

Article

EHRForge: A Reproducible Workflow for Constructing an Evidence-Grounded Synthetic Temporal QA Benchmark from Longitudinal Clinical Notes

Quang An Quoc Tran ¹, Huy Quoc To ¹, Yongpan Sheng^{2,3} and Ming Liu ^{1,*}

¹ School of IT, Deakin University, Melbourne 3125, Australia; s224696086@deakin.edu.au (Q.A.Q.T.); q.to@deakin.edu.au (H.Q.T.)

² College of Computer and Information Science, Southwest University, Chongqing 400715, China; shengyp2011@163.com

³ Guizhou Provincial Laboratory of Big Data, College of Computer Science and Information, Guizhou University, Guiyang 550025, China

* Correspondence: m.liu@deakin.edu.au

Abstract

Longitudinal electronic health records (EHRs) are a natural source of temporal QA problems, but turning them into a reliable benchmark is not simply a matter of prompting a language model to write questions and answers. We present EHRForge, a reproducible workflow for constructing evidence-grounded synthetic temporal QA benchmarks from governed clinical-note collections. The workflow builds chronological visit-level timelines, selects context according to history length, generates schema-constrained QA candidates, and then subjects those candidates to deterministic temporal and structural checks followed by segment-to-visit evidence alignment against the source notes. In a restricted institutional case study, EHRForge generated 27,558 candidates. Of these, 18,867 (68.5%) passed the first verification stage and 10,742 (39.0% of the original candidates) passed the automated evidence-alignment stage across 829 patients. Human validation included dual review of 300 Round 1 rejections, 300 Round 2 rejections, and 1000 retained items. On the retained set, raw agreement between the two human validators ranged from 96.4% to 99.3% across five quality criteria, with Cohen's κ ranging from 0.459 to 0.737. In downstream QA experiments, we evaluated four retrieval methods in a retrieval-augmented QA setting, with NV-Embed-v2 achieving 77.0% Hit@5, 59.0% Recall@5, and 41.7% complete Coverage@5 when retrieving from full patient timelines. Qwen2.5-32B reached 40.1 Token F1 with NV-Embed-v2 evidence and 46.7 with oracle evidence. Further analysis showed that much of this oracle gap arose when one or more annotated evidence visits failed to reach the reader. These results position EHRForge as an auditable and configurable approach to benchmark construction in settings where source provenance and institutional data governance must be preserved.



Academic Editor: Paolo Bellavista

Received: 21 August 2026

Revised: 19 October 2026

Accepted: 21 October 2026

Published: 23 October 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: electronic health records; synthetic QA benchmark; benchmark construction; temporal reasoning; longitudinal clinical notes; evidence grounding; privacy-conscious clinical NLP

1. Introduction

Electronic health records (EHRs) contain temporally evolving clinical narratives distributed across admissions, outpatient encounters, investigations, treatment episodes, and follow-up care. Although public resources such as MIMIC-III [1], MIMIC-IV [2], and

PhysioNet [3] have substantially advanced clinical natural language processing (NLP), question answering (QA) over longitudinal clinical notes remains challenging. The evidence needed to answer a question can be distributed in many encounters, expressed differently by multiple clinicians, and embedded within patient histories that contain hundreds of visits [4].

Longitudinal clinical QA involves more than processing a large volume of text. A system must locate relevant encounters, distinguish repeated mentions from changes in clinical state, and preserve the chronological relationships among events. For example, a medication can be initiated, continued, withheld, reintroduced, and discontinued at different points in a patient record. Similarly, a symptom may be newly reported, persistent, improving, resolved, or recurrent. Questions about the first occurrence of an event, treatment response, disease progression, recurrence, or trends, therefore, require temporal comparison across visits rather than extraction of an isolated span from one note.

Existing clinical QA resources address several related settings. emrQA [5] uses templates and logical forms to generate large-scale QA instances on clinical notes. MIMICSQL and EHRSQL formulate questions as text-to-SQL tasks on structured EHR databases [6,7], while other datasets focus on graph-based reasoning, discharge summaries, radiology reports, and medication-related questions [8–12]. More recent benchmarks move toward multi-document and temporal evaluation. EHRNoteQA includes questions on discharge summaries accumulated across multiple admissions [13], while TIMER-Bench explicitly evaluates temporal reasoning on multi-visit EHRs using timestamped evidence [14].

What remains less well studied is the construction process itself when synthetic QA is generated from long, unstructured clinical timelines. Fluency is not enough: a plausible question can still be unanswerable from the record, a reasonable-sounding answer can lack support in the cited notes, and a genuine snippet can be attached to the wrong visit. These problems are easy to miss in longitudinal records, where diagnoses and medications may be repeated or copied across many encounters. We therefore treat language-model outputs as candidate benchmark items that must be verified before they are retained.

Figure 1 illustrates a simplified first-occurrence question. Although the answer is concise, the system must distinguish the initial medication documentation from subsequent mentions of continuation and discontinuation.

Patient Timeline (Total 3 visits)

Visit 1, 2019-03-14

"Metformin 500mg daily started for newly diagnosed type 2 diabetes."

Visit 2, 2019-08-02

"Keep using Metformin."

Visit 3, 2020-01-18

"Stop using Metformin due to improvement."

Question:

When was Metformin first documented for the patient?

Gold Evidence:

Visit 1 – "Metformin 500mg daily started for newly diagnosed type 2 diabetes."

Answer:

Metformin was first documented during Visit 1 on 2019-03-14.

Reasoning Type: First/Last Occurrence

Figure 1. Example of temporally grounded clinical question answering over longitudinal EHR narratives.

The example also highlights two separable challenges. The first is *evidence localization*: identifying the visits that contain the information required by the question. The second is *temporal answer synthesis*: integrating the retrieved evidence into an answer that respects

chronology. Increasing a model’s context window may allow more notes to be processed, but it does not guarantee reliable evidence selection, correct event ordering, or faithful grounding.

Table 1 positions EHRForge relative to representative clinical QA resources. The qualitative labels describe the principal design of each resource rather than their overall quality.

Table 1. Comparison of representative clinical QA resources and the benchmark constructed by EHRForge.

Resource	Longitudinal Context	Temporal Reasoning	Multi-Visit Reasoning	Evidence Grounding	Notes	Primary Setting
emrQA [5]	Partial	Limited	Partial	Answer evidence	✓	General EHR
MIMICSQL [6]	✗	Structured	✗	–	✗	ICU database
EHRSQL [7]	✗	Structured	✗	–	✗	EHR database
RadQA [10]	✗	Minimal	✗	Span-level	✓	Radiology
RxWhyQA [12]	Partial	Medication	Limited	Context-level	✓	Medication QA
EHRNoteQA [13]	✓	Implicit	✓	Context-level	✓	Multi-admission summaries
TIMER-Bench [14]	✓	Strong	✓	Timestamp-level	✓	Multi-visit EHR
EHRForge benchmark	✓	Strong	✓	Exact snippet-to-visit	✓	Longitudinal notes

Note: A check mark ✓ indicates that a feature is present, a cross ✗ indicates that it is absent, and a dash – indicates that the criterion is not applicable.

To address this problem, we developed EHRForge, an end-to-end workflow for constructing evidence-grounded synthetic temporal QA benchmarks from longitudinal clinical notes. Here, *synthetic* refers to the generated questions, answers, labels, and evidence references. The timelines and supporting notes remain patient-derived and subject to institutional governance. The workflow starts from raw note rows, builds chronological visit-level timelines, selects context according to history length, generates schema-constrained candidates, and then verifies both temporal structure and evidence provenance.

A central design choice is that generation is only one stage of the process. EHRForge records how each item was sampled, generated, filtered, and linked back to its source visits, together with timestamps, temporal labels, and provenance metadata. This makes the construction process auditable and repeatable without assuming that patient-grounded benchmark instances themselves can be released publicly.

The main contributions of this work are as follows:

- We introduce EHRForge, a modular workflow for constructing an evidence-grounded synthetic temporal QA benchmark from longitudinal unstructured clinical notes while preserving visit order, sampling decisions, temporal labels, and source provenance.
- We define reproducible interfaces and quality gates for timeline construction, history-length-aware context selection, schema-constrained generation, deterministic temporal and structural filtering, and exact evidence alignment.
- We instantiate EHRForge in a governed institutional case study and construct a restricted benchmark containing 10,742 QA pairs across 829 patients that passed the automated segment-to-visit evidence-alignment stage.

- We conduct human validation of rejected candidates from both automatic stages and of 1000 retained items, with a supplementary LLM judge applied to the retained set. We then assess downstream utility through retrieval and reader experiments that separate evidence-localization errors from answer-synthesis errors.

2. Related Work

2.1. Clinical QA Benchmarks

Clinical QA resources differ substantially in their source representations and reasoning requirements. emrQA [5] uses expert-derived templates and logical forms to generate QA instances from clinical notes. It provides substantial scale and includes temporal and arithmetic questions, but its construction is primarily template-driven and does not principally target unrestricted longitudinal histories.

MIMICSQL [6] and EHRSQL [7] formulate clinical QA as semantic parsing over structured EHR databases. Their executable queries provide explicit answer semantics and support operations such as filtering, aggregation, and temporal constraints. However, structured-query benchmarks abstract away challenges associated with narrative documentation, including lexical variation, copied-forward information, implicit temporal references, and evidence attribution to specific passages.

Other datasets focus on particular clinical documents or information needs. RadQA evaluates extractive QA over radiology reports [10], while DrugEHRQA and RxWhyQA address medication-related reasoning [11,12]. DiSCQ focuses on the questions associated with discharge summaries [9]. These datasets provide valuable task-specific evaluation, but most questions are grounded in one report, one summary, or a restricted clinical context.

EHRNoteQA moves toward multi-document patient-specific QA by constructing questions over discharge summaries from multiple admissions [13]. Its questions were generated using a language model and subsequently reviewed by clinicians, providing a clinically relevant setting for reasoning across accumulated records. Nevertheless, its primary evidence unit is the discharge summary, and it does not principally focus on explicit temporal-reasoning labels or exact matching of every evidence snippet to an original visit.

TIMER-Bench directly targets temporal reasoning over multi-visit EHR data [14]. It associates generated instruction–response pairs with timestamped evidence and studies temporal evidence distributions across patient timelines. TIMER-Bench is therefore the closest task-level comparator to EHRForge. The primary distinction is methodological emphasis: TIMER-Bench focuses on temporal benchmarking and instruction tuning, whereas EHRForge focuses on the reproducible construction and verification process, from raw note rows and visit aggregation to schema-constrained generation and exact evidence alignment.

2.2. Clinical Temporal Reasoning

Clinical temporal information extraction has established that temporal interpretation requires more than recognizing dates. The THYME framework represents clinical events, temporal expressions, document times, and temporal relations [15]. Clinical TempEval subsequently provided shared tasks for extracting events and time expressions and determining their temporal relationships [16].

Temporal extraction and longitudinal QA are related but distinct. Extraction tasks generally predict event spans, attributes, or pairwise relations. QA additionally requires the system to determine which temporal facts are relevant to a question and synthesize them into an answer. Identifying several medication mentions, for example, does not directly establish which mention represents initiation, whether treatment was later interrupted, or whether an outcome occurred after the intervention.

Longitudinal records also require tracking changes in clinical state. A diagnosis may be copied across several notes without representing a new event, and a medication mention may refer to historical rather than current use. Questions concerning progression, recurrence, trends, and treatment response must therefore distinguish repeated documentation from meaningful transitions between visits.

EHRForge operationalizes these requirements using eight task-level categories: before/after, progression, first/last occurrence, recurrence count, trend, comparison across visits, treatment response, and other. These categories do not replace fine-grained temporal annotations. Instead, they characterize the principal longitudinal operation required to answer each question and support stratified benchmark analysis.

2.3. Synthetic QA Generation and Evidence Verification

Automatic question generation reduces the cost of constructing large QA resources. Neural question-generation research showed that questions can be generated directly from source passages [17], while emrQA demonstrated the scalability of automated and semi-automated construction in the clinical domain [5]. Modern language models further increase linguistic diversity and make it possible to generate questions for more complex reasoning categories.

However, fluent generation does not guarantee benchmark validity. A question may be ambiguous or unanswerable, and an answer may be plausible without being supported by the source record. In longitudinal settings, generated evidence may also be assigned to the wrong encounter, event order may be reversed, or one of several required visits may be omitted. These risks motivate treating generated outputs as candidates that require independent validation [18].

Schema-constrained generation can enforce required fields and categorical values, but it cannot establish that an item is temporally coherent or evidence-grounded. EHRForge therefore separates generation from two verification stages. The first applies deterministic structural and temporal rules, including field validation, visit-index checks, chronological consistency, deduplication, and multi-visit requirements for relevant categories. The second checks whether every cited evidence snippet occurs in its referenced source visit.

This design is related to supporting-fact evaluation in multi-hop QA [19], but evidence provenance and semantic correctness remain distinct. Exact matching confirms that a snippet is present at the stated location. It does not prove that the evidence completely entails the answer. EHRForge consequently combines exact provenance verification with human validation of question validity, answer correctness, evidence grounding, and temporal consistency, while using an LLM judge as a supplementary automated check.

Benchmark construction also requires documentation of the generation and filtering process. Dataset-documentation frameworks emphasize reporting motivation, composition, intended uses, and limitations [20]. For synthetic clinical QA, relevant artifacts additionally include model versions, prompts, schemas, context-selection strategies, verification rules, and stage-wise retention counts. EHRForge records these artifacts so that the workflow can be audited and reproduced on appropriately governed local data even when the patient-grounded benchmark cannot be publicly released.

2.4. Retrieval and Long-Context Clinical Reasoning

Longitudinal QA combines retrieval and reading because only a small subset of a patient's visits may be relevant to a question. Retrieval-augmented generation conditions answer generation on selected source passages rather than parametric knowledge alone [21]. In clinical QA, each visit-level note can be treated as a candidate retrieval document within the corresponding patient's timeline.

Recent clinical approaches such as DR.EHR [22], EHR-RAG [23], KG-Retriever [24], and GRail [25] explore mechanisms for selecting or organizing relevant information from long EHR histories. Related frameworks such as TRACE [26] and FHIRPath-QA [27] investigate temporal memory or structured reasoning over patient timelines. These studies show that retrieval quality can directly determine whether a downstream reader receives enough evidence to answer a question.

Long-context architectures provide a complementary strategy. Longformer [28], Big-Bird [29], and Clinical-Longformer [30] expand the amount of text that can be processed, while BioBERT [31], ClinicalBERT [32], and GatorTron [33] provide domain-specific clinical representations. Nevertheless, larger context capacity does not guarantee faithful evidence use. Relevant visits may be overwhelmed by repetitive documentation, and models may favor recent or lexically similar information rather than the temporally correct evidence.

Evaluation should therefore distinguish evidence localization from answer synthesis. Partial retrieval recall measures whether some supporting visits are recovered, whereas complete evidence coverage determines whether all required visits are available. Reader evaluation under oracle evidence can then estimate performance when retrieval is bypassed. Because EHRForge links every verified item to its original visits, it supports both forms of analysis and helps identify whether errors arise from retrieval, temporal synthesis, or both.

3. Method

This section describes the complete EHRForge methodology, from governed corpus preparation and chronological timeline construction to history-length-aware context selection, schema-constrained QA generation, and two-stage verification. The design separates data transformation, candidate generation, and quality control into auditable components so that each retained benchmark item can be traced to its source timeline and reproduced under an institution's local governance constraints.

3.1. *Input Corpus and Timeline Construction*

3.1.1. Raw Clinical Notes

The source corpus consists of 191,749 clinical-note rows collected from a local Australian hospital and associated with 961 unique patients. Each row contains six primary fields: `Visit_DateTime`, `URNNumber`, `doc_id`, `doc_text`, `person_id`, and an index column. The notes span multiple years and represent longitudinal documentation across heterogeneous care trajectories. Unlike public critical-care resources such as MIMIC-III and MIMIC-IV [1,2], this corpus was collected in a local hospital setting and is used under institutional governance. EHRForge uses this corpus as the grounding source for synthetic benchmark generation: the QA pairs and labels are generated, whereas the clinical timelines and supporting evidence remain patient-derived and restricted.

Only one `doc_text` value is missing, and no exact duplicate rows are identified during preprocessing. Table 2 summarizes the corpus after preprocessing and visit-level aggregation. Figure 2 complements these aggregate counts by showing the yearly coverage of note timestamps, the right-skewed distribution of note lengths, and the long-tailed distribution of visits per patient. Together, these distributions show that the source collection spans heterogeneous documentation volumes and timeline lengths, motivating both visit-level aggregation and history-length-aware context selection.

3.1.2. Text Normalization and Visit Aggregation

We convert `Visit_DateTime` from millisecond Unix timestamps into standard date-time representations and apply light normalization to the clinical text. The preprocessing pipeline removes repeated whitespace and newline artifacts while preserving punctuation-

sensitive clinical expressions. We retain the original surface form as much as possible because punctuation, abbreviations, and formatting patterns may carry clinically relevant temporal information. We discard empty cleaned notes before timeline construction.

Table 2. Corpus statistics after preprocessing and visit-level aggregation.

Statistic	Value
Overall corpus	
Raw note rows	191,749
Unique patients	961
Visit-level rows after aggregation	170,827
Eligible cohort	
Eligible patients with 2–1000 visits	902
Patients with 2–10 visits	107
Patients with 11–100 visits	466
Patients with 101–1000 visits	329
Excluded patients	
Single-visit patients	30
Patients with more than 1000 visits	29
Maximum visits for one patient	3516

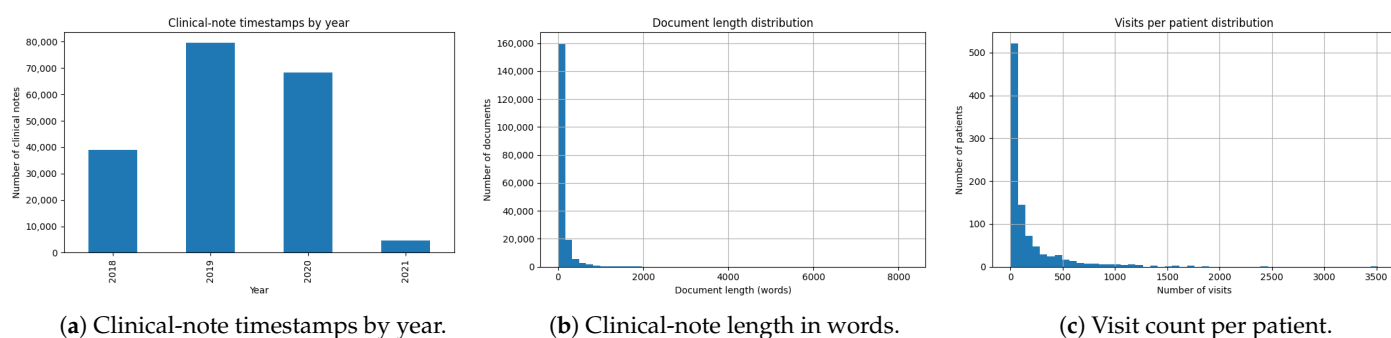


Figure 2. Corpus-level distributions after preprocessing and visit-level aggregation: (a) yearly temporal coverage, (b) note-length variability, and (c) the long-tailed distribution of patient timeline lengths.

Clinical documentation frequently contains multiple note fragments recorded at the same timestamp for the same patient. To reduce fragmentation and preserve coherent temporal events, we group rows by `person_id` and `Visit_DateTime`. Notes within the same group are concatenated into a visit-level context while retaining document identifiers as provenance metadata.

After aggregation, patient visits are sorted chronologically and serialized into structured patient timelines:

```
{person_id: int, visits: [{visit_datetime: str, text: str}, ...]}
```

Each line in `patient_sequences.jsonl` therefore represents one temporally ordered patient history.

3.1.3. Patient Eligibility and Visit Groups

The number of visits per patient varies dramatically. Short histories can often be processed in full, while long histories require sampling and chunking strategies to preserve temporal coverage within practical context limits. We retain patients with 2–1000 visits. Single-visit patients are excluded because they cannot support cross-visit temporal reasoning, and patients with more than 1000 visits are excluded from this version because they require specialized extreme-long-context summarization. Retained patients are assigned to three groups: 2–10, 11–100, and 101–1000 visits.

3.2. EHRForge Benchmark-Construction Workflow

Figure 3 presents EHRForge as a sequence of controlled transformations for synthetic benchmark construction. The workflow converts raw clinical-note rows into visit-level timelines, selects context using visit-aware sampling, generates schema-constrained temporal QA benchmark candidates, and applies two verification stages to enforce temporal validity and evidence fidelity before any item enters the final benchmark.

EHRForge Workflow for Synthetic Temporal QA Benchmark Construction

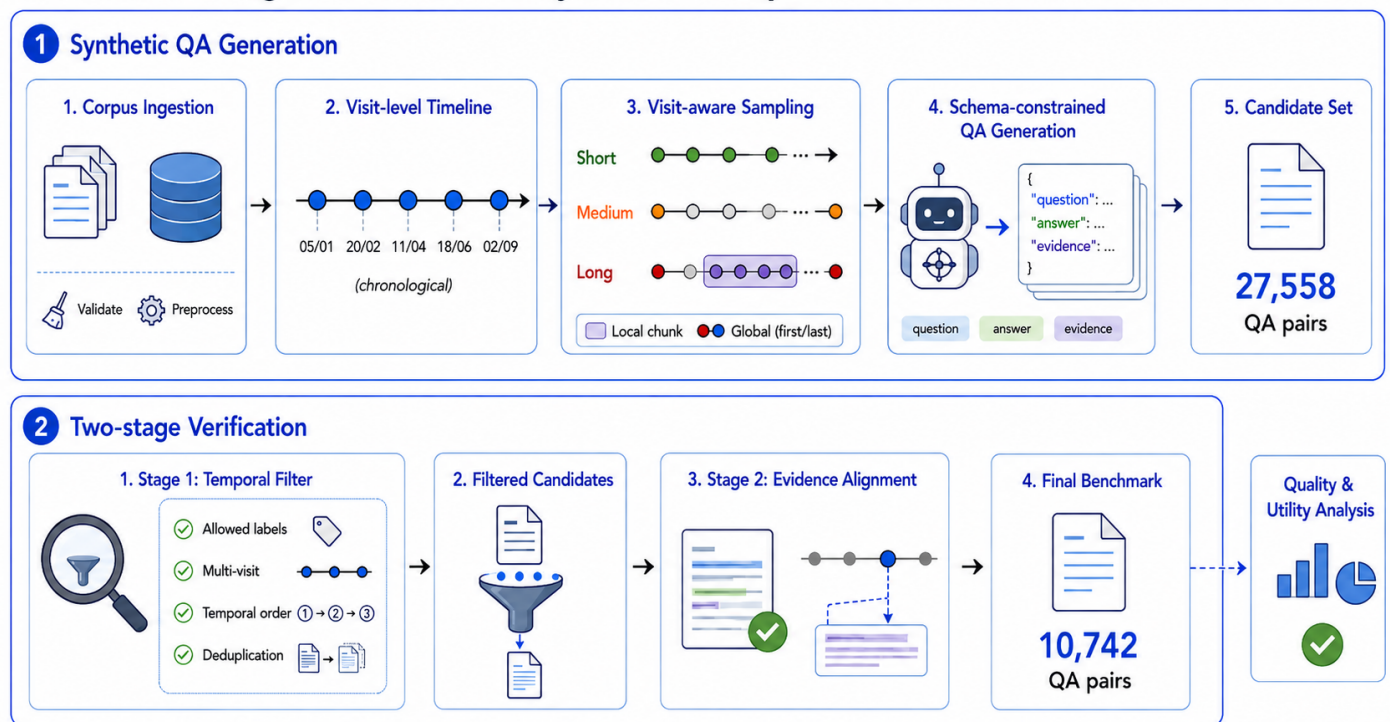


Figure 3. Overview of EHRForge. The workflow constructs an evidence-grounded synthetic temporal QA benchmark through preprocessing and visit-level timeline construction, length-aware context selection, schema-constrained candidate generation, temporal and structural filtering, exact evidence alignment, and benchmark quality and utility analysis.

3.2.1. Workflow Stages and Reproducible Interfaces

EHRForge is organized as a sequence of auditable stages with explicit inputs, outputs, and quality gates. This modular design allows an institution to replace a generation model, modify a sampling rule, extend the temporal taxonomy, or tighten a privacy constraint without redesigning the complete process. Table 3 summarizes the reusable interfaces and the condition that must be satisfied before an artifact advances to the next stage.

The principal EHRForge configuration artifacts are the source-field mapping, patient-eligibility rules, timeline-sampling parameters, generation prompt, output schema, temporal taxonomy, verification rules, and validation and evaluation protocols. Recording these artifacts together with software versions, model identifiers, randomization settings, batch checkpoints, and stage-level counts supports reproducibility and failure analysis without requiring disclosure of clinical text or patient-derived QA instances.

3.2.2. Visit-Aware Timeline Sampling

The benchmark-generation run was executed in ten batches, each containing up to 100 preprocessed patients while preserving coverage across the three visit groups. This batching strategy simplified the management and monitoring of QA pair generation, en-

abled progress to be tracked at intermediate stages, and made generation errors easier to isolate and recover from. The workflow uses a checkpoint file, `processed_patient_ids.txt`, to avoid reprocessing patients across batches and to support resumable execution after interruptions or failures.

Table 3. Reusable stages and quality gates in EHRForge.

Stage	Input	Operation	Output/Quality Gate
Corpus ingestion	Local note rows and field mapping	Validate required fields, timestamps, missing text, and duplicate records	Validated rows and corpus audit summary
Timeline construction	Validated rows	Normalize text, aggregate same-time fragments, and sort visits chronologically	Visit-level patient timelines with provenance
Context selection	Timeline and visit-count group	Apply full, U-shaped, global, or overlapping local sampling	Sampled timeline plus recorded sampling strategy
Candidate generation	Sampled timeline, prompt, and JSON schema	Generate temporal QA candidates with evidence references	Schema-valid candidate records
Temporal filtering	Candidate records and timeline metadata	Enforce allowed labels, multi-visit requirements, ordering, completeness, and deduplication	Temporally and structurally valid candidates
Evidence alignment	Filtered candidates and source notes	Match every cited snippet to the referenced original visit	Final evidence-grounded synthetic benchmark
Benchmark validation and utility assessment	Verified benchmark items	Conduct human validation and downstream retrieval/reader experiments	Benchmark quality estimates, failure analysis, and aggregate utility results

The sampling strategy varies according to timeline length to balance temporal coverage and prompt tractability. For patients with 2–10 visits, the full timeline is included. For medium-length timelines (11–100 visits), all visits are retained when the patient has at most 30 visits. Otherwise, a U-shaped sampling strategy selects the first 12 and last 18 visits to preserve both early-history and recent clinical evidence, where clinically important information is often concentrated [23].

For long timelines containing 101–1000 visits, the pipeline combines a global view with local chunk-based views. The global view captures coarse-grained longitudinal context by selecting the first 15 and last 15 visits, while the local views use overlapping chunks of 30 visits with overlap 5 to preserve fine-grained temporal continuity between adjacent clinical events. This hierarchical combination of global context and local chunking is motivated by prior work on long-document retrieval and hierarchical reasoning, suggesting that combining global temporal structure with fine-grained local context can improve reasoning over long patient timelines [34,35].

This design aims to preserve clinically informative temporal coverage while keeping generation prompts tractable. The global view supports questions about the beginning and end of a trajectory. Local chunks support denser temporal reasoning over adjacent clinical events.

3.2.3. Schema-Constrained Synthetic QA Generation

Synthetic QA benchmark candidates are generated using GPT-5.4-nano [36] through the OpenAI Responses API with strict JSON-schema output constraints. This model was se-

lected because the task primarily requires high-throughput structured generation, temporal evidence extraction, and schema-constrained annotation rather than open-ended clinical decision making. In this setting, smaller instruction-tuned language models can produce useful synthetic QA candidates when generation is tightly constrained through structured prompting, schema enforcement, evidence grounding, and downstream verification [14,18]. EHRForge therefore uses the generation model to produce provisional QA candidates rather than treating model outputs as verified benchmark items. Each generated candidate must subsequently pass the temporal, structural, and evidence-alignment checks described in the verification procedure below.

Generation Prompt and Label Taxonomy

The system prompt instructs the model to construct clinically meaningful, non-redundant QA pairs using only the supplied longitudinal timeline. It explicitly prioritizes temporal relationships across visits and timestamps, prohibits unsupported clinical inference and hallucinated conclusions, requires the omission of personally identifying information, and enforces strict JSON output. The prompt operationalizes eight temporal-reasoning categories: *before_after*, *progression*, *first_last_occurrence*, *recurrence_count*, *trend*, *comparison_across_visits*, *treatment_response*, and *other*. Difficulty reflects how much evidence must be integrated to answer the question. Easy questions use direct time-aware evidence, medium questions connect temporally distinct evidence, and hard questions require reasoning across multiple visits.

The model is instructed to:

- Generate high-quality, non-redundant QA pairs grounded only in the provided timeline.
- Prioritize temporal reasoning across visits and timestamps.
- Avoid unsupported inference or hallucinated clinical conclusions.
- Avoid personally identifying information.
- Return outputs strictly in JSON format following the predefined schema.

The core fields of the generated QA benchmark, including timeline metadata, question-answer pairs, temporal reasoning labels, and supporting evidence, are summarized in Table 4.

Table 4. Core fields of the synthetic temporal QA benchmark produced by EHRForge.

Field	Description
person_id	Internal patient identifier used for timeline construction and evidence tracing. The identifier is removed or anonymized before external release.
num_chunks	Number of timeline chunks included in the input provided to the generation model.
visit_group	Visit-frequency category assigned according to the total number of visits in the patient timeline: 2–10, 11–100, or 101–1000 visits.
timeline_sampling_strategy	Timeline-selection strategy used during QA generation, including full-timeline inclusion, U-shaped sampling, global sampling, or overlapping local chunking.
question	Natural-language clinical question generated from the longitudinal patient timeline.
answer	Evidence-grounded answer supported by cited clinical-note snippets.
reasoning_type	Temporal reasoning category, including before/after, progression, first/last occurrence, recurrence count, trend analysis, comparison across visits, treatment response, and other.
difficulty	Difficulty level assigned to the QA pair: easy, medium, or hard.
temporal_signal	Short free-text description of the temporal relationship underlying the question.
evidence	Supporting evidence snippets linked to sampled visit indices, original visit indices, timestamps, and source-text spans.

Strict JSON Output Schema

The generation call uses the following schema to require at least one QA item, restrict categorical labels, require one or more evidence references, and disallow undeclared properties.

```
{
  "type": "object",
  "properties": {
    "qas": {
      "type": "array",
      "minItems": 1,
      "items": {
        "type": "object",
        "properties": {
          "question": {"type": "string"},
          "answer": {"type": "string"},
          "reasoning_type": {
            "type": "string",
            "enum": [
              "before_after",
              "progression",
              "first_last_occurrence",
              "recurrence_count",
              "trend",
              "comparison_across_visits",
              "treatment_response",
              "other"
            ]
          },
        },
      },
    },
    "difficulty": {
      "type": "string",
      "enum": ["easy", "medium", "hard"]
    },
    "evidence": {
      "type": "array",
      "minItems": 1,
      "items": {
        "type": "object",
        "properties": {
          "visit_index": {"type": "integer"},
          "evidence_snippet": {"type": "string"}
        },
        "required": ["visit_index", "evidence_snippet"],
        "additionalProperties": false
      }
    },
    "is_temporal": {"type": "boolean"},
    "temporal_signal": {"type": "string"}
  },
  "required": [
    "question",
```

```

        "answer",
        "reasoning_type",
        "difficulty",
        "evidence",
        "is_temporal",
        "temporal_signal"
    ],
    "additionalProperties": false
  }
},
"required": ["qas"],
"additionalProperties": false
}

```

The schema guarantees structural validity, but does not establish temporal relevance, evidence sufficiency, clinical faithfulness, or privacy compliance. These semantic requirements are encoded in the prompt and then checked by the downstream filters. At generation time, evidence is represented by `visit_index` and `evidence_snippet`. During normalization, these references are assigned to original visit indices and timestamps for provenance validation.

3.3. Benchmark Verification

Following schema-constrained generation, all candidate QA pairs undergo a two-stage verification procedure designed to improve temporal validity and evidence fidelity. The first stage applies deterministic temporal and structural filtering to remove malformed QA pairs and enforce temporally consistent multi-visit reasoning, following prior work showing the importance of structured filtering and schema-guided QA validation in synthetic clinical QA generation [18]. The second stage matches the exact evidence at segment-level with the original clinical-note text to verify the provenance of the evidence at the referenced visit, motivated by evidence-based QA and supporting-fact verification frameworks that emphasize faithful evidence attribution for multi-hop reasoning tasks [19]. The reliability of these automatic decisions is later examined through independent human validation of the rejected candidates from both stages.

3.3.1. Benchmark Schema

Each verified JSONL record represents one sampled patient timeline together with its associated synthetic QA benchmark items. The schema stores timeline-level metadata describing the generation context and a list of candidate or retained QA pairs. Every benchmark item stores semantic and temporal fields, including reasoning type, difficulty level, evidence snippets, and temporal signals. Evidence is linked back to original visits using visit indices and timestamps so that each item can be audited against the source notes inside the governed environment.

3.3.2. Round 1: Automatic Temporal and Structural Filtering

The first verification stage normalizes the generated outputs and applies deterministic structural and temporal checks before the candidates proceed to evidence validation. During normalization, the pipeline standardizes the generated fields, removes invalid, empty, or duplicated evidence entries, and attaches the corresponding original visit indices and timestamps to each retained evidence item. Candidate QA pairs are discarded when the question or answer is empty, or no valid evidence remains after normalization, or when

the question is too short to constitute a meaningful QA item. Required fields and permitted reasoning types and difficulty labels are also enforced through the structured output schema. For reasoning categories configured as requiring longitudinal support, including before and after relationships, progression, recurrence count, trend, comparison across visits, and treatment response, the pipeline requires multiple evidence items. Temporal candidates containing explicit language about change or sequence may be subject to the same requirement. When valid timestamps are available, the pipeline also checks that the cited evidence appears in nondecreasing chronological order. In addition, a lexical heuristic flags answers that introduce potentially interpretive language not reflected in the cited evidence. These rules also aim to remove candidates whose evidence does not support a genuinely temporal or longitudinal relationship, for example when the question can be answered from a single visit.

After these checks are applied within each generated timeline or chunk, duplicate QA pairs are removed across outputs using normalized question text and a normalized, truncated representation of the answer. Across ten benchmark-generation batches, the pipeline produced 27,558 candidate QA pairs, of which 18,867 (68.46%) were retained and 8691 (31.54%) were rejected. Table 5 summarizes the observed rejection reasons. Insufficient evidence for temporal reasoning was the most common reason, affecting 5787 rejected candidates (66.59%), followed by evidence not being in chronological order (2680; 30.84%) and possible over-inference (303; 3.49%). The latter was identified using a lexical heuristic that flags potentially unsupported interpretations in generated answers.

Table 5. Observed rejection reasons during Round 1 temporal and structural filtering.

Rejection Reason	Count (<i>n</i>)	Rejected Items (%)
Insufficient evidence for temporal reasoning	5787	66.59
Evidence not in chronological order	2680	30.84
Possible over-inference	303	3.49

Note: Percentages are calculated relative to the 8691 unique rejected QA pairs. A candidate may trigger multiple rejection rules, so the counts and percentages sum to more than 8691 and 100%, respectively. Other filtering checks were also applied but resulted in no recorded rejections in this run.

Although Round 1 removes candidates that violate the implemented structural and temporal rules, it does not establish whether an event is truly the first or last occurrence across the complete patient history, resolve the semantic direction of temporal relations, or guarantee that the cited evidence fully supports the generated answer.

3.3.3. Round 2: Exact Evidence Matching

The second verification stage performs exact evidence matching against the original clinical-note text to confirm the provenance of every cited supporting snippet. For each evidence item, the pipeline retrieves the corresponding source visit using the internal patient identifier, original visit index, and visit timestamp recorded during normalization, and then checks whether the cited snippet occurs in the referenced visit. Verification is performed independently for every evidence item, and a QA pair is retained only when all cited snippets satisfy the implemented source-visit matching rule. This all-evidence requirement reduces fabricated or incorrectly attributed citations by requiring every accepted evidence segment to be recoverable from the referenced expert-authored clinical note.

The matching procedure also allows shortened evidence snippets that contain ellipses. In these cases, the snippet is divided into separate text segments and each segment must appear somewhere in the same referenced source visit. The implementation does not require separated ellipsis segments to be contiguous or to preserve their original within-visit order. Round 2 therefore functions as a segment-level evidence-provenance and traceability check

rather than a semantic-entailment classifier or a guarantee of faithful quote reconstruction. This distinction is important: the subsequent human audit identified a small number of retained items for which all matched fragments were present in the source visit but their composition was judged unfaithful, for example because fragments were reversed or short repeated terms were matched across different parts of a long note. Across the merged ten-batch run, 10,742 QA pairs passed the automated evidence-alignment rule, while 8125 were removed because at least one evidence item failed that rule. These automated decisions are subsequently examined through independent human validation.

4. Experiments

We evaluate EHRForge along three complementary dimensions. Aggregate benchmark statistics characterize the distributions of timeline lengths, difficulty levels, and temporal-reasoning categories. Human validation, together with a supplementary LLM judge, assesses the quality of the generated QA benchmark. Retrieval and reader experiments assess whether the benchmark presents meaningful challenges in longitudinal evidence localization and temporal answer synthesis.

4.1. Benchmark Composition and Validation Sample

The final benchmark contains 10,742 QA pairs from 829 patients that passed the automated evidence-alignment stage. Table 6 compares the full benchmark with the 1000-item quality-validation sample across patient timeline length, difficulty, and temporal reasoning type. Long histories contribute most of the retained items, with 86.6% coming from patients who have 101–1000 visits. This concentration partly reflects the use of multiple local and global generation views for longer timelines. Medium questions form the largest difficulty group, while comparison across visits, progression, and before/after are the most common reasoning types.

The 1000-item validation sample was drawn to preserve these proportions closely rather than to create a deliberately harder or easier subset. It therefore remains broadly representative of the retained benchmark while covering every visit group, difficulty level, and reasoning category.

Table 6. Composition of the full benchmark and the 1000-item quality-validation sample.

Category	Full Benchmark, <i>n</i> (%)	Validation Sample, <i>n</i> (%)
<i>Panel A: patient timeline length</i>		
2–10 visits	200 (1.9)	19 (1.9)
11–100 visits	1244 (11.6)	116 (11.6)
101–1000 visits	9298 (86.6)	865 (86.5)
<i>Panel B: difficulty</i>		
Easy	2105 (19.6)	196 (19.6)
Medium	6502 (60.5)	605 (60.5)
Hard	2135 (19.9)	199 (19.9)
<i>Panel C: temporal reasoning type</i>		
Comparison across visits	2736 (25.5)	256 (25.6)
Progression	1992 (18.5)	185 (18.5)
Before/after	1837 (17.1)	171 (17.1)
Treatment response	1299 (12.1)	121 (12.1)
Trend	1218 (11.3)	113 (11.3)
First/last occurrence	937 (8.7)	87 (8.7)
Recurrence count	623 (5.8)	58 (5.8)
Other	100 (0.9)	9 (0.9)

Some QA pairs rely on nearby visits within a short hospitalization, while others require evidence distributed across months or years. These distributions describe this particular governed execution of EHRForge. They are not intended as fixed properties of the workflow because a different institution, corpus, prompt, or sampling configuration may produce a different benchmark profile.

4.2. Benchmark Quality Validation

Human validation was extended beyond the retained benchmark so that we could also inspect decisions made by the two automatic verification stages. We conducted three consecutive validation exercises. Two samples contained 300 candidates rejected in Round 1 and 300 candidates rejected in Round 2 and the last sample contained 1000 retained QA items. The rejected-item validation checks whether independent human validators agree with the reasons for rejection at each stage. The retained-item validation assesses the quality of QA pairs that passed both automatic stages. Because the two validations use different criteria and serve different purposes, their results are not compared directly.

4.2.1. Stage-Wise Human Validation of Rejection Decisions

The human validation was designed to reflect the purpose of each verification stage. Round 1 focuses on temporal and structural filtering, while Round 2 adds evidence matching against the source notes. Round 2 therefore includes one additional criterion, *Exact Match*, together with the four criteria used in Round 1.

For Round 1 validation, two human validators independently assessed 300 rejected QA items using four binary criteria: *Insufficient Evidence*, *Not-In-Order Evidence*, *Possible Over Inference*, and *Not Temporal Evidence*. The validators agreed on 1104 of 1200 criterion-level decisions (92.0%) and agreed on all four labels for 209 of the 300 items (69.7%). Agreement was high for insufficient evidence, evidence ordering, and possible over-inference. Most disagreements involved the *Not Temporal Evidence* criterion, particularly whether a question truly required longitudinal evidence or could be answered from a single visit. This accounted for 75 of the 96 disagreements (78.1%). Although this sample represents only a subset of all Round 1 rejections, the results suggest that the first verification stage captures several clear and reproducible quality issues, while temporal relevance remains more interpretation-sensitive.

For Round 2 validation, the same process was applied to 300 rejected QA items using the four Round 1 criteria together with *Exact Match*. These candidates had already passed all automatic Round 1 checks and were rejected only during evidence matching. The validators agreed on 1355 of 1500 criterion-level decisions (90.3%) and agreed on all five labels for 175 of the 300 items (58.3%). Compared with the Round 1 sample, both validators assigned fewer True labels for *Insufficient Evidence*, *Not-In-Order Evidence*, and *Not Temporal Evidence*, while agreement remained high for the more objective criteria. This pattern is consistent with Round 1 removing many of the structural and temporal issues it was designed to detect. *Possible Over Inference* was the main exception and remained present in the Round 2 sample. Agreement for *Exact Match* was 91.7%, with Cohen's $\kappa = 0.809$ [37], showing that the additional evidence-matching decision was also comparatively reproducible. Although inter-rater agreement was high, the human validators judged some automatically rejected items as acceptable exact matches. This reflects differences between the strict automatic matching rule and human interpretation, for example when minor normalization or abbreviated terminology preserves the same source meaning. Because both validators assigned the same category to all items for evidence ordering, Cohen's κ could not be estimated for that criterion. Table 7 summarizes the judgment distributions and agreement results, where "n.e." denotes a value that could not be estimated.

Table 7. Dual-human validation of rejected candidates from Round 1 and Round 2.

Criterion	Human 1 T/F	Human 2 T/F	Agreement	Cohen's κ
<i>Round 1</i>				
Insufficient evidence	90/210	93/207	97.0%	0.929
Not-in-order evidence	8/292	8/292	100.0%	1.000
Possible over-inference	22/278	32/268	96.0%	0.757
Not temporal evidence	215/85	164/136	75.0%	0.479
<i>Round 2</i>				
Insufficient evidence	28/272	21/279	94.3%	0.623
Not-in-order evidence	0/300	0/300	100.0%	n.e. [†]
Possible over-inference	31/269	51/249	92.0%	0.664
Not temporal evidence	12/288	91/209	73.7%	0.175
Exact match	94/206	99/201	91.7%	0.809

T/F indicates the number of True and False judgments. [†] Cohen's κ could not be estimated because both validators assigned the same category to all Round 2 items.

Representative Round 1 and Round 2 cases are provided in Appendix A.1. They illustrate both agreement cases and cases where the validators disagreed about evidence sufficiency or the strictness of literal provenance matching.

4.2.2. Final-Round Validation

The final-round validation examined 1000 QA pairs that passed both automatic verification stages. For each item, the human validators were shown the generated question, gold answer, cited evidence, visit timestamp or visit index, relevant surrounding clinical-note context, temporal-reasoning category, and difficulty label. Two independent human validators assessed the same 1000 items using the same rubric. GPT-5.5 [38] was also applied to the same set as a supplementary automated judge.

Each QA pair was assessed using five binary criteria:

1. **Evidence match:** the cited evidence appears in the source note and corresponds faithfully to the referenced visit.
2. **Question validity:** the question is clinically meaningful, answerable, and unambiguous given the displayed context.
3. **Answer correctness:** the answer directly and correctly addresses the question.
4. **Evidence grounding:** the cited evidence and surrounding context sufficiently support the answer.
5. **Temporal consistency:** the answer respects chronological order, temporal expressions, and cross-visit context.

The validators based their judgments only on the information shown for each QA pair. A criterion was marked as false when the answer required unsupported inference, the evidence did not support the answer, the temporal order was inconsistent, or the question could not be answered from the displayed material. External medical knowledge was not used to infer information beyond the available evidence and context. All validation was conducted within the governed environment, and clinical material was not copied, downloaded, screenshot, or redistributed. Figure 4 shows the information presented during validation using fully synthetic text.

The two human validators gave similar judgments across the five criteria. Validator 1 marked 98.5% of items as evidence matched, 97.3% as valid questions, 95.2% as correct answers, 96.1% as adequately grounded, and 96.8% as temporally consistent. The corresponding rates for Validator 2 were 98.8%, 96.4%, 94.8%, 96.7%, and 97.3%. Raw agreement between the two validators ranged from 96.4% to 99.3%. Cohen's κ was highest for evi-

dence match and answer correctness, while evidence grounding and temporal consistency showed lower values despite high raw agreement. GPT-5.5 was more conservative than the human validators, particularly for answer correctness and evidence grounding. To estimate joint item-level validity, we additionally marked an item as passing only when the same evaluator assigned True to all five criteria. Human agreement and Cohen's κ for this composite label were calculated from the two validators' paired all-five-pass decisions. Wilson 95% confidence intervals [39] are included for the positive-label rates.

Question: When was the symptom first documented as improving?

Gold answer: The symptom was first documented as improving at Visit i on the displayed visit date.

Evidence snippet: "...symptom improved following treatment adjustment ..."

Visit information: Visit i , with the timestamp or relative visit index shown.

Surrounding note context: Relevant excerpt from the clinical-note timeline.

Judgments: Evidence match [True/False], Question validity [True/False], Answer correctness [True/False], Evidence grounding [True/False], and Temporal consistency [True/False].

Figure 4. Schematic representation of the human validation interface.

Panel B of Table 8 reports the stricter all-five-pass outcome. Validator 1 accepted 920 of the 1000 items (92.0%, 95% CI [90.2, 93.5]), while Validator 2 accepted 905 (90.5%, 95% CI [88.5, 92.2]). Both validators marked all five criteria True for 885 items (88.5%, 95% Wilson CI [86.4, 90.3]), with 94.5% agreement on the composite judgment and Cohen's $\kappa = 0.656$. GPT-5.5 independently assigned all five positive labels to 800 items (80.0%, 95% CI [77.4, 82.4]). Examples of full agreement and disagreement on this composite judgment are provided in Appendix A.1.3.

Table 8. Final-round validation of 1000 retained QA pairs.

Criterion	GPT-5.5 True, n (%) [95% CI]	Validator 1 True, n (%) [95% CI]	Validator 2 True, n (%) [95% CI]	Human Agreement	Cohen's κ
<i>Panel A. Per-criterion results</i>					
Evidence match	962 (96.2) [94.8, 97.2]	985 (98.5) [97.5, 99.1]	988 (98.8) [97.9, 99.3]	99.3%	0.737
Question validity	934 (93.4) [91.7, 94.8]	973 (97.3) [96.1, 98.1]	964 (96.4) [95.1, 97.4]	97.5%	0.591
Answer correctness	819 (81.9) [79.4, 84.2]	952 (95.2) [93.7, 96.4]	948 (94.8) [93.2, 96.0]	96.4%	0.621
Evidence grounding	817 (81.7) [79.2, 84.0]	961 (96.1) [94.7, 97.1]	967 (96.7) [95.4, 97.6]	96.4%	0.481
Temporal consistency	869 (86.9) [84.7, 88.9]	968 (96.8) [95.5, 97.7]	973 (97.3) [96.1, 98.1]	96.9%	0.459
<i>Panel B. All-criterion results</i>					
All five criteria jointly [†]	800 (80.0) [77.4, 82.4]	920 (92.0) [90.2, 93.5]	905 (90.5) [88.5, 92.2]	94.5%	0.656

[†] An item passes all five criteria jointly only when that evaluator marks Evidence match, Question validity, Answer correctness, Evidence grounding, and Temporal consistency all True for the same QA pair. Agreement and Cohen's κ in Panel B are computed on these composite binary labels.

A small number of evidence-match failures remained after Round 2. This is consistent with the segment-level matching rule used by the automatic verifier. The matcher confirms that the required text segments occur in the referenced visit, but it does not require ellipsis-separated fragments to preserve their original order or contiguity. Examples F-A and F-B in Appendix A.1.3 show two retained cases that passed Round 2 for this reason but were later marked as evidence-match failures during human validation. In one case, the relevant fragments were present in the same visit but appeared in reversed order in the generated evidence. In another, repeated short terms such as "CRP" allowed separate date, analyte, and value fragments to be matched within a long note even though their combination did not form a faithful local reconstruction.

The two human validators were closely aligned on most retained QA pairs, and both assigned high positive rates across all five criteria. The lower κ values for evidence ground-

ing and temporal consistency suggest that these judgments involve more interpretation than literal evidence matching. GPT-5.5 gave lower positive rates for several criteria and is therefore used as a supplementary check rather than as a replacement for human validation.

4.3. Downstream Utility Evaluation

For the downstream experiments, each QA item is linked to its corresponding visits in the patient timeline using `original_visit_index`. Timestamps are checked during loading, duplicate evidence visits are removed, and out-of-range indices are rejected. Predictions are matched to QA items using the composite key (`person_id`, `qa_index`) to ensure consistent alignment between benchmark items, retrieved evidence, and reader outputs.

The downstream evaluation considers retrieval and reading as two related but distinct components. The retriever identifies visits containing the supporting evidence, while the reader generates an answer from the context it receives. Retrieval is evaluated over the full patient timeline, with an additional sampled-window diagnostic, and end-to-end QA is measured using the top five retrieved visits. These results are compared with an Oracle condition in which the annotated gold visits are provided directly to the reader.

4.3.1. Experimental Setup

Retrieval Models, Search Spaces, and Metrics

We evaluate BM25 [40], MedCPT [41], NV-Embed-v2 [42], and a reciprocal-rank-fusion system [43] combining full-depth BM25 and MedCPT rankings with RRF constant 60 (Hybrid-RRF60). BM25 lowercases text and removes punctuation before indexing. For the sampled-window diagnostic, BM25 builds a separate index and inverse-document-frequency statistics only over visits in that window. NV-Embed-v2 uses its model-specific pooling implementation with `transformers` 4.42.4. All retrievers produce full rankings with scores.

The *full-timeline* search space contains every visit in the corresponding eligible patient timeline and is the primary setting. It contains 4,389,646 ranked rows per retriever, or approximately 409 candidate visits per question. The *sampled-window* setting evaluates retrieval only over visits supplied by the benchmark-generation sampling procedure and contains 304,952 ranked rows per retriever, or approximately 28 candidate visits per question.

For a question with G distinct gold visits and N candidate visits, we report Hit@ K , Recall@ K , Coverage@ K , and mean reciprocal rank (MRR). Hit@ K is one when at least one gold visit occurs in the top K . Recall@ K is the fraction of gold visits recovered. Coverage@ K is one only when all gold visits are recovered, and MRR is the reciprocal rank of the first gold visit. Exact random-ranking baselines are computed from N , G , and $K_{\text{eff}} = \min(K, N)$. Overall means use paired bootstrap 95% confidence intervals [44] with 1000 resamples, seed 42, and shared resample indices across retrieval systems.

Reader Models and Context Construction

We evaluate five readers under five evidence conditions, giving 25 runs and 268,550 predictions. The reader set includes RoBERTa [45], BioBERT [31], Longformer [28], and Qwen2.5 [46]. The extractive readers use checkpoints designed for SQuAD-style question answering. Before the main experiments, we run a sanity check on answerable questions from the SQuAD 2.0 development set [47]. For each retrieval condition, the reader receives the top five visits from the full-timeline ranking, while the Oracle condition provides the annotated gold visits directly. All readers use the same saved context for a given question and evidence condition, and the serialized context is verified with SHA256 before inference. Selected visits are ordered chronologically and prefixed with their visit index and timestamp. The input is limited to 3500 Qwen tokenizer tokens. When trimming is

required, the available budget is distributed across the selected visits while preserving every visit header.

The reader checkpoints, hardware configurations, and inference settings are summarized in Table 9.

Table 9. Reader models and inference settings used in the downstream QA experiments.

Reader	Checkpoint	Hardware	Inference Settings
RoBERTa-base (SQuAD2)	deepset/ roberta-base-squad2	1 × A100 80GB	512-token input; stride 128; maximum answer length 64; FP32
BioBERT-v1.1 (SQuAD2)	kttrapeznikov/ biobert_v1.1_pubmed_squad_v2	1 × A100 80GB	512-token input; stride 128; maximum answer length 64; FP32
Longformer-base (SQuAD2)	mrm8488/ longformer-base-4096- finetuned-squadv2	1 × A100 80GB	4096-token input; stride 128; maximum answer length 64; FP32
Qwen2.5-7B-Instruct	Qwen/Qwen2.5-7B-Instruct	1 × H100 80GB	vLLM 0.28.0; BF16; greedy decoding; seed 42; maximum 256 new tokens
Qwen2.5-32B-Instruct	Qwen/Qwen2.5-32B-Instruct	2 × H100 80GB	vLLM 0.28.0; TP = 2; BF16; greedy decoding; seed 42; maximum 256 new tokens

For the biomedical extractive baseline, we use the QA-fine-tuned kttrapeznikov/biobert_v1.1_pubmed_squad_v2 checkpoint. All extractive readers use sliding-window inference with stride 128, a maximum answer length of 64 tokens, and no null-answer option. The Qwen readers use vLLM 0.28.0 [48] in bfloat16 with greedy decoding, seed 42, an 8192-token maximum model length, and at most 256 newly generated tokens. The reference answer is never included in the reader prompt.

Answer Scoring and Statistical Analysis

Exact Match and Token F1 use SQuAD normalization and are reported on a 0–100 scale. Because this benchmark is formulated as a free-text question answering task, model responses can be semantically correct while differing substantially in wording from the reference answer. Exact Match is therefore close to zero for all readers and is reported only in Appendix A.3. The primary semantic metric is BERTScore-F1 [49] using FacebookAI/roberta-large at layer 17. A secondary clinical-domain BERTScore using emilyalsentzer/Bio_ClinicalBERT at layer 9 is reported in the appendix. Both use bert-score 0.3.13, FP32, batch size 64, idf=False, and rescale_with_baseline=False.

Prometheus 2 [50] (prometheus-eval/prometheus-8x7b-v2.0) provides a supplementary reference-answer equivalence score. The judge receives the question, candidate answer, and reference answer but not the clinical-note context. It therefore measures agreement with the reference answer rather than independent entailment from the source notes. It runs with vLLM 0.28.0 on two H100 80 GB GPUs, tensor parallelism 2, BF16, greedy decoding, and seed 42. Judge metrics use a fixed 1000-question subset stratified by reasoning type and difficulty. Scores of at least 4 on the 1–5 scale are counted as correct. Because all conditions contain the same question keys, shared bootstrap indices are used for paired comparisons.

4.3.2. Retrieval Performance

Table 10 summarizes retrieval performance over all 10,742 QA pairs. NV-Embed-v2 gives the strongest results in both search spaces. On the full patient timeline, it reaches

0.770 Hit@5, 0.590 Recall@5, 0.417 Coverage@5, and 0.603 MRR. The difference between Hit@5 and Coverage@5 is particularly informative. A relevant visit is retrieved within the top five for 77.0% of questions, but all required evidence visits are retrieved for only 41.7%. This suggests that the main retrieval challenge is not only finding relevant evidence, but finding the complete set of visits needed to answer a longitudinal question. Bootstrap 95% confidence intervals are shown for the retrieval models, while the random rows give the exact expectation under a uniform random ranking.

Table 10. Retrieval performance in the full-timeline and sampled-window search spaces.

Retriever	Hit@5	Recall@5	Coverage@5	Recall@20	MRR
<i>Panel A: full patient timeline</i>					
Random	0.077	0.048	0.023	0.142	0.061
BM25	0.634 [0.626, 0.644]	0.469 [0.462, 0.478]	0.318 [0.309, 0.327]	0.660 [0.653, 0.668]	0.489 [0.481, 0.497]
MedCPT	0.586 [0.577, 0.595]	0.429 [0.422, 0.437]	0.287 [0.279, 0.295]	0.672 [0.665, 0.680]	0.429 [0.422, 0.437]
NV-Embed-v2	0.770 [0.762, 0.778]	0.590 [0.582, 0.598]	0.417 [0.408, 0.427]	0.812 [0.806, 0.818]	0.603 [0.596, 0.611]
Hybrid-RRF60	0.685 [0.676, 0.693]	0.509 [0.502, 0.517]	0.348 [0.339, 0.357]	0.741 [0.734, 0.748]	0.520 [0.513, 0.528]
<i>Panel B: sampled-window diagnostic</i>					
Random	0.340	0.193	0.069	0.700	0.222
BM25	0.833 [0.826, 0.841]	0.658 [0.651, 0.666]	0.487 [0.478, 0.497]	0.943 [0.940, 0.947]	0.643 [0.636, 0.650]
MedCPT	0.908 [0.903, 0.913]	0.750 [0.744, 0.757]	0.588 [0.579, 0.597]	0.989 [0.987, 0.990]	0.702 [0.696, 0.709]
NV-Embed-v2	0.967 [0.964, 0.971]	0.851 [0.846, 0.856]	0.719 [0.711, 0.728]	0.993 [0.992, 0.994]	0.820 [0.815, 0.825]
Hybrid-RRF60	0.913 [0.908, 0.919]	0.756 [0.749, 0.762]	0.596 [0.586, 0.605]	0.982 [0.980, 0.984]	0.725 [0.718, 0.731]

Note: Italics denote panel headings; bold indicates the best-performing retriever and its scores in each panel.

Retrieval scores are consistently higher in the sampled-window setting because the search is restricted to a much smaller candidate pool. For this reason, we treat the sampled-window results as a diagnostic rather than as the primary measure of retrieval difficulty. The remaining analysis focuses on the full patient timeline, with Table 11 showing how performance varies with timeline length, difficulty, and temporal reasoning type.

Timeline length has the clearest effect on retrieval difficulty. For patients with only 2–10 visits, even random selection has a Hit@5 of 0.906, and all retrievers are close to saturation. This changes substantially for histories containing 101–1000 visits, where the random baseline falls to 0.031. NV-Embed-v2 still achieves 0.743 Hit@5 in this group, but complete Coverage@5 falls to 0.366. The contrast shows why retrieval scores on short histories should not be interpreted in the same way as scores on long timelines. In long records, identifying at least one relevant encounter remains feasible, while recovering the complete evidence set becomes much more difficult.

Difficulty level shows a related pattern. NV-Embed-v2 has similar Hit@5 for medium and hard questions, at 0.757 and 0.770, respectively, yet Coverage@5 decreases from 0.392 to 0.360. At the same time, the mean number of required gold visits increases from 2.06 to 2.22. This indicates that harder questions are not necessarily difficult because the retriever fails to find any relevant visit. Instead, they more often depend on several pieces of evidence that must be retrieved together.

The reasoning categories show a similar distinction between finding some evidence and recovering enough evidence. Treatment-response and first/last-occurrence questions have relatively strong retrieval performance, while trend and comparison-across-visits questions are more challenging. Trend questions have the lowest NV-Embed-v2 Hit@5 and MRR, at 0.672 and 0.490. These questions often depend on observations distributed across several visits, so retrieving a single relevant encounter is less likely to provide sufficient evidence for the downstream reader. The residual Other category is omitted from this analysis because corresponding stratified retrieval values were not available.

Table 11. Retrieval performance on the full patient timeline stratified by timeline length, difficulty, and temporal reasoning type. Random Hit@5 is included for the timeline-length groups because the probability of retrieving a relevant visit by chance depends strongly on the number of candidate visits. Mean G is included for the difficulty and reasoning-type groups to show the average number of gold evidence visits required by each question.

<i>Panel A: patient timeline length</i>								
Visit group	<i>n</i>	Random Hit@5	BM25 Hit@5	MedCPT Hit@5	NV Hit@5	Hybrid Hit@5	NV Coverage@5	NV MRR
2–10 visits	200	0.906	0.995	0.995	1.000	0.995	0.995	0.923
11–100 visits	1244	0.287	0.848	0.881	0.937	0.909	0.700	0.808
101–1000 visits	9298	0.031	0.598	0.537	0.743	0.648	0.366	0.569
<i>Panel B: difficulty</i>								
Difficulty	<i>n</i>	Mean G	BM25 Hit@5	MedCPT Hit@5	NV Hit@5	Hybrid Hit@5	NV Coverage@5	NV MRR
Easy	2105	1.73	0.727	0.633	0.812	0.749	0.552	0.646
Medium	6502	2.06	0.613	0.577	0.757	0.671	0.392	0.592
Hard	2135	2.22	0.607	0.566	0.770	0.661	0.360	0.596
<i>Panel C: temporal reasoning type</i>								
Reasoning type	<i>n</i>	Mean G	BM25 Hit@5	MedCPT Hit@5	NV Hit@5	Hybrid Hit@5	NV Coverage@5	NV MRR
Treatment response	1299	1.93	0.694	0.684	0.851	0.764	0.497	0.683
First/last occurrence	937	1.51	0.728	0.667	0.830	0.753	0.647	0.692
Before/after	1837	1.99	0.660	0.596	0.825	0.691	0.437	0.651
Recurrence count	623	2.26	0.637	0.570	0.777	0.642	0.416	0.604
Progression	1992	2.17	0.562	0.572	0.749	0.633	0.341	0.583
Comparison across visits	2736	2.10	0.622	0.550	0.727	0.673	0.360	0.563
Trend	1218	2.14	0.585	0.491	0.672	0.655	0.345	0.490

4.3.3. End-to-End and Oracle-Evidence QA

Table 12 gives the primary QA results for all 25 reader and evidence combinations. NV-Embed-v2 is the strongest retrieval condition for every reader on Token F1 and Prometheus mean. The Qwen readers are well ahead of the extractive baselines, but the difference between the 7B and 32B models is modest compared with the effect of supplying better evidence. With NV-Embed-v2, Qwen2.5-7B and Qwen2.5-32B obtain Token F1 scores of 38.98 and 40.08. With Oracle evidence, the scores rise to 45.37 and 46.67. Their Oracle Prometheus means are 4.12 and 4.16, respectively.

Token F1 and BS-RL use all 10,742 QA pairs. Prometheus mean and Correct ≥ 4 use the fixed 1000-question judge subset. Underlining marks the best retrieval condition for each reader, while bold values correspond to Oracle evidence.

We use the same RoBERTa-large BERTScore configuration for every run. The secondary Bio_ClinicalBERT BERTScore and Exact Match results are retained in the supplementary results. We also checked whether the Prometheus judge favored longer generations. It did not: across the Qwen runs, judge score is weakly negatively correlated with answer length (Spearman $\rho \approx -0.13$ to -0.26), and the padding probe changes the mean score by 0.000.

The paired comparisons support the same interpretation without changing the overall ranking. For Qwen2.5-32B, NV-Embed-v2 improves Token F1 over Hybrid-RRF60 by +1.6 points [1.3, 1.9], while Oracle improves over NV-Embed-v2 by +6.6 [6.3, 6.9] and over Hybrid-RRF60 by +8.2 [7.9, 8.5]. The complete reader-by-reader paired contrasts, including Prometheus differences, are reported in Appendix A.2.

Table 12. Reader performance across retrieval and Oracle evidence conditions.

Reader	Evidence Condition	Token F1	BS-RL	Prometheus	Correct ≥ 4 (%)
RoBERTa-base	BM25	8.49	78.92	1.96	10.3
	MedCPT	8.55	78.56	1.98	9.7
	NV-Embed-v2	9.83	79.04	2.10	12.2
	Hybrid-RRF60	8.77	79.20	1.94	10.3
	Oracle	13.51	79.85	2.35	16.6
BioBERT-v1.1	BM25	8.86	81.89	1.94	9.0
	MedCPT	9.13	82.06	1.99	9.4
	NV-Embed-v2	<u>10.34</u>	<u>82.26</u>	<u>2.10</u>	<u>11.5</u>
	Hybrid-RRF60	9.27	82.03	2.00	10.2
	Oracle	14.03	82.76	2.26	14.4
Longformer	BM25	9.16	81.49	1.86	8.4
	MedCPT	9.09	81.57	1.91	9.5
	NV-Embed-v2	<u>10.24</u>	<u>81.89</u>	<u>2.05</u>	<u>10.8</u>
	Hybrid-RRF60	9.56	81.55	1.92	9.1
	Oracle	14.61	82.89	2.39	17.1
Qwen2.5-7B	BM25	36.31	88.62	3.78	65.2
	MedCPT	35.66	88.57	3.72	64.0
	NV-Embed-v2	38.98	89.14	3.89	72.1
	Hybrid-RRF60	37.40	88.83	3.81	68.7
	Oracle	45.37	90.21	4.12	85.0
Qwen2.5-32B	BM25	37.49	88.88	3.84	65.2
	MedCPT	35.98	88.66	3.82	63.4
	NV-Embed-v2	40.08	89.35	3.94	72.8
	Hybrid-RRF60	38.48	89.06	3.86	67.8
	Oracle	46.67	90.47	4.16	87.1

Note: Bold indicates Oracle results; underlining indicates the best retrieval-based score for each reader and metric.

Stratified QA Analysis

The subgroup analyses reuse the same saved predictions and scores as the main evaluation, so no additional model inference is performed. Table 13 brings difficulty, timeline length, and temporal reasoning type into one view and focuses on Token F1, making the Oracle minus NV-Embed-v2 gap easier to compare across groups.

The three stratification axes tell somewhat different stories. Difficulty is associated with lower absolute performance, but the Oracle gap is not monotonic from easy to hard. Timeline length is more informative: the gap is essentially zero for 2–10-visit histories and rises above seven F1 points for both Qwen readers on 101–1000-visit histories. Reasoning type shows a related pattern. Trend and comparison-across-visits questions have the largest Oracle minus NV gaps, whereas first/last-occurrence and recurrence-count questions show smaller differences. These strata are descriptive rather than causal, but they are consistent with retrieval becoming more consequential when the answer depends on evidence distributed across a longer temporal span.

The $\Delta F1$ values are paired Oracle minus NV-Embed-v2 differences calculated from the unrounded scores, with 95% bootstrap confidence intervals.

Sensitivity Analyses

We ran two sensitivity checks to better understand the Oracle gap. The first addresses the fact that retrieval supplies at most five visits while Oracle can contain more. This affects only 20 of the 10,742 questions. Restricting the paired analysis to the 10,722 questions with $G \leq 5$ leaves the Oracle-minus-retrieval differences essentially unchanged, indicating that the aggregate Oracle advantage is not explained by the small number of questions requiring more than five gold visits.

The second sensitivity analysis examines whether the annotated gold visits reach the reader without being shortened. We define `gold_fully_survived` as true only when all

gold visits are included and none of their text is trimmed during context construction. This is a stricter visit-level measure than simply checking whether the required visits are present. As shown in Table 14, this condition holds for 40.72% of NV-Embed-v2 questions and 99.25% of Oracle questions.

Table 13. Qwen Token F1 stratified by difficulty, patient timeline length, and temporal reasoning type.

Stratum	n	Qwen2.5-7B			Qwen2.5-32B		
		NV F1	Oracle F1	ΔF1	NV F1	Oracle F1	ΔF1
<i>Difficulty</i>							
Easy	2105	40.9	47.0	6.1 [5.5, 6.8]	43.7	49.4	5.7 [5.0, 6.4]
Medium	6502	38.9	45.6	6.8 [6.4, 7.1]	39.7	46.8	7.1 [6.7, 7.5]
Hard	2135	37.5	43.0	5.5 [5.0, 6.1]	37.6	43.5	5.9 [5.4, 6.4]
<i>Patient timeline length</i>							
2–10 visits	200	43.2	43.3	0.1 [−1.2, 1.4]	47.5	47.7	0.1 [−1.1, 1.4]
11–100 visits	1244	40.6	43.4	2.7 [2.0, 3.4]	43.6	45.5	1.8 [1.1, 2.5]
101–1000 visits	9298	38.7	45.7	7.0 [6.7, 7.3]	39.4	46.8	7.4 [7.1, 7.7]
<i>Temporal reasoning type</i>							
Before/after	1837	40.1	45.8	5.7 [5.1, 6.3]	42.0	47.6	5.6 [4.9, 6.2]
Comparison across visits	2736	38.6	45.7	7.1 [6.6, 7.6]	38.6	46.4	7.8 [7.2, 8.3]
First/last occurrence	937	41.7	46.3	4.6 [3.7, 5.5]	45.1	49.2	4.1 [3.1, 5.1]
Other	100	44.5	49.3	4.8 [2.0, 7.6]	47.9	49.9	2.0 [−0.5, 4.6]
Progression	1992	37.7	44.5	6.8 [6.3, 7.4]	38.0	44.8	6.8 [6.2, 7.4]
Recurrence count	623	36.8	41.3	4.4 [3.5, 5.4]	37.4	42.0	4.6 [3.6, 5.6]
Treatment response	1299	40.0	45.4	5.4 [4.7, 6.1]	41.9	47.6	5.7 [4.9, 6.6]
Trend	1218	37.8	46.4	8.6 [7.9, 9.5]	38.7	48.2	9.4 [8.4, 10.4]

We examined the 81 Oracle questions (0.75%) that did not meet this criterion. All annotated gold visits were correctly selected and remained in the final input, but their text was proportionally shortened to fit the shared 3500-token context limit. A further check of the cited evidence snippets showed that all snippet segments remained present in 65 of these questions, while 16 lost at least one segment, corresponding to 0.15% of the full benchmark. Because trimming removes text from the ends of visits, a supporting snippet may be lost even when most of the visit text is retained. No missing gold visits or visit-index mapping errors were found.

Within NV-Embed-v2, meeting the complete-visit criterion is associated with gains of 10.3 F1 points for Qwen2.5-7B and 13.9 for Qwen2.5-32B. On the stricter subset of 1277 questions meeting this criterion under every evidence condition, the Oracle-minus-NV gap decreases to +1.3 F1 for Qwen2.5-7B and −0.2 for Qwen2.5-32B, with both Prometheus confidence intervals including zero. The corresponding within-condition results are provided in Appendix A.4.

Panel A reports the percentage of questions for which all annotated gold visits were included without text trimming. Panel B compares reader performance on the 1277 questions meeting this criterion under every evidence condition.

Taken together, these results indicate that delivering the required visits is an important bottleneck in longitudinal QA. Oracle largely removes the retrieval component of this problem, although the shared context limit can still shorten long visits and occasionally remove cited snippets. When we restrict the comparison to questions for which all gold visits are included without trimming under every condition, the difference between retrieval-based and Oracle performance becomes much smaller. In these experiments, reliable delivery of the supporting evidence has a larger effect than increasing the Qwen model from 7B to 32B.

Table 14. Gold-visit preservation during reader context construction.

<i>Panel A: all gold visits included and untrimmed</i>				
Condition	All questions (%)		Judge subset (%)	
BM25	23.42		22.6	
MedCPT	24.33		25.2	
NV-Embed-v2	40.72		41.2	
Hybrid-RRF60	25.84		25.8	
Oracle	99.25		98.6	
<i>Panel B: common complete-visit subset ($n = 1277$; $n_J = 123$)</i>				
Reader	NV F1	Oracle F1	Oracle – NV F1	Oracle – NV Judge
Qwen2.5-7B	45.5	46.8	+1.3 [0.7, 1.9]	−0.073 [−0.213, 0.081]
Qwen2.5-32B	49.6	49.4	−0.2 [−0.8, 0.4]	−0.041 [−0.151, 0.066]

5. Discussion, Limitations, and Future Work

5.1. Main Findings

The filtering results show that valid structure alone is not enough to produce a reliable benchmark. EHRForge generated 27,558 candidate QA pairs, of which 18,867 passed the temporal and structural checks and 10,742 also passed source-visit evidence alignment. This stepwise reduction reflects the role of verification in removing candidates that are well formed but do not satisfy the benchmark’s temporal or evidence-grounding requirements. We therefore treat generation as an initial candidate-production stage rather than as the final benchmark-construction step.

The structured human validation provides additional support for the retained benchmark while also showing where judgment remains subjective. The two human validators produced similar marginal quality rates, with raw agreement between 96.4% and 99.3% across the five criteria. GPT-5.5 was more conservative than either human validator. The rejected-item validation showed a similar pattern. The validators were generally consistent when judging evidence order and whether cited text could be traced back to the source notes. More disagreement arose when deciding whether a question genuinely required evidence from multiple time points, which appears to involve more subjective interpretation. GPT-5.5 is therefore used only as an additional automated check, while the human validation remains the primary assessment of benchmark quality.

The downstream results are clearer when retrieval and answer generation are examined separately. On full patient timelines, NV-Embed-v2 achieves a Hit@5 of 0.770, but complete Coverage@5 is only 0.417. For longitudinal QA, retrieving one relevant encounter is not equivalent to retrieving all of the visits needed for the answer. The timeline-length analysis also shows why top-K results should be interpreted relative to the candidate pool. Hit@5 can be almost automatic for short histories, while the same score represents a much larger improvement over chance for long histories.

The aggregate Oracle comparison initially appears to show a substantial reader advantage. Relative to NV-Embed-v2, Oracle raises Token F1 by 6.4 points for Qwen2.5-7B and by 6.6 points for Qwen2.5-32B. The subgroup analysis shows that this difference is not uniform. It is minimal for histories with 2–10 visits, moderate for histories with 11–100 visits, and above seven F1 points for histories with 101–1000 visits. The gap is also larger for trend and comparison-across-visits questions than for first/last-occurrence or recurrence-count questions. This pattern is more consistent with a longitudinal evidence-localization problem than with a reader that fails in the same way across all question types.

The evidence-survival analysis helps explain this result. Under NV-Embed-v2, both Qwen models perform substantially better when all annotated gold visits are included without text trimming. We therefore repeated the comparison on the 1277 questions meeting this criterion under every evidence condition. On this common subset, the Oracle

advantage falls to +1.3 F1 for Qwen2.5-7B and -0.2 for Qwen2.5-32B, while the Prometheus intervals include zero. The separate $G \leq 5$ analysis also shows that the result is not explained by Oracle having access to more than five visits. These findings motivate a three-stage view of the downstream task: retrieving the relevant visits, preserving them within the reader's context budget, and synthesizing the final answer. Keeping these stages separate makes it easier to identify where performance is lost.

Model size has a smaller effect in these zero-shot experiments. With Oracle evidence, moving from Qwen2.5-7B to Qwen2.5-32B increases Token F1 from 45.37 to 46.67 and Prometheus mean from 4.12 to 4.16. These gains are smaller than the losses observed when complete evidence does not reach the reader. We interpret this finding only within the present benchmark and evaluation protocol, rather than as a general statement about model capacity.

5.2. Limitations

Several limitations should be considered when interpreting these results. First, the study is based on data from a single hospital, so both the composition of the benchmark and the behavior of the pipeline reflect the characteristics of that local setting.

Differences in documentation practices, visit frequency, clinical specialties, and workflow can all affect the types of temporal questions that are generated and the proportion of candidates that pass verification. The retention rates reported here should therefore be viewed as specific to this deployment rather than as fixed expectations for EHRForge in other institutions. The full set of 10,742 retained QA pairs was also not reviewed by clinicians. Although the 1000-item human validation provides evidence about the quality of the retained benchmark, some generation and labeling errors are likely to remain, particularly in questions involving implicit progression, treatment response, or evidence spread across widely separated visits.

Exact evidence matching is deliberately conservative. It gives us a clear provenance test, but it does not establish semantic entailment and, when ellipsis matching is used, it does not guarantee that matched fragments are contiguous or preserve their original within-visit order. The dual-human retained-set validation exposed a small number of such residual provenance-composition errors despite high evidence-match rates. A future semantic verifier or stricter ordered-span matcher would need to improve this behavior without weakening auditability.

The retrieval analysis is also sensitive to the size of each patient's candidate pool. Hit@5 can be nearly trivial for short histories and much harder for long ones, which is why we report exact random baselines and MRR alongside top- K metrics. The retrieval subgroup analysis covers timeline length, difficulty, and temporal reasoning type. For readability, the main stratified table reports Hit@5 for all retrievers and adds Coverage@5 and MRR for NV-Embed-v2. A fully expanded matrix of Recall@5, Coverage@5, and random baselines for every retriever within every stratum would be a useful supplementary analysis.

For the retrieval and reader experiments, confidence intervals use item-level bootstrap resampling over QA keys. Because a patient can contribute several questions, this procedure does not fully capture dependence between patients. A patient-cluster bootstrap would therefore provide a useful confirmatory analysis. Prometheus estimates are also less stable in small subgroups, so we suppress them when fewer than 30 judged questions are available.

Reproducibility in this setting does not require releasing the underlying clinical notes. We make the construction and evaluation process transparent by documenting the workflow, schemas, prompts, verification rules, evaluation code, and aggregate results. When EHRForge is applied to another governed clinical corpus, the resulting bench-

mark may differ because of local documentation practices, the generation model, and the prompt configuration.

5.3. Ethics, Privacy, and Reproducibility

EHRForge was evaluated on longitudinal clinical data used under institutional governance and privacy requirements. The original clinical notes, patient identifiers, complete visit histories, and verbatim source records are not published. Although the benchmark questions, answers, and labels are generated synthetically and de-identified, they are still derived from real patient timelines and may retain sensitive contextual or temporal information. For this reason, the benchmark is provided through controlled access rather than as an open dataset. Access is subject to institutional approval, licensing and data-use conditions, and is also limited to research and evaluation purposes. The benchmark is not intended to use in clinical decision-making.

Researchers may inspect the released methodology and request access to the synthetic benchmark under the applicable governance framework. They may also adapt EHRForge to construct local temporal QA benchmarks from appropriately authorized data under their own institutional requirements. This controlled-access model preserves the methodological contribution while ensuring that sensitive source information, such as original clinical notes, patient identifiers, and complete patient timelines, is not distributed as part of the benchmark release.

5.4. Future Work

An important next step is to evaluate EHRForge in a second clinical setting. Applying the same construction and verification workflow to another health system would help determine which retention patterns and failure modes are consistent across institutions and which are specific to the current corpus. A multi-site study could also show how differences in visit frequency, specialty mix, and documentation practices affect the types of temporal questions that are generated.

A further priority is broader clinical expert adjudication beyond the present validation by two human domain researchers. More adjudication, particularly for progression, treatment response, recurrence, and long-range comparison, would help characterize the remaining errors after deterministic filtering and resolve cases where the two validators disagree about the sufficiency of evidence or whether a question genuinely requires multiple time points. It would also provide a better environment for semantic-entailment methods that could support exact matching without replacing its provenance. Disagreements among the automatic filters, the LLM judge, and human validations also help identify where the verification pipeline can be improved.

For future modeling work, improving retrieval over long patient histories appears especially important. Approaches such as hierarchical indexing, temporally aware reranking, multi-hop retrieval, and event-centered representations may help recover evidence that is spread across many visits. Our results suggest that end-to-end performance is often limited by whether the necessary evidence reaches the reader, rather than by reader capacity alone. Future evaluations should therefore report not only answer quality, but also whether the supporting evidence is retrieved and used in a temporally consistent way.

6. Conclusions

We introduced EHRForge as a workflow for constructing temporal QA benchmarks from longitudinal clinical notes while preserving both evidence provenance and institutional governance. In our deployment, 27,558 generated candidates were reduced to 10,742 QA pairs after temporal, structural, and source-visit evidence checks. Human val-

idation of rejected and retained items further showed which verification decisions were consistent and which still depended on human interpretation.

The downstream experiments point to evidence delivery as a major challenge. NV-Embed-v2 achieved 77.0% Hit@5 on full patient timelines, but complete Coverage@5 was only 41.7%, meaning that many questions still lacked one or more required visits in the retrieved context. The Qwen readers clearly outperformed the extractive baselines, but increasing model size from 7B to 32B produced smaller gains than improving the evidence available to the reader. This pattern was especially clear on long patient histories, where the Oracle advantage exceeded seven F1 points, but became much smaller when all annotated evidence survived context construction.

These results suggest that performance in longitudinal QA depends not only on answer generation, but also on finding and preserving the complete set of supporting evidence. Because the underlying clinical data remain governed, EHRForge is designed to be transferred through its construction and evaluation procedure rather than through unrestricted release of the instantiated dataset. The timeline representation, sampling strategy, generation schema, verification rules, and reproducibility artifacts provide the components needed to apply the workflow in other appropriately governed settings.

Author Contributions: Conceptualization, Q.A.Q.T., H.Q.T. and M.L.; methodology, Q.A.Q.T.; validation, Q.A.Q.T.; data curation, Q.A.Q.T.; writing—original draft preparation, Q.A.Q.T.; writing—review and editing, H.Q.T., Y.S. and M.L.; supervision, M.L.; funding acquisition, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw clinical notes, original patient timelines, document identifiers, patient identifiers, instantiated benchmark question and answer pairs, and verbatim evidence snippets are not publicly available because of patient privacy, institutional governance, and data-use restrictions. The benchmark is synthetic at the question and answer and labeling levels but remains grounded in patient-derived timelines and evidence. Subject to institutional and software-licensing approval, the authors will release non-sensitive EHRForge artifacts, including preprocessing and sampling specifications, prompt templates, the JSON schema, filtering and evidence-matching rules, evaluation scripts, experimental baselines and aggregate benchmark statistics. Fully synthetic benchmarks will be made available upon reasonable request. The source code and reproducibility artifacts are available at <https://github.com/20127304-AQ/EHRForge-A-Reproducible-Workflow-For-Benchmark-Generation-from-EHRs> (accessed on 8 October 2026).

Acknowledgments: Quang An Quoc Tran is a master student supervised by Dr. Ming Liu, Huy Quoc To is a PhD student supervised by Dr. Ming Liu. Dr. Ming Liu was supported by Australian Research Linkage Project (LP220200746) and Deakin Minor Research Project Scheme. Dr. Yongpan Sheng was supported by the National Natural Science Foundation of China (Grant No. 62566007), Fundamental Research Funds for the Central Universities (No. SWU-KR24008), Key Laboratory of Data Science and Smart Education, Hainan Normal University, Ministry of Education (No. DSIE202206).

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Supplementary Experimental Results

Appendix A.1. Representative Human-Validation Examples

This appendix presents representative examples from the three human-validation stages: Round 1 rejected candidates, Round 2 rejected candidates, and the final retained-set validation after Round 2. The examples illustrate both agreement and disagreement between human validators, including the all-five-pass assessment introduced in Table 8. The Round 1 validation report and the records available for Examples F-A and F-B do

not preserve a separate generated-answer field; we therefore do not infer those missing answers. In contrast, the original QA records for Examples F-C and F-D include the generated answers, which are reproduced alongside their evidence and de-identified source excerpts.

Appendix A.1.1. Round 1 Rejected Candidates

Example R1-A: validators agree

Question: After the patient returned from CTPA (progress note 2019-08-30 13:10–13:12), what oxygen saturation change was noted and what action was taken?

Generated evidence: “pt returned post ctpa cont nasal prongs oxygen as sao2 low to 90% in room air alarming go down to 87%”

De-identified source text: “Progress Note pt returned post ctpa cont nasal prongs oxygen as sao2 low to 90% in room air alarming go down to 87% voided declined analgesia pain remains 4/10 await trop 2345 admission to cardio suma”

Validation outcome: Both validators marked Insufficient Evidence = False, Not-In-Order Evidence = False, Possible Over Inference = False, and Not Temporal Evidence = True.

Interpretation: The evidence directly supports the oxygen desaturation and continuation of nasal-prong oxygen, but all required information is contained in one visit. Both validators therefore agreed that the item was not genuinely longitudinal.

Example R1-B: validators disagree

Question: Was the patient on supplemental oxygen continuously during the daytime shift on 08/07/2020 11:01, and what did she do with the oxygen?

Generated evidence: “Pt refusing O2 therapy via NP removing by herself.”

De-identified source text: “Progress Note Pt disorientated to person time place. Min oral intake, 1.5ltr FR, HFR however refusing to stay in bed. Ambulating without assistance. BO x 2, Urine voided in toilet (full assist) & bed pan. Pt refusing O2 therapy via NP removing by herself. NIL pain stated, IV canula in RCF. Incontinent of urine.”

Validation outcome: Both validators marked Not-In-Order Evidence = False, Possible Over Inference = False, and Not Temporal Evidence = True. Validation 1 marked Insufficient Evidence = True, whereas Validation 2 marked Insufficient Evidence = False.

Interpretation: The disagreement concerns evidence sufficiency rather than chronology. One validator considered a single observation of oxygen refusal/removal insufficient to establish continuity across the whole daytime shift, whereas the other considered the note sufficient to show that oxygen was not continuous.

Appendix A.1.2. Round 2 Rejected Candidates

Example R2-A: validators agree

Question: Across 2018-11-21 and 2018-12-03, what was the trend in lactate dehydrogenase (LDH) value?

Generated answer: LDH increased from 186 U/L on 2018-11-21 to 281 U/L on 2018-12-03.

Generated evidence: “SERUM LACTATE DEHYDROGENASE ... 186 U/L

(240–480) ... 21/11/18"

"SERUM/PLASMA BIOCHEMISTRY ... 281 U/L (240–480)"

De-identified source text: "=== VISIT 338 | 2018-11-21 04:51:16 === SERUM LACTATE DEHYDROGENASE Lactate Dehydrogenase: 186 U/L (240–480) *
=== VISIT 347 | 2018-12-03 04:24:39 === SERUM LACTATE DEHYDROGENASE Lactate Dehydrogenase: 281 U/L (240–480)"

Validation outcome: Both validators marked Insufficient Evidence = False, Not-In-Order Evidence = False, Possible Over Inference = False, Not Temporal Evidence = False, and Exact Match = False.

Interpretation: The numerical trend is supported, but the second generated evidence snippet contains "SERUM/PLASMA BIOCHEMISTRY," which does not appear in the corresponding source visit. Both validators therefore agreed that the evidence citation was not an exact source-text match.

Example R2-B: validators disagree

Question: From the 2019-01-25 INR report to the 2019-01-30 INR report, what were the INR values and did the INR increase or decrease?

Generated answer: INR decreased slightly from 1.1 on 2019-01-25 to 1.0 on 2019-01-30.

Generated evidence: "2019-01-25 10:09:45 INR : 1.1 (<1.3)"
"2019-01-30 03:14:14 INR : 1.0 (<1.3)"

De-identified source text: "=== VISIT 1 | 2019-01-25 10:09:45 === Coagulation Report (Plasma) Prothrombin time : 12.8 secs (10.0–14.0) Prothrombin Ratio (INR) : 1.1 (<1.3) Please note the Reference Range for Fibrinogen and TCT have been changed. === VISIT 7 | 2019-01-30 03:14:14 === Coagulation Report (Plasma) Prothrombin time : 12.4 secs (10.0–14.0) Prothrombin Ratio (INR) : 1.0 (<1.3) Please note the Reference Range for Fibrinogen and TCT have been changed."

Validation outcome: Both validators marked Insufficient Evidence = False, Not-In-Order Evidence = False, Possible Over Inference = False, and Not Temporal Evidence = False. Validation 1 marked Exact Match = True, whereas Validation 2 marked Exact Match = False.

Interpretation: The disagreement concerns literal provenance rather than clinical meaning. The generated snippets shorten the source label "Prothrombin Ratio (INR)" to "INR." One validator accepted that normalization, while the other required a stricter verbatim match.

Appendix A.1.3. Final-Round Validation of Retained QA Pairs

Example F-A: failure of evidence-match

Question: What did the CT abdomen/pelvis on 2019-03-01 show regarding small bowel obstruction, and how did it compare to the plain film concern for dilated bowel?

Generated evidence: "No obstruction is evident ... oral contrast is seen to pass through to the large bowel."

De-identified source text: "The small bowel is non dilated and oral contrast is seen to pass through to the large bowel. No obstruction is evident."

Validation outcome: Evidence match = False.

Interpretation: Both evidence fragments occur in the same source visit, so the automatic matcher accepted them. However, the generated snippet reverses their

original order. The citation is therefore not a faithful reconstruction of the source passage even though the underlying clinical information is present.

Example F-B: failure of evidence-match

Question: Was CRP higher earlier in the illness course and then lower by 2019-01-28, based on the lab trend shown in the notes?

Generated evidence: “25/01/19 ... CRP ... 139”
“28/01/19 ... CRP ... 71 mg/L”

De-identified source text: “SERUM C-REACTIVE PROTEIN (CRP) ... CRP (0–10) ... 25/01/19 ... 139 ... 28/01/19 ... 71”

Validation outcome: Evidence match = False.

Interpretation: The automatic matcher found the date, the term “CRP,” and the laboratory value somewhere in the same note. However, these elements do not occur together in the exact sequence implied by the generated evidence snippet. Because short repeated terms such as “CRP” can appear elsewhere in a long note, segment-based ellipsis matching can occasionally produce a false positive.

Example F-C: validators agree on all five criteria

Question: Was NCare registration for ongoing Resource 2.0 provision completed on 2020-11-16 or 2020-11-18, based on the dietetic documentation?

Generated answer: It was documented as completed on 2020-11-18 (registered on NCare for Resource 2.0 at hospital tender price), following the 2020-11-16 education about registering.

Generated evidence: “Discussed registering with NCare for ongoing ONS provision at home.”

“Pt registered on NCare for Resource 2.0 @ hospital tender price 2. D/C summary completed”

De-identified source text: “=== VISIT 7 | 2020-11-16 04:45:38 === ... Discussed registering with NCare for ongoing ONS provision at home. Pt daughter requested sign up for Resource 2.0. DT will also provide hospital tender prices, purchasing and details prior to d/c. ... === VISIT 24 | 2020-11-18 02:06:19 === 1. Pt registered on NCare for Resource 2.0 @ hospital tender price 2. D/C summary completed”

Validation outcome: Both validators marked Evidence match = True, Question validity = True, Answer correctness = True, Evidence grounding = True, and Temporal consistency = True.

Interpretation: Both validators agreed that the generated answer faithfully distinguished the earlier discussion of NCare registration on 16 November from the documented completion on 18 November. The supporting passages are linked to separate visits and establish the required temporal sequence, so the item meets all five human-validation criteria simultaneously.

Example F-D: validators disagree on joint validity

Question: What change in antibiotic route was planned around 30/01/2019?

Generated answer: The plan was to switch from IV ceftriaxone to oral antibiotics the next morning (30/01/2019 morning).

Generated evidence: “Switch ceftriaxone to oral tomorrow morning aim discharge home tomorrow morning”

“Pt has been swapped to oral antibiotics.”

De-identified source text: “=== VISIT 4 | 2019-01-29 22:27:15 === ... Anticipated

Discharge Date: 31/01/19 ... Assessment and Plan ... Switch ceftriaxone to oral tomorrow morning aim discharge home tomorrow morning ... === VISIT 12 | 2019-01-30 09:28:53 === Progress Note ... Pt has been swapped to oral antibiotics. Pt for ?d/c home tommorow morning ..."

Validation outcome: Validator 1 marked all five criteria True. Validator 2 marked Evidence match = True, Question validity = True, and Evidence grounding = True, but Answer correctness = False and Temporal consistency = False.

Interpretation: The disagreement concerns how the relative expression "tomorrow morning" should be interpreted in relation to the recorded clinical timestamps. The earlier note describes a planned switch from intravenous to oral antibiotics, while the later note confirms that the switch had already occurred. Validator 1 accepted the date stated in the generated answer, whereas Validator 2 considered its temporal interpretation insufficiently supported by the available documentation. This example illustrates how relative temporal expressions can lead to different human judgments, even when the supporting evidence is correctly linked to the source visits.

Appendix A.2. Paired QA Contrasts

Table A1 gives the complete paired bootstrap contrasts for NV-Embed-v2 versus Hybrid-RRF60 and for Oracle versus the two strongest retrieval conditions. Prometheus differences use the 1–5 scale.

Table A1. Paired reader-performance contrasts.

Reader	NV – Hybrid		Oracle – NV		Oracle – Hybrid	
	$\Delta F1$	$\Delta Judge$	$\Delta F1$	$\Delta Judge$	$\Delta F1$	$\Delta Judge$
RoBERTa-base	+1.1 [0.8, 1.3]	+0.168 [0.090, 0.243]	+3.7 [3.4, 4.0]	+0.245 [0.167, 0.317]	+4.7 [4.5, 5.0]	+0.413 [0.331, 0.490]
BioBERT-v1.1	+1.1 [0.8, 1.3]	+0.098 [0.019, 0.163]	+3.7 [3.4, 4.0]	+0.164 [0.097, 0.234]	+4.8 [4.5, 5.0]	+0.262 [0.187, 0.329]
Longformer	+0.7 [0.4, 0.9]	+0.131 [0.064, 0.199]	+4.4 [4.1, 4.7]	+0.345 [0.270, 0.421]	+5.1 [4.7, 5.4]	+0.476 [0.397, 0.554]
Qwen2.5-7B	+1.6 [1.3, 1.8]	+0.075 [0.010, 0.138]	+6.4 [6.1, 6.6]	+0.239 [0.173, 0.305]	+8.0 [7.7, 8.2]	+0.314 [0.255, 0.372]
Qwen2.5-32B	+1.6 [1.3, 1.9]	+0.077 [0.015, 0.131]	+6.6 [6.3, 6.9]	+0.223 [0.170, 0.279]	+8.2 [7.9, 8.5]	+0.300 [0.243, 0.355]

Appendix A.3. Secondary Reader Metrics

Table A2 reports the primary RoBERTa-large BERTScore together with the secondary Bio_ClinicalBERT BERTScore and Exact Match for all 25 reader and evidence-condition runs. Including both variants of the BERTScore provides a direct comparison between the general semantic metric used in the main evaluation and its counterpart in the clinical-domain. Exact Match is not used as a headline metric because the benchmark is formulated as free-text question answering, where semantically correct responses may differ in wording from the reference answer. All Exact Match values are at or below 0.1% after rounding to one decimal percentage point.

BS-RL denotes BERTScore-F1 computed with RoBERTa-large, while BS-Bio denotes BERTScore-F1 computed with Bio_ClinicalBERT. Exact Match uses SQuAD normalization.

Table A2. BERTScore and Exact Match across reader and evidence conditions.

Reader	Evidence Condition	BS-RL	BS-Bio	Exact Match (%)
RoBERTa-base	BM25	78.92	54.84	0.1
	MedCPT	78.56	54.71	0.1
	NV-Embed-v2	79.04	55.55	0.1
	Hybrid-RRF60	79.20	55.18	0.1
	Oracle	79.85	57.43	0.1

Table A2. Cont.

Reader	Evidence Condition	BS-RL	BS-Bio	Exact Match (%)
BioBERT-v1.1	BM25	81.89	56.93	0.1
	MedCPT	82.06	57.33	0.1
	NV-Embed-v2	82.26	57.99	0.1
	Hybrid-RRF60	82.03	57.26	0.0
	Oracle	82.76	59.57	0.1
Longformer-base	BM25	81.49	56.89	0.1
	MedCPT	81.57	56.95	0.1
	NV-Embed-v2	81.89	57.68	0.1
	Hybrid-RRF60	81.55	57.11	0.1
	Oracle	82.89	59.83	0.1
Qwen2.5-7B	BM25	88.62	71.84	0.0
	MedCPT	88.57	71.70	0.0
	NV-Embed-v2	89.14	73.13	0.0
	Hybrid-RRF60	88.83	72.38	0.0
	Oracle	90.21	75.53	0.0
Qwen2.5-32B	BM25	88.88	72.18	0.0
	MedCPT	88.66	71.68	0.0
	NV-Embed-v2	89.35	73.44	0.0
	Hybrid-RRF60	89.06	72.69	0.0
	Oracle	90.47	76.05	0.1

Appendix A.4. Evidence-Survival Breakdown

Table A3 reports reader performance according to whether all annotated gold visits were included without text trimming in the final reader context. The groups are defined separately for each evidence condition and should therefore be interpreted as within-condition comparisons rather than paired comparisons between NV-Embed-v2 and Oracle.

Table A3. Qwen performance by complete gold-visit preservation.

Panel A: Token F1 and BERTScore					
Condition	Survived	<i>n</i>	Reader	F1	BS-RL
NV-Embed-v2	True	4374	Qwen2.5-7B	45.1 [44.7, 45.5]	90.0 [90.0, 90.1]
NV-Embed-v2	False	6368	Qwen2.5-7B	34.8 [34.4, 35.1]	88.5 [88.5, 88.6]
NV-Embed-v2	True	4374	Qwen2.5-32B	48.3 [47.9, 48.8]	90.6 [90.6, 90.7]
NV-Embed-v2	False	6368	Qwen2.5-32B	34.4 [34.0, 34.8]	88.5 [88.4, 88.5]
Oracle	True	10,661	Qwen2.5-7B	45.4 [45.1, 45.6]	90.2 [90.2, 90.3]
Oracle	False	81	Qwen2.5-7B	45.0 [41.6, 48.8]	90.3 [89.7, 90.8]
Oracle	True	10,661	Qwen2.5-32B	46.7 [46.4, 47.0]	90.5 [90.4, 90.5]
Oracle	False	81	Qwen2.5-32B	46.7 [43.5, 49.9]	90.6 [90.0, 91.1]
Panel B: Prometheus metrics					
Condition	Survived	<i>n_j</i>	Reader	Judge	Correct _{≥4} (%)
NV-Embed-v2	True	412	Qwen2.5-7B	4.25 [4.19, 4.32]	87.1 [84.0, 90.3]
NV-Embed-v2	False	588	Qwen2.5-7B	3.63 [3.55, 3.71]	61.6 [57.4, 65.3]
NV-Embed-v2	True	412	Qwen2.5-32B	4.24 [4.17, 4.31]	88.1 [85.0, 91.3]
NV-Embed-v2	False	588	Qwen2.5-32B	3.72 [3.66, 3.79]	62.1 [57.9, 66.1]
Oracle	True	986	Qwen2.5-7B	4.13 [4.08, 4.17]	85.1 [82.9, 87.1]
Oracle	False	14	Qwen2.5-7B	—	—
Oracle	True	986	Qwen2.5-32B	4.16 [4.12, 4.21]	87.2 [85.1, 89.3]
Oracle	False	14	Qwen2.5-32B	—	—

References

- Johnson, A.E.W.; Pollard, T.J.; Shen, L.; Lehman, L.W.H.; Feng, M.; Ghassemi, M.; Moody, B.; Szolovits, P.; Celi, L.A.; Mark, R.G. MIMIC-III, a Freely Accessible Critical Care Database. *Sci. Data* **2016**, *3*, 160035. <https://doi.org/10.1038/sdata.2016.35>.
- Johnson, A.E.W.; Bulgarelli, L.; Shen, L.; Gayles, A.; Shammout, A.; Horng, S.; Pollard, T.J.; Hao, S.; Moody, B.; Gow, B.; et al. MIMIC-IV, a Freely Accessible Electronic Health Record Dataset. *Sci. Data* **2023**, *10*, 1. <https://doi.org/10.1038/s41597-022-01899-x>.
- Goldberger, A.L.; Amaral, L.A.N.; Glass, L.; Hausdorff, J.M.; Ivanov, P.C.; Mark, R.G.; Mietus, J.E.; Moody, G.B.; Peng, C.K.; Stanley, H.E. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. *Circulation* **2000**, *101*, e215–e220. <https://doi.org/10.1161/01.CIR.101.23.e215>.
- Bardhan, J.; Roberts, K.; Wang, D.Z. Question Answering for Electronic Health Records: Scoping Review of Datasets and Models. *J. Med. Internet Res.* **2024**, *26*, e53636. <https://doi.org/10.2196/53636>.
- Pampari, A.; Raghavan, P.; Liang, J.; Peng, J. emrQA: A Large Corpus for Question Answering on Electronic Medical Records. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 2357–2368. <https://doi.org/10.18653/v1/D18-1258>.
- Wang, P.; Shi, T.; Reddy, C.K. Text-to-SQL Generation for Question Answering on Electronic Medical Records. In Proceedings of the Web Conference 2020, New York, NY, USA, 20–24 April 2020; pp. 350–361. <https://doi.org/10.1145/3366423.3380120>.
- Lee, G.; Hwang, H.; Bae, S.; Kwon, Y.; Shin, W.; Yang, S.; Seo, M.; Kim, J.Y.; Choi, E. EHRSQL: A Practical Text-to-SQL Benchmark for Electronic Health Records. In *Proceedings of the Advances in Neural Information Processing Systems*; Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A., Eds.; Curran Associates, Inc.: San Jose, CA, USA, 2022; Volume 35, pp. 15589–15601. <https://doi.org/10.52202/068431-1134>.
- Park, J.; Cho, Y.; Lee, H.; Choo, J.; Choi, E. Knowledge Graph-based Question Answering with Electronic Health Records. In *Proceedings of the 6th Machine Learning for Healthcare Conference*; Proceedings of Machine Learning Research; Jung, K., Yeung, S., Sendak, M., Sjoding, M., Ranganath, R., Eds.; PMLR: New York, NY, USA, 2021; Volume 149, pp. 36–53.
- Lehman, E. *Learning to Ask Like a Physician: A Discharge Summary Clinical Questions (DiSCQ) Dataset*; Version 1.0; PhysioNet: Cambridge, MA, USA, 2022. <https://doi.org/10.13026/7v8e-h745>.
- Soni, S.; Gudala, M.; Pajouhi, A.; Roberts, K. RadQA: A Question Answering Dataset to Improve Comprehension of Radiology Reports. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*; Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., et al., Eds.; European Language Resources Association: Marseille, France, 2022; pp. 6250–6259.
- Bardhan, J.; Colas, A.; Roberts, K.; Wang, D.Z. DrugEHRQA: A Question Answering Dataset on Structured and Unstructured Electronic Health Records for Medicine Related Queries. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*; European Language Resources Association: Marseille, France, 2022; pp. 1083–1097.
- Moon, S.; He, H.; Jia, H.; Liu, H.; Fan, J.W. Extractive clinical question-answering with multianswer and multifocus questions: Data set development and evaluation study. *JMIR AI* **2023**, *2*, e41818. <https://doi.org/10.2196/41818>.
- Kweon, S.; Kim, J.; Kwak, H.; Cha, D.; Yoon, H.; Kim, K.; Yang, J.; Won, S.; Choi, E. EHRNoteQA: An LLM Benchmark for Real-World Clinical Practice Using Discharge Summaries. In *Proceedings of the Advances in Neural Information Processing Systems*; NeurIPS Proceedings: San Diego, CA, USA, 2024; Volume 37, pp. 124575–124611. <https://doi.org/10.52202/079017-3958>.
- Cui, H.; Unell, A.; Chen, B.; Fries, J.A.; Alsentzer, E.; Koyejo, S.; Shah, N.H. TIMER: Temporal Instruction Modeling and Evaluation for Longitudinal Clinical Records. *npj Digit. Med.* **2025**, *8*, 577. <https://doi.org/10.1038/s41746-025-01965-9>.
- Styler IV, W.F.; Bethard, S.; Finan, S.; Palmer, M.; Pradhan, S.; de Groen, P.C.; Erickson, B.; Miller, T.; Lin, C.; Savova, G.; et al. Temporal Annotation in the Clinical Domain. *Trans. Assoc. Comput. Linguist.* **2014**, *2*, 143–154. https://doi.org/10.1162/tacl_a_00172.
- Bethard, S.; Savova, G.; Chen, W.T.; Derczynski, L.; Pustejovsky, J.; Verhagen, M. SemEval-2016 Task 12: Clinical TempEval. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*; Association for Computational Linguistics: San Diego, CA, USA, 2016; pp. 1052–1062. <https://doi.org/10.18653/v1/S16-1165>.
- Du, X.; Shao, J.; Cardie, C. Learning to Ask: Neural Question Generation for Reading Comprehension. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 1342–1352. <https://doi.org/10.18653/v1/P17-1123>.
- Bai, F.; Harrigan, K.; Stremmel, J.; Hassanzadeh, H.; Saeedi, A.; Dredze, M. Give Me Some Hard Questions: Synthetic Data Generation for Clinical QA. *arXiv* **2024**. <https://doi.org/10.48550/arXiv.2412.04573>.
- Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W.; Salakhutdinov, R.; Manning, C.D. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In Proceedings of the EMNLP, Brussels, Belgium, 31 October–4 November 2018; pp. 2369–2380. <https://doi.org/10.18653/v1/D18-1259>.
- Geburu, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J.W.; Wallach, H.; Daumé III, H.; Crawford, K. Datasheets for Datasets. *Commun. ACM* **2021**, *64*, 86–92. <https://doi.org/10.1145/3458723>.

21. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.t.; Rocktäschel, T.; et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In Proceedings of the 34th International Conference on Neural Information Processing Systems, NeurIPS '20, Red Hook, NY, USA, 6–12 December 2020.
22. Zhao, Z.; Ying, H.; Zhong, Y.; Yu, S. DR.EHR: Dense Retrieval for Electronic Health Record with Knowledge Injection and Synthetic Data. *arXiv* **2025**. <https://doi.org/10.48550/arXiv.2507.18583>.
23. Cao, L.; Chen, Q.; Guo, Y. EHR-RAG: Bridging Long-Horizon Structured Electronic Health Records and Large Language Models via Enhanced Retrieval-Augmented Generation. *arXiv* **2026**. <https://doi.org/10.48550/arXiv.2601.21340>.
24. Chen, W.; Bai, T.; Su, J.; Luan, J.; Liu, W.; Shi, C. KG-Retriever: Efficient Knowledge Indexing for Retrieval-Augmented Large Language Models. *arXiv* **2025**. <https://doi.org/10.48550/arXiv.2412.05547>.
25. Qu, Z.; Färber, M. GRAIL: Geometry-Aware Retrieval-Augmented Inference with LLMs over Hyperbolic Representations of Patient Trajectories. *arXiv* **2026**. <https://doi.org/10.48550/arXiv.2602.12828>.
26. Qu, Z.; Färber, M. Vital Trace: Protocol-Constrained Patient-State Reasoning for Longitudinal Clinical Trajectories. *arXiv* **2026**. <https://doi.org/10.48550/arXiv.2602.12833>.
27. Frew, M.; Bheda, N.; Tripp, B. FHIRPath-QA: Executable Question Answering over FHIR Electronic Health Records. *arXiv* **2026**. <https://doi.org/10.48550/arXiv.2602.23479>.
28. Beltagy, I.; Peters, M.E.; Cohan, A. Longformer: The Long-Document Transformer. *arXiv* **2020**. <https://doi.org/10.48550/arXiv.2004.05150>.
29. Zaheer, M.; Guruganesh, G.; Dubey, K.A.; Ainslie, J.; Alberti, C.; Ontañón, S.; Pham, P.; Ravula, A.; Wang, Q.; Yang, L.; et al. Big Bird: Transformers for Longer Sequences. In Proceedings of the Advances in Neural Information Processing Systems, Virtual, 6–12 December 2020; Volume 33, pp. 17283–17297.
30. Li, Y.; Wehbe, R.M.; Ahmad, F.S.; Wang, H.; Luo, Y. Clinical-Longformer and Clinical-BigBird: Transformers for Long Clinical Sequences. *arXiv* **2022**. <https://doi.org/10.48550/arXiv.2201.11838>.
31. Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C.H.; Kang, J. BioBERT: A Pre-trained Biomedical Language Representation Model for Biomedical Text Mining. *Bioinformatics* **2020**, *36*, 1234–1240. <https://doi.org/10.1093/bioinformatics/btz682>.
32. Alsentzer, E.; Murphy, J.R.; Boag, W.; Weng, W.H.; Jin, D.; Naumann, T.; McDermott, M.B.A. Publicly Available Clinical BERT Embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*; Association for Computational Linguistics: San Diego, CA, USA, 2019; pp. 72–78. <https://doi.org/10.18653/v1/W19-1909>.
33. Yang, X.; Chen, A.; PourNejatian, N.; Shin, H.C.; Smith, K.E.; Parisien, C.; Compas, C.; Martin, C.; Costa, A.B.; Flores, M.G.; et al. A Large Language Model for Electronic Health Records. *npj Digit. Med.* **2022**, *5*, 194. <https://doi.org/10.1038/s41746-022-00742-2>.
34. Choi, E.; Hewlett, D.; Lacoste, A.; Polosukhin, I.; Uszkoreit, J.; Berant, J. Hierarchical Question Answering for Long Documents. *arXiv* **2016**. <https://doi.org/10.48550/arXiv.1611.01839>.
35. Liu, Y.; Hashimoto, K.; Zhou, Y.; Yavuz, S.; Xiong, C.; Yu, P. Dense Hierarchical Retrieval for Open-domain Question Answering. In *Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2021*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2021; pp. 188–200. <https://doi.org/10.18653/v1/2021.findings-emnlp.19>.
36. OpenAI. Introducing GPT-5.4 Mini and Nano. 2026. Available online: <https://openai.com/index/introducing-gpt-5-4-mini-and-nano/> (accessed on 7 October 2026).
37. Cohen, J. A Coefficient of Agreement for Nominal Scales. *Educ. Psychol. Meas.* **1960**, *20*, 37–46. <https://doi.org/10.1177/001316446002000104>.
38. OpenAI. Introducing GPT-5.5. 2026. Available online: <https://openai.com/index/introducing-gpt-5-5/> (accessed on 7 October 2026).
39. Wilson, E.B. Probable Inference, the Law of Succession, and Statistical Inference. *J. Am. Stat. Assoc.* **1927**, *22*, 209–212. <https://doi.org/10.1080/01621459.1927.10502953>.
40. Robertson, S.E.; Walker, S. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the SIGIR'94: Proceedings of the Seventeenth Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval, Organised by Dublin City University*; Springer: Berlin/Heidelberg, Germany, 1994; pp. 232–241. https://doi.org/10.1007/978-1-4471-2099-5_24.
41. Jin, Q.; Dhingra, B.; Cohen, W.W.; Lu, X. MedCPT: Contrastive Pre-trained Transformers with Large-scale PubMed Search Logs for Zero-shot Biomedical Information Retrieval. *Bioinformatics* **2023**, *39*, btad651. <https://doi.org/10.1093/bioinformatics/btad651>.
42. Lee, C.; Roy, R.; Xu, M.; Raiman, J.; Shoeybi, M.; Catanzaro, B.; Ping, W. NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models. *arXiv* **2024**. <https://doi.org/10.48550/arXiv.2405.17428>.
43. Cormack, G.V.; Clarke, C.L.A.; Buettcher, S. Reciprocal Rank Fusion Outperforms Condorcet and Individual Rank Learning Methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*; Association for Computing Machinery: New York, NY, USA, 2009; pp. 758–759. <https://doi.org/10.1145/1571941.1572114>.
44. Efron, B. Bootstrap Methods: Another Look at the Jackknife. *Ann. Stat.* **1979**, *7*, 1–26. <https://doi.org/10.1214/aos/1176344552>.

45. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv* **2019**. <https://doi.org/10.48550/arXiv.1907.11692>.
46. Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; et al. Qwen2.5 Technical Report. *arXiv* **2024**. <https://doi.org/10.48550/arXiv.2412.15115>.
47. Rajpurkar, P.; Jia, R.; Liang, P. Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2018; pp. 784–789. <https://doi.org/10.18653/v1/P18-2124>.
48. Kwon, W.; Li, Z.; Zhuang, S.; Sheng, Y.; Zheng, L.; Yu, C.H.; Gonzalez, J.E.; Zhang, H.; Stoica, I. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*; Association for Computing Machinery: New York, NY, USA, 2023; pp. 611–626. <https://doi.org/10.1145/3600006.3613165>.
49. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. BERTScore: Evaluating Text Generation with BERT. In *Proceedings of the International Conference on Learning Representations*, Virtual, 26 April–1 May 2020.
50. Kim, S.; Suk, J.; Longpre, S.; Lin, B.Y.; Shin, J.; Welleck, S.; Neubig, G.; Lee, M.; Lee, K.; Seo, M. Prometheus 2: An Open Source Language Model Specialized in Evaluating Other Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*; Al-Onaizan, Y., Bansal, M., Chen, Y.N., Eds.; Association for Computational Linguistics: Miami, FL, USA, 2024; pp. 4334–4353. <https://doi.org/10.18653/v1/2024.emnlp-main.248>.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.