Zhang, J., Jurgens, D. (eds.) Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 20557–20584. Association for Computational Linguistics, San Diego, California, United States (2026). https://doi.org/10.18653/v1/2026.acl-long.942 . https://aclanthology.org/2026.acl-long.942/

[14] Zhang, B., Soh, H.: Extract, define, canonicalize: An LLM-based framework for knowledge graph construction. In: Al-Onaizan, Y., Bansal, M., Chen, Y.-N. (eds.) Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 9820–9836. Association for Computational Linguistics, Miami, Florida, USA (2024). https://doi.org/10.18653/v1/2024.emnlp-main.548 . https://aclanthology.org/2024.emnlp-main.548/

[15] Van Veen, D., Van Uden, C., Blankemeier, L., Delbrouck, J.-B., Aali, A., Bluethgen, C., Pareek, A., Polacin, M., Reis, E.P., Seehofnerová, A., Rohatgi, N., Hosamani, P., Collins, W., Ahuja, N., Langlotz, C.P., Hom, J., Gatidis, S., Pauly, J., Chaudhari, A.S.: Adapted large language models can outperform medical experts in clinical text summarization. Nature Medicine **30**(4), 1134–1142 (2024) https://doi.org/10.1038/s41591-024-02855-5 . Accessed 2025-09-28

[16] Madaan, A., Tandon, N., Gupta, P., Hallinan, S., Gao, L., Wiegreffe, S., Alon, U., Dziri, N., Prabhumoye, S., Yang, Y., Gupta, S., Majumder, B.P., Hermann, K., Welleck, S., Yazdanbakhsh, A., Clark, P.: SELF-REFINE: iterative refinement with self-feedback. In: Proceedings of the 37th International Conference on Neural Information Processing Systems. NIPS '23, pp. 46534–46594. Curran Associates Inc., Red Hook, NY, USA (2023)

[17] Wadhwa, M., Zhao, X., Li, J.J., Durrett, G.: Learning to Refine with Fine-Grained Natural Language Feedback. In: Al-Onaizan, Y., Bansal, M., Chen, Y.-N. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2024, pp. 12281–12308. Association for Computational Linguistics, Miami, Florida, USA (2024). https://doi.org/10.18653/v1/2024.findings-emnlp.716 . https://aclanthology.org/2024.findings-emnlp.716/ Accessed 2026-04-14

[18] Xu, D., Chen, W., Peng, W., Zhang, C., Xu, T., Zhao, X., Wu, X., Zheng, Y., Wang, Y., Chen, E.: Large language models for generative information extraction: a survey. Frontiers of Computer Science **18**(6) (2024) https://doi.org/10.1007/s11704-024-40555-y

[19] Evans, C.S., Bunn, B., Reeder, T., Patterson, L., Gertsch, D., Medford, R.J.: Standardization of Emergency Department Clinical Note Templates: A Retrospective Analysis across an Integrated Health System. Applied Clinical Informatics **15**(02), 397–403 (2024) https://doi.org/10.1055/a-2301-7496 . Accessed 2026-05-11

[20] Augustine, A., Barbosa Carroll, C., Saner, A., Pier, J., Lavitschke, M., Hollnagel, F., Emamekhoo, H.: Breaking barriers in oncology data capture: Use of

standardized oncology-specific note templates to enhance accuracy and completeness of structured data at the point of care. JCO Oncology Practice **21**(10_suppl), 619–619 (2025) https://doi.org/10.1200/OP.2025.21.10_suppl.619 . Accessed 2026-05-11

[21] Muhammed, A., Abdalgadir Hamdnaalla, M.A., Fakher Aldeen Noman, F.A.R., Mohamed, M., Mohammed Ali, M.A., Abdallah Awadelkareem, M.M., Hamid Mohammed, M.I., Awad, I., Abdualhi Ali, R.E., Noureldin Mustafa, N.M., Abdelhay Mahmoud, M.E., Bakri Mohamed, R.A., Elawad Wedatalla, M.A., Abdalla Ibrahim, A.I., Abdelhay Mahmoud, M.E., Ali Gagar, S.M., Fagier Mohamed, A.E., Mohammed Hamed, F.J.: Evaluating and Improving the Quality of the Follow-Up Checklist at Dongola Specialised Hospital: A Clinical Audit. Cureus **16**(12), 75266 (2024) https://doi.org/10.7759/cureus.75266

[22] Vetter, C.D., Kim, J.H.: Impact of implementing structured note templates on data capture for hernia surgery. Health Information Management Journal **52**(2), 87–91 (2023) https://doi.org/10.1177/18333583211001584 . Accessed 2026-05-11

[23] Frei, J., Feldhus, N., Raithel, L., Roller, R., Meyer, A., Kramer, F.: Infherno: End-to-end Agent-based FHIR Resource Synthesis from Free-form Clinical Notes. In: Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 3: System Demonstrations), pp. 163–174 (2026). https://doi.org/10.18653/v1/2026.eacl-demo.13 . arXiv:2507.12261 [cs.CL]. http://arxiv.org/abs/2507.12261 Accessed 2026-06-09

[24] Keerthana, G., Gupta, M.: CLI-RAG: A Retrieval-Augmented Framework for Clinically Structured and Context Aware Text Generation with LLMs. arXiv. arXiv:2507.06715 [cs] (2025). https://doi.org/10.48550/arXiv.2507.06715 . http://arxiv.org/abs/2507.06715 Accessed 2026-01-21

[25] Keerthana, G., Gupta, M.: TriMedPrompt: A unified prompting framework for realistic and layout-conformant clinical progress note synthesis. Journal of Biomedical Informatics **171**, 104927 (2025) https://doi.org/10.1016/j.jbi.2025.104927 . Accessed 2026-01-21

[26] Folstad, M., Augustine, A., Hollnagel, F., Carroll, C., Saner, A., Pier, J., Lavitschke, M., Kleinschmidt, P., Twedt, H., Tevaarwerk, A.J., Emamekhoo, H.: Improving quality of oncology (onc) documentation and enhancing structured data collection using a standardized onc note template. Journal of Clinical Oncology **42**(16_suppl), 11128–11128 (2024) https://doi.org/10.1200/JCO.2024.42.16_suppl.11128 . Accessed 2026-05-11

[27] Gireesh, H., Shukla, L., Shivaprakash, P., Mukherjee, A., Chand, P., Murthy, P.: Language Models for Standardising Clinical Notes and Information Extraction in Addiction Psychiatry—An Empirical Study. Drug and Alcohol Review **45**(1), 70059 (2026) https://doi.org/10.1111/dar.70059 . _eprint:

https://onlinelibrary.wiley.com/doi/pdf/10.1111/dar.70059. Accessed 2026-01-21

[28] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E.H., Le, Q.V., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. NIPS '22, pp. 24824–24837. Curran Associates Inc., Red Hook, NY, USA (2022)

[29] Ji, Z., Yu, T., Xu, Y., Lee, N., Ishii, E., Fung, P.: Towards Mitigating LLM Hallucination via Self Reflection. In: Bouamor, H., Pino, J., Bali, K. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 1827–1843. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.findings-emnlp.123 . https://aclanthology.org/2023.findings-emnlp.123/ Accessed 2025-11-17

[30] Huang, J., Chen, X., Mishra, S., Zheng, H.S., Yu, A.W., Song, X., Zhou, D.: Large Language Models Cannot Self-Correct Reasoning Yet (2024). https://arxiv.org/abs/2310.01798

[31] Kamoi, R., Zhang, Y., Zhang, N., Han, J., Zhang, R.: When Can LLMs Actually Correct Their Own Mistakes? A Critical Survey of Self-Correction of LLMs. Transactions of the Association for Computational Linguistics **12**, 1417–1440 (2024) https://doi.org/10.1162/tacl_a_00713 . Accessed 2026-07-02

[32] Wu, Z., Zeng, Q., Zhang, Z., Tan, Z., Shen, C., Jiang, M.: Large Language Models Can Self-Correct with Key Condition Verification. In: Al-Onaizan, Y., Bansal, M., Chen, Y.-N. (eds.) Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 12846–12867. Association for Computational Linguistics, Miami, Florida, USA (2024). https://doi.org/10.18653/v1/2024.emnlp-main.714 . https://aclanthology.org/2024.emnlp-main.714/ Accessed 2026-07-02

[33] Shinn, N., Cassano, F., Berman, E., Gopinath, A., Narasimhan, K., Yao, S.: Reflexion: Language Agents with Verbal Reinforcement Learning. arXiv. arXiv:2303.11366 [cs.AI] (2023). https://doi.org/10.48550/arXiv.2303.11366 . http://arxiv.org/abs/2303.11366 Accessed 2026-07-02

[34] Asgari, E., Montaña-Brown, N., Dubois, M., Khalil, S., Balloch, J., Yeung, J.A., Pimenta, D.: A framework to assess clinical safety and hallucination rates of LLMs for medical text summarisation. npj Digital Medicine **8**(1), 274 (2025) https://doi.org/10.1038/s41746-025-01670-7 . Accessed 2025-09-23

[35] Lin, C.-Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Text Summarization Branches Out, pp. 74–81. Association for Computational Linguistics, Barcelona, Spain (2004). https://aclanthology.org/W04-1013/ Accessed 2025-10-02

[36] Wan, D., Liu, M., McKeown, K., Dreyer, M., Bansal, M.: Faithfulness-Aware Decoding Strategies for Abstractive Summarization. In: Vlachos, A., Augenstein, I. (eds.) Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pp. 2864–2880. Association for Computational Linguistics, Dubrovnik, Croatia (2023). https://doi.org/10.18653/v1/2023.eacl-main.210 . https://aclanthology.org/2023.eacl-main.210/ Accessed 2025-11-18

[37] Wan, D., Bansal, M.: Evaluating and Improving Factuality in Multimodal Abstractive Summarization. arXiv. arXiv:2211.02580 [cs] (2022). https://doi.org/10.48550/arXiv.2211.02580 . http://arxiv.org/abs/2211.02580 Accessed 2025-11-18

[38] Li, S., Zhang, Y., Deng, C., Niu, Y., Zhao, H.: Better Late Than Never: Model-Agnostic Hallucination Post-Processing Framework Towards Clinical Text Summarization. In: Ku, L.-W., Martins, A., Srikumar, V. (eds.) Findings of the Association for Computational Linguistics: ACL 2024, pp. 995–1011. Association for Computational Linguistics, Bangkok, Thailand (2024). https://doi.org/10.18653/v1/2024.findings-acl.59 . https://aclanthology.org/2024.findings-acl.59/ Accessed 2025-11-15

[39] Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., Zhu, C.: G-Eval: NLG Evaluation using Gpt-4 with Better Human Alignment. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 2511–2522. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.emnlp-main.153 . https://aclanthology.org/2023.emnlp-main.153/ Accessed 2026-05-12

[40] Manakul, P., Liusie, A., Gales, M.: SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models. In: Bouamor, H., Pino, J., Bali, K. (eds.) Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 9004–9017. Association for Computational Linguistics, Singapore (2023). https://doi.org/10.18653/v1/2023.emnlp-main.557 . https://aclanthology.org/2023.emnlp-main.557/ Accessed 2026-05-12

[41] Manakul, P., Liusie, A., Gales, M.J.F.: MQAG: Multiple-choice Question Answering and Generation for Assessing Information Consistency in Summarization. arXiv. arXiv:2301.12307 [cs.CL] (2023). https://doi.org/10.48550/arXiv.2301.12307 . http://arxiv.org/abs/2301.12307 Accessed 2026-05-12

[42] Panickssery, A., Bowman, S.R., Feng, S.: LLM Evaluators Recognize and Favor Their Own Generations. In: Advances in Neural Information Processing Systems, vol. 37, pp. 68772–68802. Curran Associates, Inc., ??? (2024). https://doi.org/10.52202/079017-2197 . https://proceedings.neurips.cc/paper_files/paper/2024/hash/7f1f0218e45f5414c79c0679633e47bc-Abstract-Conference.html Accessed 2026-07-02

[43] Hüske-Kraus, D.: Suregen-2: a shell system for the generation of clinical documents. In: Proceedings of the Tenth Conference on European Chapter of the Association for Computational Linguistics - Volume 2. EACL '03, pp. 215–218. Association for Computational Linguistics, USA (2003). https://doi.org/10.3115/1067737.1067788 . https://dl.acm.org/doi/10.3115/1067737.1067788 Accessed 2025-11-18

[44] Huang, K., Altosaar, J., Ranganath, R.: ClinicalBERT: Modeling Clinical Notes and Predicting Hospital Readmission. arXiv. arXiv:1904.05342 [cs] (2020). https://doi.org/10.48550/arXiv.1904.05342 . http://arxiv.org/abs/1904.05342 Accessed 2025-10-13

[45] Ben Abacha, A., Yim, W.-w., Fan, Y., Lin, T.: An Empirical Study of Clinical Note Generation from Doctor-Patient Encounters. In: Vlachos, A., Augenstein, I. (eds.) Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pp. 2291–2302. Association for Computational Linguistics, Dubrovnik, Croatia (2023). https://doi.org/10.18653/v1/2023.eacl-main.168 . https://aclanthology.org/2023.eacl-main.168/ Accessed 2025-09-23

[46] Chiang, Y.-L., Yang, K.-F., Su, P.-C., Tsai, S.-F., Liang, K.-L.: Leveraging a Large Language Model for Streamlined Medical Record Generation: Implications for Health Care Informatics. Applied Clinical Informatics **16**(05), 1493–1506 (2025) https://doi.org/10.1055/a-2707-2959 . Accessed 2026-01-21

[47] Rosenbloom, S.T., Denny, J.C., Xu, H., Lorenzi, N., Stead, W.W., Johnson, K.B.: Data from clinical notes: a perspective on the tension between structure and flexible documentation. Journal of the American Medical Informatics Association **18**(2), 181–186 (2011) https://doi.org/10.1136/jamia.2010.007237 . Accessed 2026-05-11

[48] Muennighoff, N., Tazi, N., Magne, L., Reimers, N.: MTEB: Massive Text Embedding Benchmark. In: Vlachos, A., Augenstein, I. (eds.) Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pp. 2014–2037. Association for Computational Linguistics, Dubrovnik, Croatia (2023). https://doi.org/10.18653/v1/2023.eacl-main.148 . https://aclanthology.org/2023.eacl-main.148/ Accessed 2026-02-22

[49] McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv. arXiv:1802.03426 [stat] (2020). https://doi.org/10.48550/arXiv.1802.03426 . http://arxiv.org/abs/1802.03426 Accessed 2026-02-22

[50] Allaoui, M., Kherfi, M.L., Cheriet, A.: Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study. In: El Moataz, A., Mammass, D., Mansouri, A., Nouboud, F. (eds.) Image and Signal Processing, pp. 317–325. Springer, Cham (2020). https://doi.org/10.1007/

978-3-030-51935-3_34

[51] Asyaky, M.S., Mandala, R.: Improving the Performance of HDBSCAN on Short Text Clustering by Using Word Embedding and UMAP. In: 2021 8th International Conference on Advanced Informatics: Concepts, Theory and Applications (ICAICTA), pp. 1–6 (2021). https://doi.org/10.1109/ICAICTA53211.2021.9640285 . https://ieeexplore.ieee.org/document/9640285 Accessed 2026-02-22

[52] Oubenali, N., Messaoud, S., Filiot, A., Lamer, A., Andrey, P.: Visualization of medical concepts represented using word embeddings: a scoping review. BMC Medical Informatics and Decision Making **22**(1), 83 (2022) https://doi.org/10.1186/s12911-022-01822-9 . Accessed 2026-02-22

[53] Campello, R.J.G.B., Moulavi, D., Sander, J.: Density-Based Clustering Based on Hierarchical Density Estimates. In: Pei, J., Tseng, V.S., Cao, L., Motoda, H., Xu, G. (eds.) Advances in Knowledge Discovery and Data Mining, pp. 160–172. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37456-2_14

[54] Biswas, P.K., Iakubovich, A.: Extractive Summarization of Call Transcripts. IEEE Access **10**, 119826–119840 (2022) https://doi.org/10.1109/ACCESS.2022.3221404 . Accessed 2026-02-23

[55] Johnson, A.E.W., Pollard, T.J., Shen, L., Lehman, L.-w.H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., Mark, R.G.: MIMIC-III, a freely accessible critical care database. Scientific Data **3**(1), 160035 (2016) https://doi.org/10.1038/sdata.2016.35 . Accessed 2026-07-02

[56] Schumacher, E., Rosenthal, D., Naik, D., Nair, V., Price, L., Tso, G., Kannan, A.: MED-OMIT: Extrinsically-Focused Evaluation Metric for Omissions in Medical Summarization. In: Proceedings of the 4th Machine Learning for Health Symposium, pp. 897–922. PMLR, ??? (2025). https://proceedings.mlr.press/v259/schumacher25a.html Accessed 2025-11-19

[57] Tang, L., Sun, Z., Idnay, B., Nestor, J.G., Soroush, A., Elias, P.A., Xu, Z., Ding, Y., Durrett, G., Rousseau, J.F., Weng, C., Peng, Y.: Evaluating large language models on medical evidence summarization. npj Digital Medicine **6**(1), 158 (2023) https://doi.org/10.1038/s41746-023-00896-7 . Accessed 2026-07-02

[58] Zheng, L., Yin, L., Xie, Z., Sun, C., Huang, J., Yu, C.H., Cao, S., Kozyrakis, C., Stoica, I., Gonzalez, J.E., Barrett, C., Sheng, Y.: SGLang: efficient execution of structured language model programs. In: Proceedings of the 38th International Conference on Neural Information Processing Systems. NIPS '24, vol. 37, pp. 62557–62583. Curran Associates Inc., Red Hook, NY, USA (2024)

[59] Feblowitz, J.C., Wright, A., Singh, H., Samal, L., Sittig, D.F.: Summarization of clinical information: A conceptual model. Journal of Biomedical Informatics

44(4), 688–699 (2011) https://doi.org/10.1016/j.jbi.2011.03.008 . Accessed 2026-06-16

[60] Genes, N., Sills, J., Heaton, H.A., Shy, B.D., Scofi, J.: Addressing Note Bloat: Solutions for Effective Clinical Documentation. Journal of the American College of Emergency Physicians Open **6**(1), 100031 (2025) https://doi.org/10.1016/j.acepjo.2024.100031 . Accessed 2026-06-16

[61] Zhang, T., Kishore, V., Wu, F., Weinberger, K.Q., Artzi, Y.: BERTScore: Evaluating Text Generation with BERT. arXiv. arXiv:1904.09675 [cs] (2020). https://doi.org/10.48550/arXiv.1904.09675 . http://arxiv.org/abs/1904.09675 Accessed 2025-10-02

[62] Williams, C.Y.K., Bains, J., Tang, T., Patel, K., Lucas, A.N., Chen, F., Miao, B.Y., Butte, A.J., Kornblith, A.E.: Evaluating large language models for drafting emergency department encounter summaries. PLOS Digital Health **4**(6), 0000899 (2025) https://doi.org/10.1371/journal.pdig.0000899 . Accessed 2025-11-19

[63] Shang, Y., Tian, Y., Lyu, K., Zhou, T., Zhang, P., Chen, J., Li, J.: Electronic Health Record–Oriented Knowledge Graph System for Collaborative Clinical Decision Support Using Multicenter Fragmented Medical Data: Design and Application Study. Journal of Medical Internet Research **26**(1), 54263 (2024) https://doi.org/10.2196/54263 . Accessed 2026-07-02

[64] Khashabi, D., Stanovsky, G., Bragg, J., Lourie, N., Kasai, J., Choi, Y., Smith, N.A., Weld, D.: GENIE: Toward Reproducible and Standardized Human Evaluation for Text Generation. In: Goldberg, Y., Kozareva, Z., Zhang, Y. (eds.) Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 11444–11458. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (2022). https://doi.org/10.18653/v1/2022.emnlp-main.787 . https://aclanthology.org/2022.emnlp-main.787/ Accessed 2026-02-17