

Article

Feasibility of Using Large Language Models for Structured Medication Extraction from Clinical Text: A Comparative Analysis of Zero-Shot and Few-Shot Paradigms

Evan Schulte ¹, Mohamed Abusharkh ^{1,*}, Kushal Dahal ² , Michael Klepser ²  and Minji Sohn ² ¹ Software Engineering, Ferris State University, Grand Rapids, MI 49503, USA; etschulte@gmail.com² Collaboration to Advance Pharmacy Enterprises (CAPE), College of Pharmacy, Ferris State University, Grand Rapids, MI 49503, USA; kushaldahal@ferris.edu (K.D.); michaelklepser@ferris.edu (M.K.); minjisohn@ferris.edu (M.S.)

* Correspondence: mohamedabusharkh@ferris.edu

Abstract

The digitization of healthcare has been accompanied by a rapid expansion of electronic health records (EHRs); however, a significant proportion of critical patient data, specifically medication regimens, remains entrapped within unstructured clinical narratives. The inability to seamlessly compute this data hinders advancements in pharmacovigilance, clinical decision support, and population health management. This study presents a comprehensive, rigorous evaluation of the feasibility of deploying Large Language Models (LLMs) to automate the extraction of structured dosage information (Dose, Daily Frequency, Duration) from outpatient antimicrobial clinical notes sourced from the Collaboration to Harmonize Antimicrobial Registry Measures (CHARM) registry. We scrutinized the performance of five distinct open-weight architectures, namely GPT-OSS:20B, Gemma 2:9B, Mistral 7B, Qwen3:14B and Llama 3.2, across both Zero-Shot and Retrieval Augmented Generation (RAG)-based Few-Shot prompting paradigms. Our analysis reveals a fundamental architectural trade-off: the reasoning-optimized GPT-OSS:20B dominates the zero-shot landscape ($F1 > 0.90$) by leveraging abstract schema understanding, whereas the instruction-tuned Gemma 2:9B excels in the few-shot setting ($F1 \sim 0.99$), effectively utilizing examples as guardrails to surpass larger models. Conversely, smaller models (Mistral, Llama) exhibit a prohibitive “hallucination barrier,” rendering them unsafe for unsupervised clinical application. Furthermore, we identify “Inconsistent Unit Handling” and “Complex Temporal Logic” as persistent failure modes that resist simple scaling laws. This report provides a definitive framework for selecting model architectures based on the availability of few-shot examples and highlights the necessity of dynamic RAG strategies to achieve production-grade reliability in medical informatics.



Academic Editor: Andrea Prati

Received: 27 January 2026

Revised: 21 February 2026

Accepted: 24 February 2026

Published: 27 February 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: large language models; medication extraction; natural language processing; zero-shot learning; few-shot learning; healthcare informatics; open-weight models

1. Introduction

The modern healthcare ecosystem is predicated on the availability of data. The global transition from paper-based charting to Electronic Health Records (EHRs) was driven by the promise of data accessibility; the idea that once clinical information was digitized, it would become instantly available for computation, analysis, and decision support. However,

reality has diverged significantly from this ideal. While the medium of storage has changed, the mode of communication has largely remained rooted in narrative prose. Physicians, nurses, and pharmacists continue to document the most critical aspects of patient care, particularly complex medication instructions, in unstructured free-text notes [1].

This reliance on unstructured text creates a “computability gap” [2]. A structured database field might indicate that a patient is on “Metoprolol,” but the critical context, that they are taking it at a dose of “50 mg twice daily, hold for heart rate less than 60”, is buried in the prescription sig or progress note. This seemingly minor distinction directly affects whether dosing instructions can be reliably retrieved and applied in clinical decision-making. Traditional extraction systems, typically rule-based or early statistical models, have historically failed to bridge this gap due to the immense linguistic variability of clinical shorthand (e.g., “QD,” “PO,” “PRN”) and the semantic ambiguity inherent in medical syntax [3,4]. This, in turn, leads to the necessity to manually extract relevant information from this unstructured string data. Preliminary workflow analyses conducted at our institution indicate that manually structuring 100 rows of raw medication data requires approximately 30 min of pharmacist labor. Given an average operational frequency of twice daily, this task introduces a continuous administrative burden of five hours per week, thereby diverting highly specialized clinical resources away from direct patient care and advanced clinical initiatives.

The emergence of Large Language Models (LLMs) based on the transformer architecture represents a potential paradigm shift [5]. Unlike their predecessors, LLMs possess a semantic understanding of language that allows for “text expansion”; the ability to infer that “PO” means “Orally” and “q.h.s” means “at bedtime” without explicit hard-coded rules [6]. However, the application of LLMs in the high-stakes domain of healthcare is not straightforward. Issues of hallucination (the generation of plausible but false information), inconsistent outputs, and data privacy remain significant barriers to adoption [7].

1.1. The Dosage Extraction Challenge

The specific task of extracting dosage information is uniquely challenging due to the density of numerical information and the subtle dependencies between fields [8]. As noted by the seminal work surrounding the 2009 i2b2 medication extraction challenge, extracting the name of a drug is a relatively solved problem, with F-measures historically exceeding 90% even with pre-LLM technologies. The difficulty lies in the attributes of Dose, Frequency, Route, and Duration [9].

- **Semantic Ambiguity:** A phrase like “500 mg” usually refers to the Strength of the tablet, whereas “2 tablets” refers to Dosage. However, a physician might write “Take 500 mg,” conflating the two. A robust system must normalize this to Dose: “500 mg” or Dose: “1 tablet”, Strength: “500 mg” depending on the schema, a nuance that frequently confounds automated systems.
- **Temporal Logic:** Instructions often contain complex temporal conditions, such as “Take 1 tablet daily for 3 days, then 1/2 tablet daily for 3 days.” Extracting a single “Frequency” or “Duration” from this requires a level of reasoning that goes beyond simple entity recognition; it requires understanding the sequence of events.
- **Boundary Errors:** In fast-paced clinical environments, typos are common. A string like “Lisinopril10mg” lacks the whitespace delimiters that traditional tokenizers rely on. LLMs, which utilize sub-word tokenization (e.g., Byte-Pair Encoding), are theoretically better equipped to handle these boundary errors, but their performance in practice requires rigorous validation.

1.2. Research Objectives

This research aims to move beyond anecdotal evidence of LLM capabilities and provide a rigorous, quantitative assessment of their feasibility for this specific clinical task. We structure our inquiry around four primary objectives:

1. **Benchmarking Model Architectures:** We aim to compare the performance of five distinct classes of open-weight models:
 - GPT-OSS:20B: A “Mixture-of-Experts” (MoE) model representing the upper bound of open-source reasoning capabilities [10].
 - Gemma 2:9B & Qwen3:14B: Dense, instruction-tuned models representing the “mid-sized” efficiency class [11,12].
 - Mistral 7B & Llama 3.2: Smaller models representing the lower bound of computational requirements, critical for edge deployment considerations [13,14].
2. **Evaluating Prompting Paradigms:** We seek to determine the optimal prompting strategy by comparing Zero-Shot (relying on internal knowledge) versus Few-Shot (relying on in-context learning) approaches. Specifically, we investigate whether few-shot examples act as a “guardrail” against hallucinations for smaller models, as suggested by Isaradech et al. [6].
3. **Analyzing Failure Modes:** Beyond aggregate F1-scores, we aim to categorize the specific types of errors (e.g., Hallucination vs. Unit Inconsistency) to understand why models fail.
4. **Assessing Stability and Scaling:** We analyze how model performance degrades or improves as the complexity of the dataset increases (from 50 to 150 samples).

1.3. Significance of Local and Open-Weight Models

A critical constraint in healthcare AI is privacy. The Health Insurance Portability and Accountability Act (HIPAA) in the US and the General Data Protection Regulation (GDPR) in Europe place strict limits on data sharing [15,16]. While proprietary models like GPT-4 may offer state-of-the-art performance, sending patient data to external API endpoints is often a non-starter for hospital IT security.

Therefore, this study explicitly focuses on Open-Weight Models (GPT-OSS, Gemma, Qwen, Mistral, Llama). These models can be downloaded and deployed locally, ensuring that patient data never leaves the local firewall environment. Validating that these local models can achieve parity with, or sufficiently approach, the performance of closed-source giants is a prerequisite for widespread clinical adoption. Our findings regarding the optimal size-to-performance ratio of the 9B parameter class have direct implications for resource-constrained hospitals that cannot afford the massive GPU clusters required for 70B+ parameter models [17].

2. Related Work

The pursuit of automated medication extraction is not new; it has evolved in lockstep with the broader field of Natural Language Processing (NLP). Understanding this historical trajectory is essential for appreciating where LLMs offer genuine innovation versus where they inherit the limitations of the past.

2.1. The Era of Rules and RegEx (Pre-2015)

In the early stages of clinical informatics, extraction systems were predominantly rule-based. Systems like MedEx and various implementations of Apache cTAKES relied on elaborate regular expressions and dictionary lookups [18,19].

- **Mechanism:** These systems worked by matching specific string patterns. A rule might define a dose as “a number followed by a unit (mg, g, mL).”
- **Limitations:** As highlighted by Uzuner et al. (2010) [9] in the context of the i2b2 challenge, these systems were incredibly brittle. They could easily extract “500 mg” but would fail on “two tablets,” which requires mapping the word “two” to the number “2.” More critically, they struggled with the “context” of the extraction. They could not distinguish between a medication the patient is taking versus a medication listed in the patient’s allergy list or family history.
- **Legacy:** The primary legacy of this era is the rigorous definition of the extraction schema (Drug, Dose, Route, Frequency, Duration) which remains the standard for evaluation today.

2.2. The Deep Learning Hybrid Era (2015–2022)

The introduction of Word Embeddings (Word2Vec, GloVe) and Recurrent Neural Networks (RNNs) ushered in a new era [20–22]. Models like BiLSTM-CRF (Bidirectional Long Short-Term Memory with Conditional Random Fields) became state of the art [23].

- **Advancement:** These models could use “context windows” to understand the words surrounding a target entity. Alfattni et al. (2021) [24] demonstrated that these architectures significantly reduced “boundary errors” (e.g., separating “Lisinopril” from “10 mg” even without spaces) because they operated on character-level or sub-word embeddings rather than just whitespace-delimited tokens.
- **The Annotation Bottleneck:** However, these supervised learning models required massive amounts of annotated training data. To train a BiLSTM-CRF to recognize “Duration,” one needed thousands of sentences where the duration was manually highlighted by human experts. This annotation process is expensive, time-consuming, and prone to inter-annotator disagreement. Furthermore, these models were “black boxes” regarding their failure modes; when they failed, it was difficult to understand why.
- **The Dosage vs. Strength Problem:** Chen et al. (2020) [25] noted that even these advanced models struggled with “semantic ambiguity.” They often confused “Dosage” (the amount per administration) with “Strength” (the manufacturing specification), leading to dangerous extraction errors where a patient might be recorded as taking “500 tablets” instead of “500 mg”.

2.3. The Generative LLM Era (2023–Present)

The Transformer architecture and the rise of Generative Pre-trained Transformers (GPT) fundamentally changed the approach from “extraction” to “generation.” Instead of highlighting text spans, the model is asked to generate a structured object (JSON) representing the information.

- **Zero-Shot and Few-Shot Learning:** The most significant capability of LLMs is “In-Context Learning.” As Isaradech et al. (2024) [6] explored, LLMs can perform tasks with zero examples (Zero-Shot) or a handful of examples (Few-Shot). This potentially eliminates the “Annotation Bottleneck” of the Deep Learning era. Their study was pivotal in identifying that Few-Shot prompting acts as a safety mechanism, reducing the rate of hallucinations, where the model invents medications or instructions not present in the source text.
- **Text Expansion:** Another key advantage identified in the literature is “Text Expansion.” Clinical notes are rife with abbreviations (“ASA”, “HCTZ”, “BID”). Isaradech et al. [6] noted that LLMs can automatically expand these into their full forms (“Aspirin”,

- “Hydrochlorothiazide”, “Twice Daily”) during the extraction process, a feature that previously required manually curated lookup tables.
- Temporal Reasoning: Despite these advances, Fornasiere et al. (2024) [26] highlight that LLMs still struggle with “Temporal Logic.” Extracting the duration of a medication course involves understanding complex timelines (“start X, then switch to Y after 3 days”). This remains the “frontier” of difficulty for current models, often resulting in “Frequency” and “Duration” being the lowest-performing fields.

2.4. Privacy-Preserving AI

Finally, the literature emphasizes the critical need for “Local” AI. Richter-Pechanski et al. (2025) [17] provided compelling evidence that local LLMs (open-weights running on local servers) can achieve performance parity with cloud-based models. This is a crucial validation for our study’s focus on models like Llama, Mistral, and Gemma, which are designed for such deployments. The concept of using a “Feedback LLM”, a second local model to grade the work of the first, emerges from this literature as a method to ensure quality without human intervention.

To summarize the trajectory of the field and define the specific scope of our study, Table 1 outlines the critical limitations of prior extraction methodologies, the resulting gaps in the current literature, and how our experimental design addresses them.

Table 1. Summary of Prior Limitations, Identified Gaps, and Proposed Solutions.

Limitations of Prior Work	The Resulting Knowledge Gap	How This Paper Addresses It
Rule-Based Systems: Rely on rigid regular expressions; fail on linguistic variability and semantic ambiguity (e.g., Dose vs. Strength).	Lack of systems that can dynamically interpret clinical shorthand and contextual phrasing without hard-coded rules.	Evaluates LLMs that utilize semantic “text expansion” to infer meanings (e.g., “PO” to “Orally”) without rigid templates.
Deep Learning (RNNs/CRFs): Suffer from requiring thousands of manually labeled examples to train effectively.	Need for extraction frameworks that can operate accurately with little to no task-specific training data.	Compares Zero-Shot (no examples) and Few-Shot (3 examples) prompting paradigms to bypass massive annotation requirements.
Generative LLMs (e.g., GPT-4): Proprietary, cloud-based models pose severe patient privacy and HIPAA compliance risks.	Unproven feasibility and safety of running privacy-preserving, local open-weight models for dosage extraction.	Benchmarks five specific local open-weight architectures (ranging from 3B to 20B parameters) to identify their viability for this task.

3. Methodology

This section details the experimental design, model architectures, prompting strategies, and evaluation metrics used to assess the feasibility of LLM-based dosage extraction.

3.1. Dataset and Task Definition

Data were obtained from the Collaboration to Harmonize Antimicrobial Registry Measures (CHARM) [27]. CHARM is a multi-institutional partnership established to track outpatient antibiotic prescribing patterns. The dataset comprises electronic health record (EHR) data aggregated from diverse healthcare systems across the United States. For the purpose of this study, we focused on extracting specific medication attributes, including Dose, Frequency, and Duration. This structured extraction task is consistent with standard clinical NLP benchmarks, such as the n2c2 [24].

To establish a high-quality reference standard for evaluation, a clean database consisting of over 1000 clinical records was curated from this larger dataset. These examples were manually reviewed and validated by specialized pharmacists, with a specific focus on ensuring the accuracy of prescription and dosage information. Prior to inclusion in this study, all patient identifiers were stripped from the data, restricting the dataset

solely to the raw medication and instruction text fields (Raw Drug, Raw Qty, Instructions). To strictly prevent data leakage, a record-level partitioning protocol was implemented; the de-identified clinical text records used in the 50, 100, and 150-row evaluation sets were mutually exclusive from, and strictly held out of, the 1000+ clean examples used in the RAG reference database. Table 2 provides five representative examples of the records in this validated reference database.

Ethical oversight and monitoring for the CHARM project are provided by the Ferris State University Institutional Review Board (IRB) [27]. All clinical records utilized in this study were de-identified and maintained behind a HIPAA-compliant firewall prior to extraction. This strict adherence to privacy protocols ensured that no Protected Health Information (PHI) was exposed to or processed by the local LLMs during our experiments.

Table 2. Clean Clinical Records (Validated Examples).

Input Data			Extracted (Clean) Data				
Raw Drug	Raw Qty	Instructions	Drug	Dose	Unit	Freq	Dur
NITROFURANTOIN MONOHYDRATE/ MACROCRYSTALS 100 MG CAPSULE	100 mg	Take 1 capsule by mouth 2 (two) times daily for 7 days. Finish entire dose...	Nitrofurantoin	100	mg	2	7
CEFDINIR 300 MG CAPSULE	300 mg	Take 1 capsule by mouth 2 (two) times daily for 10 days, Starting Tue 1/14/2025, Until Fri 1/24/2025, Normal	Cefdinir	300	mg	2	10
AMOXICILLIN 500 MG CAPSULE	500 mg	Take 2 capsules by mouth 2 (two) times daily for 10 days, Starting Thu 12/12/2024, Until Sun 12/22/2024, Normal	Amoxicillin	500	mg	2	10
DOXYCYCLINE HYCLATE 100 MG TABLET	100 mg	Take 1 tablet by mouth 2 (two) times daily for 7 days, Starting Fri 8/23/2024, Until Fri 8/30/2024, Normal	Doxycycline	100	mg	2	7
CEPHALEXIN 500 MG CAPSULE	500 mg	Take 1 capsule by mouth 4 (four) times daily, Starting Wed 12/4/2024, Normal	Cephalexin	500	mg	4	10

The models were tasked with extracting information into a strict numeric format. This requirement tests not only the model’s language understanding but also its ability to generate correct numeric values based on that understanding and to adhere to a specific schema.

- Target Fields: Dose (numeric + unit), Frequency (standardized string), Duration (standardized string).

Annotation and Markup Protocol

To establish a rigorous ground truth for evaluation, the reference dataset was annotated by specialized pharmacists. The annotation process followed a strict consensus-based protocol to resolve ambiguities inherent in clinical text, particularly regarding the distinction between unit strength and total dosage.

- Unit Strength vs. Total Dose: A critical ambiguity in medication extraction is distinguishing between the strength of the formulation (e.g., “500 mg”) and the total amount ingested (e.g., “Take 2 tablets”). For this study, the Dose field was strictly defined as the numeric strength of the prescribed unit. For example, an instruction to “Take two 500 mg capsules” is annotated as Dose: 500, not 1000. This decision was made to isolate the model’s extraction capability from its arithmetic reasoning capability.
- Frequency Normalization: Textual frequency instructions were mapped to a standardized daily count integer. Standard Latin abbreviations (“BID”, “TID”) and natural language phrases (“twice a day”, “every 12 h”) were normalized to their numeric

equivalents (e.g., Frequency: 2). “PRN” (as needed) instructions without a specific cap were annotated as 0 to distinguish them from scheduled dosing.

- Duration Standardization: Duration was strictly defined as the total number of days of therapy. Instructions specifying a date range (e.g., “1 January to 10 January”) were converted to the integer difference (e.g., Duration: 10).

Disagreements between annotators were resolved through discussion until 100% consensus was reached on the final reference set.

3.2. Model Architectures

We selected five open-weight models that represent different points on the trade-off curve between Size, Reasoning Capability, and Instruction Following.

1. GPT-OSS:20B

- Architecture: This model utilizes Mixture-of-Experts (MoE) architecture. Unlike a dense model where every parameter is used for every token, an MoE model routes tokens to specific “experts” (sub-networks).
- Specs: It has 20 billion total parameters but only activates approximately 3.6 billion parameters per token during inference.
- Rationale: This architecture allows for high “reasoning” capability (due to the large total parameter count storing vast knowledge) while maintaining fast inference speeds (due to the low active parameter count). It is explicitly optimized for reasoning and function calling, making it the theoretical heavyweight for zero-shot tasks.

2. Gemma 2:9B:

- Architecture: Developed by Google, Gemma 2 is a dense model (all parameters active).
- Specs: Nine billion parameters. It utilizes knowledge distillation during training, where it learns from a much larger model, effectively compressing high-level capabilities into a smaller frame.
- Rationale: As an “Instruct” version, it has been fine-tuned with Reinforcement Learning from Human Feedback (RLHF) specifically to follow user instructions. We hypothesized this would make it superior at adhering to the strict formatting constraints of the Few-Shot paradigm.

3. Qwen3:14B:

- Architecture: This model is a dense Transformer from the Qwen3 family.
- Specs: It has 14.8 billion parameters.
- Rationale: This model serves as a bridge between the lightweight (7B–9B) class and the larger reasoning models. It tests whether pure parameter scaling in a dense setup yields proportional gains in a zero-shot reasoning without the complexity of an MoE architecture.

4. Mistral 7B:

- Architecture: This model utilizes a dense Transformer model.
- Specs: It has 7 billion parameters. Known for its high performance relative to its small size and its use of “Sliding Window Attention” to handle longer contexts.
- Rationale: Mistral 7B serves as the standard baseline for “competent” small open-source models. It represents the minimum viable size for many NLP tasks.

5. Llama 3.2:

- Architecture: A highly optimized dense model designed for edge devices (mobile/local deployment).

- Specs: It has 3 billion parameters. An instruction tuned text-only model that is optimized for on device use cases like summarization, instruction following, and rewriting tasks.
- Rationale: Including this model tests the lower bound of feasibility. If a 3B model could perform this task, it would enable extraction directly on clinicians' tablets or lightweight hospital terminals, radically changing the deployment economics.

Experimental Environment and Inference Configuration

To ensure reproducibility and standardize the generation environment, all models were deployed locally using the Ollama framework [28]. This approach natively handles model quantization, with models running in default 4-bit integer quantization to allow for efficient local execution.

Inference parameters were strictly controlled to balance deterministic data extraction with the semantic flexibility required for natural language processing. Across all models, the temperature was set to 0.3. For the GPT-OSS:20B architecture, the context window was explicitly restricted to 2048 tokens, and the maximum token generation limit was capped at 200 to optimize memory utilization. For all other evaluated models, the limit was set to 500 tokens.

The complete hardware and software specifications used for these experiments are detailed in Table 3.

Table 3. Experimental Hardware and Software Configuration.

Component	Specification
Processor (CPU)	13th Gen Intel Core i9-13900K
System RAM	64 GB, 4 × 16 GB, DDR5, 4000 MHz, ECC
Inference Engine	Ollama (v0.1.20) with llama.cpp backend
Quantization	4-bit (Q4_K_M)
Operating System	Windows 11 Pro

3.3. Prompting Strategies

We compared two distinct prompting paradigms to understand how in-context learning influences extraction performance.

1. Zero-Shot Prompting [29]

Definition 1. *The model is provided with the clinical text and the schema definition (e.g., “Extract dosage info into numeric format...”) but is given no examples of complete extractions.*

- Mechanism: The model must rely entirely on its pre-trained internal knowledge and its ability to abstractly understand the instructions. It has no reference point for how to handle ambiguities.

Hypothesis 1. *Larger, reasoning-focused models (GPT-OSS) should excel here due to superior abstract comprehension.*

2. Few-Shot Prompting [30]

Definition 2. *The prompt (see Appendix A) includes the instruction, the text, and 3 examples of valid input–output pairs (e.g., “Input: ‘Take 500 mg daily’ → Output: Dose: ‘500’, Freq: 1”).*

- Mechanism:** This utilizes in-context learning. The examples serve as a template (see Appendix A), implicitly teaching the model not just what to extract, but how to format it (e.g., standardizing “QD” to “Daily”, or stripping units if the examples show unit-less numbers). The examples are chosen using RAG. The application partitions the raw input data into discrete, ten-row batches. This textual data is converted into vector embeddings to facilitate a similarity search within a reference database containing over 1000 validated clean examples. For each row in each batch, the three most semantically similar reference records are retrieved. The LLM is then supplied with the task prompt, the structural template, the raw input batch, and the retrieved reference data. This iterative process continues until the entire dataset is evaluated, culminating in a unified, structured tabular output. Figure 1 illustrates this process.

Hypothesis 2. This should act as a “guardrail,” boosting the performance of smaller models (Llama, Mistral) and improving the formatting consistency of all models.

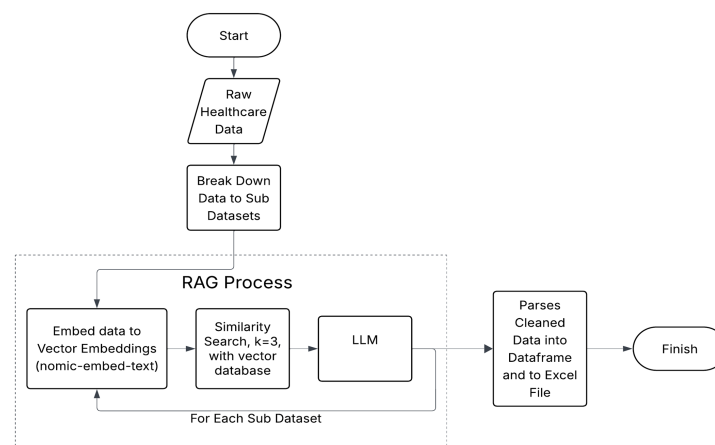


Figure 1. RAG Process.

3.4. Evaluation Metrics

To ensure a rigorous quantitative assessment, we utilized standard classification metrics [31]. The performance of the models in extracting clinical entities was evaluated using Precision, Recall, and the F1-score, which serves as the established framework for benchmarking Large Language Models in clinical informatics [32].

- F1-Score:** The harmonic mean of Precision and Recall.

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (1)$$

- This metric penalizes both “missed” information (False Negatives) and “hallucinated/incorrect” information (False Positives).

To confirm that performance differences were not due to random chance, we employed the Friedman Test, a non-parametric statistical test used to detect differences in treatments across multiple test attempts.

- Values and p -values were calculated.
- Post-hoc Analysis: Pairwise comparisons (e.g., Nemenyi test equivalent) were used to identify exactly which models were significantly different from each other.

4. Results

To provide a comprehensive quantitative baseline across all evaluated architectures, Table 4 details the granular performance metrics for the 150-row data split. Rather than relying solely on aggregate scores, this table breaks down performance by specific extraction fields (Frequency, Dose, Duration) across both Zero-Shot and Few-Shot paradigms.

Furthermore, to address the clinical safety of these models, the results are presented using a standard machine learning confusion matrix framework. True Positives (TP) indicate correct extractions, False Positives (FP) quantify the rate of clinically unsafe ‘hallucinations’ (over-predictions), and False Negatives (FN) represent missed or improperly formatted data. This field-level breakdown isolates exactly where the models succeed and where they experience catastrophic failure.

Table 4. Detailed Performance Metrics and Confusion Matrix by Field, Model, and Prompting Paradigm (150-Row Split).

Model	Paradigm	Field	TP	FP	FN	Precision	Recall	F1
GPT-OSS:20B	Zero-Shot	Frequency	143	0	7	1.00	0.95	0.98
		Dose	101	12	37	0.89	0.73	0.80
		Duration	148	0	2	1.00	0.99	0.99
	Few-Shot	Frequency	107	41	2	0.72	0.98	0.83
		Dose	72	65	13	0.53	0.85	0.65
		Duration	97	49	4	0.66	0.96	0.79
Gemma:9B	Zero-Shot	Frequency	63	32	55	0.66	0.53	0.59
		Dose	42	73	35	0.37	0.55	0.44
		Duration	55	35	60	0.61	0.48	0.54
	Few-Shot	Frequency	100	48	2	0.68	0.98	0.80
		Dose	144	6	0	0.96	1.00	0.98
		Duration	124	25	1	0.83	0.99	0.91
Qwen3:14B	Zero-Shot	Frequency	127	20	3	0.86	0.98	0.92
		Dose	56	67	27	0.46	0.67	0.54
		Duration	118	24	8	0.83	0.94	0.88
	Few-Shot	Frequency	79	70	1	0.53	0.99	0.69
		Dose	77	73	0	0.51	1.00	0.68
		Duration	82	67	1	0.55	0.99	0.71
Mistral:7B	Zero-Shot	Frequency	4	53	93	0.07	0.04	0.05
		Dose	5	52	93	0.09	0.05	0.06
		Duration	8	22	120	0.27	0.06	0.10
	Few-Shot	Frequency	35	62	53	0.36	0.40	0.38
		Dose	50	48	52	0.51	0.49	0.50
		Duration	54	37	59	0.59	0.48	0.53
Llama3.2:3B	Zero-Shot	Frequency	26	75	49	0.26	0.35	0.30
		Dose	31	71	48	0.30	0.39	0.34
		Duration	44	55	51	0.44	0.46	0.45
	Few-Shot	Frequency	50	88	12	0.36	0.81	0.50
		Dose	34	103	13	0.25	0.72	0.37
		Duration	25	114	11	0.18	0.69	0.29

Note: TP = True Positives, FP = False Positives (Hallucinations), FN = False Negatives.

4.1. Zero-Shot Performance

The Zero-Shot experiments serve as a test of the models’ “raw intelligence”, that is, their ability to understand a task description and execute it without mimicking examples. The results revealed a stark hierarchy dictated by model architecture and size.

4.1.1. The Dominance of Reasoning Models

As illustrated in Figure 2, GPT-OSS:20B emerged as the undisputed leader in the zero-shot setting.

- **Performance:** It consistently achieved F1-scores exceeding 0.90 across all extraction fields (Frequency, Dose, Duration).
- **Stability:** Crucially, its performance remained stable across all data splits (50, 100, 150 rows), indicating that its reasoning is robust and generalizable, not brittle.

The MoE architecture of GPT-OSS allows it to access a broader base of knowledge (20B parameters) while maintaining efficient processing. Its reasoning training focus enables it to handle the abstract requirements of the prompt, understanding implicit schema rules, better than the smaller dense models.

4.1.2. The “Zero-Shot Wall” for Small Models

In sharp contrast, the smaller models (Mistral 7B and Llama 3.2) hit what we term the “Zero-Shot Wall.”

- **Performance Collapse:** Their F1-scores frequently dropped below 0.50. Mistral 7B struggled severely, posting an F1-score of 0.12 in some splits.
- **Instruction Following Failure:** The primary cause of failure was not necessarily a lack of medical knowledge, but a failure in strict instruction following. Without examples to guide the output format, these models often generated conversational text (e.g., “Here is the data you asked for...”) instead of valid numeric format. This confirms that sub-10B parameter models generally lack the inherent robustness to handle complex structured extraction tasks without the “priming” of few-shot examples.

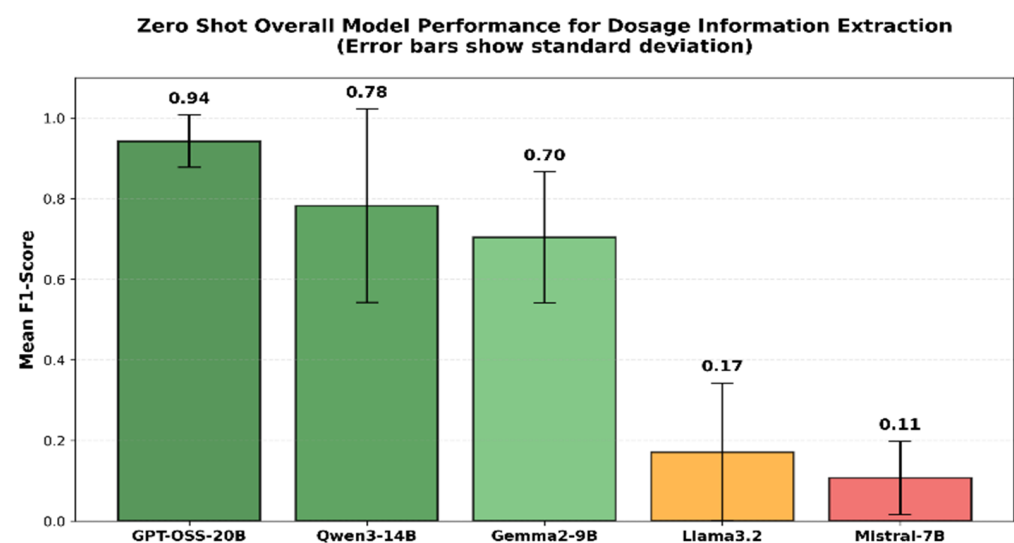


Figure 2. Zero Shot Overall Model Performance.

4.1.3. Field-Specific Difficulty Analysis

The zero-shot data validated the historical “difficulty hierarchy” of clinical extraction fields (Table 5).

The Dose field was the most fragile because of the “Unit Problem.” In a zero-shot setting, the prompt might ask for Dose, but the model is unsure whether to return the string “500 mg” or the integer 500. GPT-OSS and Gemma often split on this, leading to lower exact-match scores despite semantically correct extractions. Figure 3 shows that, of all fields extracted, dose performed the worst across all models.

Table 5. Zero-Shot Field Difficulty.

Field	Difficulty	Primary Challenge	Observed Failure Mode	Example Error
Duration	Low	Distinct linguistic markers (e.g., “for 7 days”) are easy to parse.	Boundary Errors	Missing “days” unit
Frequency	Medium	Distinguishing “Frequency” (times/day) from “Dosage” (pills/intake).	Hallucination	Inferring “Daily” from “Take 1 tablet”
Dose	High	Normalizing variable units (“500”, “500 mg”, “0.5 g”).	Unit Inconsistency	Returning “500 mg” when schema requires 500

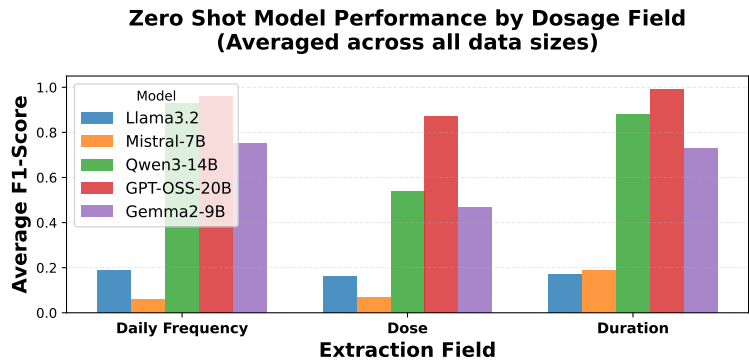


Figure 3. Zero Shot Model Performance by Field.

4.1.4. Error Analysis

The granular error analysis visualized in Figure 4 reveals distinct failure modes that correlate with model size.

- Hallucination: Llama 3.2 and Mistral 7B suffered heavily from “Over-Prediction.” They would frequently infer a daily frequency when the text merely said “Take as needed” or “Take 1 tablet.” This tendency to fill in blanks with probable (but factually absent) data makes them dangerous for zero-shot clinical use.
- Inconsistent Unit Handling: Interestingly, the larger models (GPT-OSS, Gemma) suffered less from hallucination and more from “Inconsistent Unit Handling.” They would correctly identify the value “500” but fail to format it as requested. This is a competence error; the model found the right data but failed the formatting check.
- Statistical Significance: The Friedman Test ($\chi^2 = 32.94, p < 0.001$) confirmed that the superiority of GPT-OSS in this setting was statistically significant. Pairwise comparisons showed that Llama 3.2 and Mistral 7B were significantly different (worse) than GPT-OSS ($p < 0.001$).

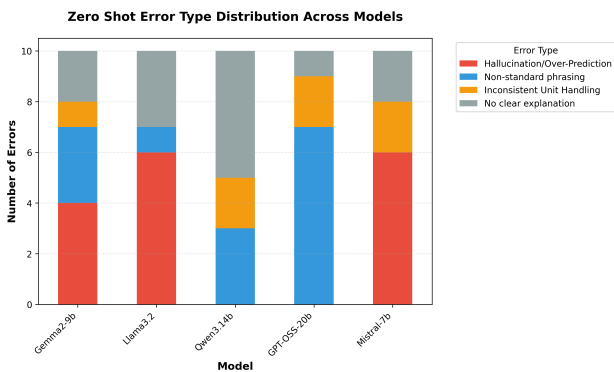


Figure 4. Zero Shot Error Type Distribution.

4.2. Few-Shot Performance

The introduction of 3 few-shot examples produced a dramatic inversion of the performance hierarchy, as illustrated in Figure 5. This “Few-Shot Reversal” is the most critical finding of our study, challenging the assumption that bigger is always better.

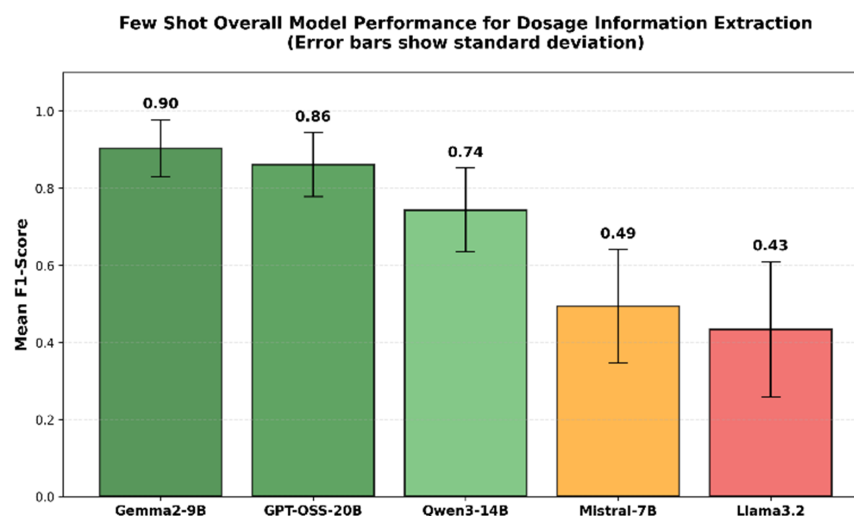


Figure 5. Few Shot Overall Model Performance.

4.2.1. The Rise of Gemma 2:9B

In the few-shot setting, Gemma 2:9B outperformed the significantly larger GPT-OSS:20B.

- **Peak Performance:** Gemma achieved near-perfect F1 scores on the 50-row split (Figure 6): 0.99 for Dose and 0.98 for Duration.
- **Pattern Matching Efficacy:** Error analysis indicates that the performance differential between the models is largely attributable to the instruction-tuned architecture’s robust pattern-matching capabilities. By providing a strict structural template via few-shot prompting, the Gemma 2 model adhered rigidly to the prescribed format, effectively reducing the extraction task to a pattern-completion mechanism rather than relying on generalized reasoning.
- **Observed Failure Modes:** The primary observed failure mode occurred when the input clinical text exhibited substantial syntactic deviation from the provided few-shot examples. These findings suggest that for standardized extraction protocols, strict adherence to structural templates yields higher fidelity outputs than deep semantic comprehension.

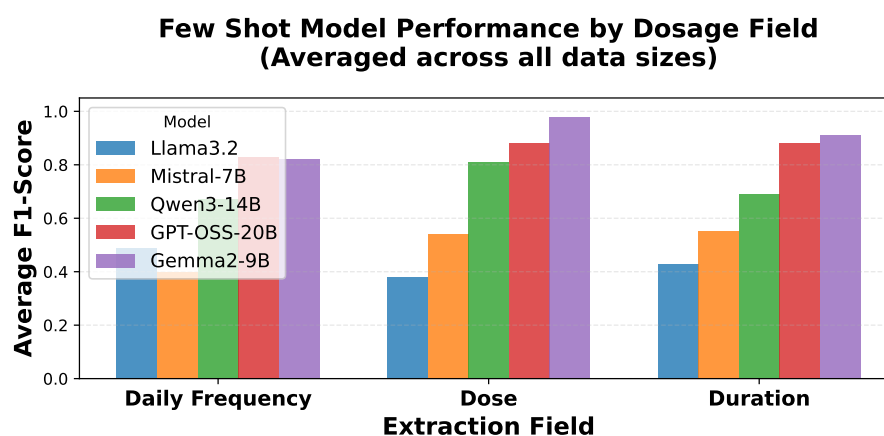


Figure 6. Few Shot Model Performance by Field.

4.2.2. The Regression of GPT-OSS:20B

Counter-intuitively, the reasoning powerhouse GPT-OSS saw its relative performance degrade in the few-shot setting compared to Gemma. Its F1 score for Dose dropped to 0.74 in the 100-row split.

- **The Descriptive Bias Problem:** The error analysis reveals that GPT-OSS had the highest count of inconsistent unit handling errors (4 cases) in the few-shot setting.
- **Interpretation:** While the few-shot examples implicitly taught the rule, strip the units, GPT-OSS likely over-generated beyond the strict extraction parameters. Its internal training (which likely prioritizes completeness and accuracy) might have overridden the few-shot context, leading it to include “mg” because “500 mg” is scientifically more accurate than “500.” This resistance to implicit formatting constraints makes the reasoning model less controllable for strict data entry.

4.2.3. The Persistence of Hallucination in Small Models

Few-shot examples improved the syntax of Llama 3.2 and Mistral 7B; they produced valid numeric outputs more often but failed to fix their semantics.

- **Hallucinations:** These models continued to suffer from hallucination/over-prediction. For example, if the text did not specify a duration, Llama 3.2 would often guess “1 day” or “1 week” rather than outputting 0 (as defined in the prompt).
- **Implication:** This suggests that few-shot examples act as syntactic guardrails (fixing the format) but not semantic guardrails (fixing the truthfulness) for models with limited capacity. A 3B parameter model simply lacks the world model to understand that “not mentioned” means “do not extract.”

4.3. Stability and Dataset Size

A crucial finding for real-world deployment is the degradation of performance as the dataset size increases. As the evaluation set grew from 50 to 100 rows, Gemma’s performance dropped from ≈ 0.99 to ≈ 0.80 , and Llama 3.2 collapsed from ≈ 0.53 to ≈ 0.25 . The Friedman Test ($\chi^2 = 32.18, p < 0.001$) confirmed that Gemma 2:9B’s lead was statistically significant.

5. Discussion

The divergence between the Zero-Shot and Few-Shot results offers profound insights into the mechanics of LLMs and their suitability for healthcare.

5.1. The “Alignment Tax” on Reasoning Models

The observed results indicate an “alignment tax” inherent to reasoning-optimized architectures such as GPT-OSS. These models are heavily fine-tuned to generate comprehensive and descriptive responses. However, this bias toward descriptive elaboration (e.g., appending units of measurement for clarity) directly conflicts with the rigid syntactic constraints required for structured database schemas. Conversely, instruction-tuned architectures like Gemma 2 exhibit superior adaptability to structural constraints when provided with few-shot examples. In the specific context of structured data extraction, strict syntactical compliance proves significantly more advantageous than complex generative reasoning.

5.2. Privacy and the “Local” Advantage

The fact that Gemma 2:9B, a model small enough to run on a single consumer GPU (e.g., NVIDIA RTX 3090/4090 or A10G), outperformed the 20B model is a massive win for privacy. Facilities working with healthcare data can deploy Gemma 2 on premises. No patient data needs to be sent to OpenAI or Google Cloud APIs. This resolves the

primary compliance barrier (HIPAA/GDPR) identified in the literature. Running a 9B model is significantly cheaper (energy and hardware) than a 20B or 70B model, making high-throughput processing of millions of historical notes economically viable.

5.3. Clinical Safety and Harm Analysis

While F1-scores provide a quantitative measure of extraction accuracy, they do not fully capture the clinical risk associated with model errors. To address the concern regarding “unacceptable safety risks,” we analyzed the shift in error distribution between paradigms (comparing Figures 4 and 7).

- **Critical Errors (High Harm Potential):** We define “Hallucinations” as critical errors where the model generates a dosage, frequency, or duration that is factually absent from the source text (e.g., inferring “Daily” from “PRN”). As illustrated in Figure 4, zero-shot prompting yields high hallucination rates in small models. Crucially, Figure 7 demonstrates that while few-shot prompting reduces these rates, it does not eliminate them for the 3B/7B class, confirming they remain unsafe for unsupervised use.
- **Formatting Errors (Low Harm Potential):** We classify “Unit Inconsistencies” and “Invalid Format” errors as low-harm. For example, extracting “500 mg” when the schema requires the integer “500” preserves the clinical intent but fails the strict formatting check. Larger models like GPT-OSS:20B primarily suffered from these errors. While they reduce automation efficiency, they do not pose a direct safety risk to the patient and can be mitigated with simple post-processing rules.

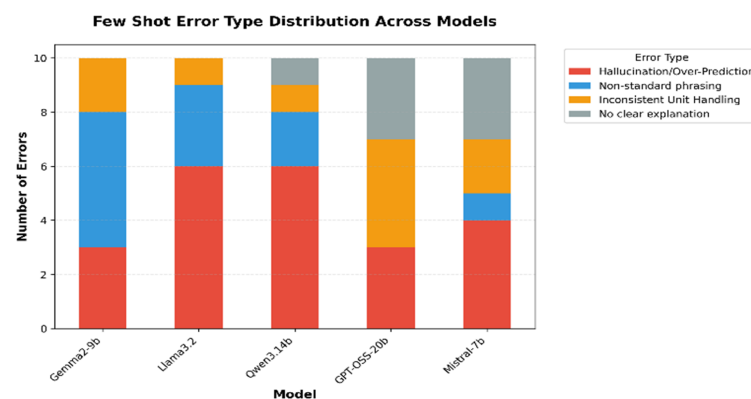


Figure 7. Few-Shot Error Type Distribution.

This distinction justifies our conclusion that while 9B+ models are feasible for “human-in-the-loop” workflows, sub-7B models pose a direct safety risk due to their intrinsic hallucination rates.

5.4. Limitations

While this study demonstrates the feasibility of local LLMs for clinical extraction, several limitations must be acknowledged. First, the dataset was derived exclusively from outpatient antimicrobial registries. Outpatient prescriptions tend to follow more predictable patterns than inpatient or ICU orders, which often involve complex titration instructions and frequent regimen changes. Consequently, our performance metrics may not fully generalize to high-acuity inpatient settings.

Second, our experimental design prioritized privacy-preserving open-weight models. We explicitly excluded proprietary state-of-the-art models (e.g., GPT-4o, Claude 3.5 Sonnet) to simulate a strict local-deployment environment. It is possible that these larger, closed-

source models possess reasoning capabilities that could overcome the “Zero-Shot Wall” we observed, though they remain unsuitable for privacy-constrained environments.

Finally, we relied on in-context learning and RAG rather than fine-tuning the models. While we demonstrated that RAG is effective, fine-tuning the smaller models (Llama 3.2, Mistral) specifically on the CHARM dataset might yield better results than prompting alone, albeit at the cost of higher training overhead. Future work should investigate whether fine-tuning can mitigate the hallucination rates observed in the sub-7B parameter class.

6. Conclusions

This study provides a definitive feasibility assessment for using LLMs to extract medication dosage information. We conclude that LLMs are highly feasible, but a “one-size-fits-all” approach is fundamentally flawed.

- **Paradigm Selection:** Few-Shot Prompting is mandatory. Zero-Shot approaches, while promising with large reasoning models, lack the formatting consistency required for database integration.
- **Model Selection:** Of the models tested, Gemma2:9B performed the best overall. It balances high performance ($F1 \sim 0.99$), manageable size for local deployment, and superior instruction-following capabilities compared to larger reasoning models.
- **Safety:** Small models (sub-7B parameters like Mistral/Llama 3.2) currently pose an unacceptable safety risk due to high hallucination rates and should not be deployed for clinical extraction without massive guardrails.

We recommend utilizing few-shot prompting with Google’s Gemma2:9B model. Additionally, we do not recommend utilizing sub 7B parameter models for clinical extraction as they pose a safety risk due to high hallucination rates. By adopting these architectural and procedural recommendations, the structuring of healthcare data can be robustly automated, enabling clinical personnel to allocate their expertise toward tasks of greater clinical complexity and consequence.

Author Contributions: Conceptualization, E.S., M.A., K.D. and M.S.; Methodology, E.S. and K.D.; Software, E.S.; Validation, E.S.; Formal analysis, E.S. and K.D.; Investigation, E.S.; Resources, M.A., K.D., M.K. and M.S.; Data curation, E.S., K.D., M.K. and M.S.; Writing—original draft, E.S.; Writing—review & editing, E.S., M.A. and M.S.; Supervision, M.A., K.D. and M.S.; Project administration, E.S.; Funding acquisition, M.K. and M.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors thank the Ferris State University College of Pharmacy and the Collaboration to Harmonize Antimicrobial Registry Measures (CHARM) for providing the computational resources and data necessary to conduct this research. Additionally, the authors thank the Illinois Department of Public Health (IDPH) for sponsoring this effort.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Prompts Used

Appendix A.1. Zero-Shot Template

“You are an expert pharmaceutical data analyst specializing in extracting dosage information from patient instructions. Here is the raw data to process: {data} Task: {prompt} Output format: {format} Important Instructions:

- Extract daily frequency (how many times per day), dose amount in mg, and duration in days
- Analyze the patient instructions carefully to determine the correct values
- Look for patterns like “twice daily”, “3 times per day”, dosage amounts (e.g., “500 mg”), and duration (e.g., “for 7 days”)
- If uncertain about any value, use 0
- Focus on numerical extraction only”

Appendix A.2. Few-Shot Template

“You are an expert pharmaceutical data analyst specializing in extracting dosage information from patient instructions. Here is the raw data to process: {data} Here are relevant reference examples from similar cases: {reference} Task: {prompt} Output format: {format} Important Instructions:

- Use the reference examples to understand how similar raw data was processed
- Extract daily frequency (how many times per day), dose amount in mg, and duration in days
- Look for patterns in the reference examples that match your input data
- If there is an exact match in the reference examples use that, otherwise do your best to determine what the correct values are based on the other examples in the reference
- If uncertain about any value, use 0
- Focus on numerical extraction only”

References

1. Carayon, P.; Hoonakker, P.; Cartmill, R.S.; Scanner, L.; Janes, J.; Walker, J.M. Clinician perceptions on the use of free-text communication orders. *Appl. Clin. Inform.* **2021**, *12*, 548–557.
2. Johnson, S.B.; Bakken, S.; Dine, D.; Hyun, S.; Mendonça, E.; Morrison, F.; Bright, T.; Van Vleck, T.; Wrenn, J.; Stetson, P. An electronic health record based on structured narrative. *J. Am. Med. Inform. Assoc.* **2008**, *15*, 54–64. [CrossRef]
3. Ge, Y.; Wang, J.; Lai, T.; Zhang, Y.; Li, F. Balancing natural language processing accuracy and normalisation in extracting medical insights. *arXiv* **2025**, arXiv:2511.15778. [CrossRef]
4. Wu, Y.; Denny, J.C.; Rosenbloom, S.T.; Miller, R.A.; Giuse, D.A.; Xu, H. A comparative study of current clinical natural language processing systems on handling abbreviations and acronyms. *AMIA Annu. Symp. Proc.* **2012**, *2012*, 997–1003.
5. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A survey of large language models. *arXiv* **2023**, arXiv:2303.18223. [PubMed]
6. Isaradech, N.; Riedel, A.; Sirikul, W.; Kreuzthaler, M.; Schulz, S. Zero- and Few-shot Named Entity Recognition and Text Expansion in Medication Prescriptions using ChatGPT. *arXiv* **2024**, arXiv:2409.17683. [CrossRef]
7. Bélisle-Pipon, J.-C. Why we need to be careful with LLMs in medicine. *Front. Med.* **2024**, *11*, 1495582. [CrossRef] [PubMed]
8. Landolsi, M.Y.; Hlaoua, L.; Ben Romdhane, L. Information extraction from electronic medical documents: State of the art and future research directions. *Knowl. Inf. Syst.* **2022**, *65*, 463–516. [CrossRef]
9. Uzuner, Ö.; Solti, I.; Xia, F. Extracting medication information from clinical text. *J. Am. Med. Inform. Assoc.* **2010**, *17*, 514–518. [CrossRef] [PubMed]
10. OpenAI. Introducing GPT-OSS: The Open Source Successor to GPT-2. OpenAI 2024. Available online: <https://openai.com/index/introducing-gpt-oss/> (accessed on 29 January 2026).
11. Google. Gemma 2: Improving open language models at a practical size [Model card]. *arXiv* **2024**, arXiv:2408.00118.
12. Alibaba Cloud. Qwen3-14B: A Versatile Large Language Model [Model Card]. Hugging Face 2025. Available online: <https://huggingface.co/Qwen/Qwen3-14B> (accessed on 29 January 2026).

13. Meta. Llama 3.2: Lightweight Models for Edge Devices [Model Card]. Hugging Face 2024. Available online: <https://huggingface.co/meta-llama/Llama-3.2-3B> (accessed on 29 January 2026).
14. Jiang, A.Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D.S.; Casas, D.d.l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. Mistral 7B [Model Announcement]. Mistral AI 2023. Available online: <https://mistral.ai/news/announcing-mistral-7b/> (accessed on 29 January 2026).
15. Assistant Secretary for Planning and Evaluation (ASPE). *Health Insurance Portability and Accountability Act of 1996*; Pub. L. No. 104–191. 110 Stat. 1936; ASPE: Washington, DC, USA, 1996.
16. European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). *Off. J. Eur. Union* **2016**, L119, 1–88.
17. Richter-Pechanski, P.; Seiferling, M.; Kiriakou, C.; Schwab, D.M.; Geis, N.A.; Dieterich, C.; Frank, A. Medication information extraction using local large language models. *J. Biomed. Inform.* **2025**, *169*, 104898. [\[CrossRef\]](#)
18. Xu, H.; Stenner, S.P.; Doan, S.; Johnson, K.B.; Waitman, L.R.; Denny, J.C. MedEx: A medication information extraction system for clinical narratives. *J. Am. Med. Inform. Assoc.* **2010**, *17*, 19–24. [\[CrossRef\]](#)
19. Savova, G.K.; Masanz, J.J.; Ogren, P.V.; Zheng, J.; Sohn, S.; Kipper-Schuler, K.C.; Chute, C.G. Mayo clinical Text Analysis and Knowledge Extraction System (cTAKES): Architecture, component evaluation and applications. *J. Am. Med. Inform. Assoc.* **2010**, *17*, 507–513. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. *arXiv* **2013**, arXiv:1301.3781. [\[CrossRef\]](#)
21. Pennington, J.; Socher, R.; Manning, C. GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014.
22. Schmidt, R.M. Recurrent neural networks (RNNs): A gentle introduction and overview. *arXiv* **2019**, arXiv:1912.05911. [\[CrossRef\]](#)
23. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991. [\[CrossRef\]](#)
24. Alfattni, G.; Belousov, M.; Peek, N.; Nenadic, G. Extracting Drug Names and Associated Attributes From Discharge Summaries: Text Mining Study. *JMIR Med. Inform.* **2021**, *9*, e24678. [\[CrossRef\]](#)
25. Chen, L.; Gu, Y.; Ji, X.; Sun, Z.; Li, H.; Gao, Y.; Huang, Y. Extracting medications and associated adverse drug events using a natural language processing system combining knowledge base and deep learning. *J. Am. Med. Inform. Assoc.* **2020**, *27*, 56–64. [\[CrossRef\]](#)
26. Fornasiere, R.; Brunello, N.; Scotti, V.; Carman, M. Medical Information Extraction with Large Language Models. In Proceedings of the 7th International Conference on Natural Language and Speech Processing (ICNLSP 2024), Trento, Italy, 19–20 October 2024.
27. Ferris State University. CHARM Project. Available online: <https://www.ferris.edu/cape/charm.htm> (accessed on 13 February 2026).
28. Ollama. Ollama: Get up and Running with Large Language Models. GitHub Repository. 2024. Available online: <https://github.com/ollama/ollama> (accessed on 13 February 2026).
29. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 22199–22213.
30. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
31. Sokolova, M.; Lapalme, G. A systematic analysis of performance measures for classification tasks. *Inf. Process. Manag.* **2009**, *45*, 427–437. [\[CrossRef\]](#)
32. Villanueva-Miranda, I.; Gu, Z.; Yang, D.; Nezafati, K.; Huang, J.; Ruan, P.; Zhan, X.; Xiao, G.; Xie, Y. ClinBench: A Standardized Multi-Domain Framework for Evaluating Large Language Models in Clinical Information Extraction. In Proceedings of the NeurIPS 2025, San Diego, CA, USA, 30 November–7 December 2025.

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.