Symphony for Medical Coding: A Next-Generation Agentic System for Scalable and Explainable Medical Coding

Joakim Edin[†] **Andreas Motzfeldt**[†] **Simon Flachs** **Lars Maaløe***

Corti

Abstract

Medical coding translates free-text clinical documentation into standardized codes drawn from classification systems that contain tens of thousands of entries and are updated annually. It is central to billing, clinical research, and quality reporting, yet remains largely manual, slow, and error-prone. Existing automated approaches learn to predict a fixed set of codes from labeled data, thereby preventing adaptation to new codes or different coding systems without retraining on different data. They also provide no explanation for their predictions, limiting trust in safety-critical settings. We introduce Symphony for Medical Coding, a system that approaches the task the way expert human coders do: by reasoning over the clinical narrative with direct access to the coding guidelines. This design allows Symphony to operate across any coding system and to provide span-level evidence linking each predicted code to the text that supports it. We evaluate on two public benchmarks and three real-world datasets spanning inpatient, outpatient, emergency, and subspecialty settings across the United States and the United Kingdom. Symphony achieves state-of-the-art results across all settings, establishing itself as a flexible, deployment-ready foundation for automated clinical coding.

1 Introduction

In August 2025, the Corti team introduced Code Like Humans (CLH), a fundamentally different approach to automated medical coding [Motzfeldt et al., 2025]. Rather than treating coding as a purely supervised multi-label classification problem learned from a fixed dataset [Huang et al., 2022, Edin et al., 2023], CLH decomposed the task into a structured multi-agent workflow that emulates the reasoning process of expert human coders with full access to reference materials. This paradigm shifted the focus from closed-set statistical prediction to ontology-aware, reasoning-driven code assignment. It has since inspired a new class of agentic coding systems, with teams at Oracle Health & AI and AWS AI achieving state-of-the-art performance using approaches based on large language models such as OpenAI's GPT and Anthropic's Claude [Zheng et al., 2025, Yuan et al., 2025a].

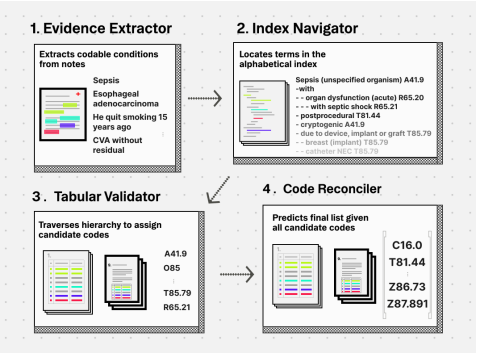

Figure 1: Overview of the reasoning workflow learned in Symphony for Medical Coding.

†Equal contribution.

*Correspondence to `lm@corti.ai`, Corti. To get access to Symphony for Medical Coding, visit Corti Docs.

We now present **Symphony for Medical Coding**, a next-generation medical coding system that advances both performance and deployment readiness. Symphony achieves state-of-the-art accuracy while preserving the flexibility to work with any coding ontology, even as it changes.

In this paper, we make the following contributions:

1. **Benchmark performance:** We demonstrate state-of-the-art results on major public medical coding benchmarks, outperforming leading foundation models and healthcare software providers.
2. **Real-world validation:** We report superior performance on production data from a large U.S. and U.K. provider system, highlighting robustness beyond curated benchmark datasets.
3. **Subspecialty adaptation:** We show that Symphony can be systematically adapted to subspecialty-specific contexts, yielding further performance improvements without retraining.
4. **High-precision operation:** We demonstrate that the system maintains state-of-the-art precision, a critical requirement in safety-sensitive environments.
5. **Evidence-grounded explainability:** We provide span-level evidence attribution for each predicted code, enabling transparent and auditable decision support.
6. **Agentic integration:** We illustrate how Symphony functions as a modular component within larger multi-agent systems, supporting long-running and autonomous clinical workflows.
7. **Ontology scalability:** We demonstrate rapid onboarding of new coding systems and seamless adaptation to updates without retraining, enabling continuous alignment with evolving medical standards.

Symphony for Medical Coding is accessible via a production-grade API and can be deployed either as a standalone coding service or as part of a broader agentic framework[1].

2 Background

Clinical care generates vast amounts of unstructured data, including progress notes, lab reports, prescriptions, referral letters, treatment plans, and administrative documentation [Liu et al., 2022]. Hidden within this unstructured data is information essential for medical decision-making, downstream operations, and medical science. To use this information, we must structure the data.

Healthcare systems structure the data by using medical coding systems: the process of translating clinical information into standardized, machine-readable codes that represent diagnoses, procedures, phenotypes, medications, and laboratory results. In the US, classification systems such as ICD-10, ICD-10-PCS, and CPT form the foundation of healthcare data systems, enabling reliable storage, automated validation, consistency checks, and auditing [Barta, 2009]. These taxonomies are developed and maintained by large expert committees and reflect broad clinical consensus. They are continuously updated to reflect new medical knowledge, changing practices, and evolving regulations, making them living references of clinical reality.

However, assigning these codes remains largely manual. It is both slow and expensive. In Scotland, a medical coder processes roughly 60 cases per day at 7–8 minutes per case [Dong et al., 2022]. In the US, time per case ranges from under a minute for outpatient encounters to over thirty minutes for inpatient stays [Tseng et al., 2018, Stanfill et al., 2014]. These bottlenecks create backlogs that can delay coding by months, sometimes even a full year [Alonso et al., 2020]. The financial cost is equally stark. In 2012, billing and insurance-related activities (of which coding is a major component) cost the U.S. healthcare system an estimated $471 billion ($330B–$597B) [Jiwani et al., 2014]. In systems where physicians perform coding themselves, this burden directly reduces time available for patient care.

Beyond being slow and costly, manual coding is error-prone. A systematic review of thirty-two studies found that the median accuracy of the primary diagnosis code was just 80.3% (IQR: 63.3–94.1%) [Burns et al., 2012]. These errors have cascading consequences. Healthcare providers are under-reimbursed by governments and insurers. Clinical research that relies on coded data is

[1]Find more information on docs.corti.ai/coding or try a demo on console.corti.app

contaminated with noise [Jørgensen and Brunak, 2021]. Patients may be directly harmed by an incorrect medical history; a pilot erroneously coded with depression, for example, may be denied licensure [Dong et al., 2022].

An AI system capable of reliably mapping free-text clinical narratives to standardized medical codes could address these challenges simultaneously. By converting unstructured documentation into validated, ontology-aligned representations, such a system would directly strengthen downstream analytics, quality reporting, and regulatory compliance [Dong et al., 2022]. Perhaps most immediately, it would alleviate pressure on clinical documentation integrity (CDI) and revenue cycle management (RCM): two of the most labor-intensive and resource-constrained processes in modern healthcare operations.

Efforts to automate coding have progressed substantially, from early rule-based systems [Farkas and Szarvas, 2008, Campbell and Giadresco, 2020] through supervised machine learning approaches [Mullenbach et al., 2018, Edin et al., 2023] to more recent work with large language models [Soroush et al., 2024, Kwan, 2024]. Despite this progress, most existing approaches remain fundamentally constrained in two ways. First, they are locked to the coding system they were trained on. Because they learn statistical associations between clinical text and a fixed set of codes, they cannot generalize to other ontologies or accommodate yearly updates to classification systems without being retrained on newly labelled data. Second, they are opaque. The reasoning behind a suggested code is distributed across billions of learned parameters, offering clinicians no interpretable justification they can verify or challenge [Edin et al., 2024]. In combination, these limitations create a gap between benchmark performance and practical usability.

Beyond these methodological constraints, most prior work is limited in the scope of its evaluation. Studies typically report results on a single dataset drawn from one hospital and one clinical specialty, leaving it unclear whether performance generalises across settings [Mullenbach et al., 2018, Dong et al., 2022, Edin et al.]. This is a particularly acute concern for ICD-10 coding, where inpatient and outpatient encounters follow different coding guidelines, meaning that a system validated in one setting cannot be assumed to transfer to the other [Barta, 2009]. The field's heavy reliance on MIMIC, a publicly available inpatient dataset, has further narrowed the evidence base, raising the risk that reported results reflect overfitting to the idiosyncrasies of a single institution rather than genuine coding ability [Johnson et al., 2016, 2023].

We propose Symphony for Medical Coding, a next-generation system that can adapt to arbitrary coding ontologies without retraining and that provides span-level evidence attribution for every predicted code, making its reasoning transparent and auditable. We evaluate Symphony across five datasets spanning both inpatient and outpatient settings, multiple specialties, and two countries (the United States and the United Kingdom), providing the most comprehensive assessment of an automated coding system to date.

3 Methods

3.1 Symphony for Medical Coding

Symphony for medical coding is an extension to the CLH framework [Motzfeldt et al., 2025]. The CLH framework formulates medical coding as a structured, multi-step reasoning process rather than a single-step classification task. Inspired by the workflow of expert human coders, CLH assigns codes across multiple subsequent steps, explicitly incorporating official coding resources and conventions.

The framework consists of four pillars:

1. **Evidence Extraction:** Identify which medical codes should be coded within a clinical text.
2. **Index Navigation:** For each medical code, find the term in the alphabetic index that best describes it. This term will refer to a location within the code hierarchy.
3. **Tabular Validation:** From the location, navigate in the hierarchy to find the most precise code.
4. **Code Reconciliation:** In the end, consider all the identified codes and remove those that are mutually exclusive.

Each stage is implemented as an LLM-powered agent with access to structured coding resources, enabling ontology-aware reasoning over the full medical coding label space (approximately 70,000 codes for ICD-10-CM). By explicitly modeling the medical coding steps followed by human experts, CLH supports open-set coding and improves robustness on rare and long-tail codes compared to conventional supervised multi-label classifiers.

Symphony introduces several improvements over the original CLH implementation. These include more a priori knowledge, and stronger reasoning and search capabilities. These improvements result in substantial performance gains.

Since Symphony extracts the mentions of diagnoses and procedures that should be coded, it naturally provides span-level evidence for each code. This explainability makes it particularly well-suited for integration into larger clinical coding pipelines (see Fig. 2): enabling transparent validation, auditing, and human-in-the-loop review, which are critical requirements in safety-sensitive and regulatory environments. This explicit grounding enables downstream systems and medical coders to efficiently verify decisions, resolve ambiguities, and ensure compliance with coding guidelines. As a result, Symphony can function not only as an autonomous coding system but also as a reliable decision-support component within hybrid workflows that combine automation with expert oversight.

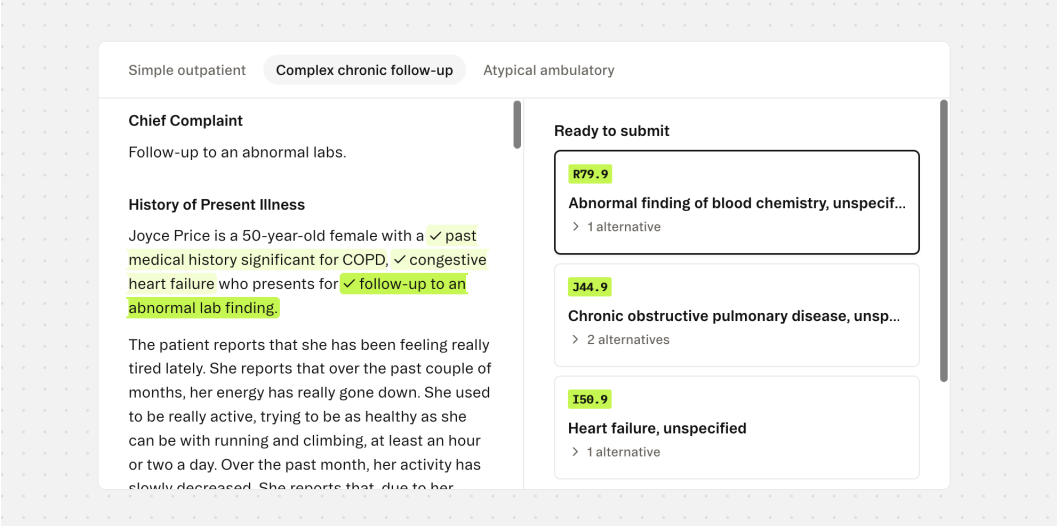

Figure 2: A sample user interface showing the evidence spans provided by Symphony for Medical Coding.

Because Symphony breaks medical coding into explicit steps — extracting evidence, finding candidate codes, and verifying guidelines — its intermediate outputs are structured and well-defined. This makes it a natural building block in larger multi-agent clinical systems[2], where upstream agents can feed it structured inputs and downstream agents can act directly on its outputs, without duplicating reasoning or re-processing raw clinical documentation.

3.2 Data

We evaluate Symphony on five datasets that collectively span inpatient and outpatient settings, multiple clinical specialties, two countries, and both public and proprietary sources.

3.2.1 Public datasets

ACI ACI-BENCH is a small outpatient dataset originally developed for evaluating AI scribes [Yim et al., 2023]. Each case consists of a transcribed simulated dialogue between a physician and patient, from which a separate physician wrote a clinical note. Yuan et al. [2025a] subsequently engaged professional medical coders to assign ICD-10-CM codes to these notes. We used the clinical note and

[2]Symphony for Medical Coding is available, not only as an API endpoint, but also as MCPs that can plug into a multi-agent system (See Corti's Agents Library).

Dataset	Source	#Diagnosis Codes	#Procedure Codes
ACI	Public	225	-
MDACE	Public	904	118
Neurology (NEURO)	Proprietary	667	-
Emergency Departement (ED)	Proprietary	11,148	655
Ambulatory (AMB)	Proprietary	17,561	2,640

Table 1: Summary of evaluation datasets. The number of unique diagnosis and procedure codes reflects the target code space per dataset across all splits. Public datasets come from prior academic benchmarks; proprietary datasets are drawn from production clinical workflows.

the ICD-10-CM codes for the evaluation. While limited in scale, ACI-BENCH provides a controlled outpatient evaluation setting with high-quality annotations.

MDACE MDACE comprises 302 intensive care unit encounters drawn from MIMIC-III [Cheng et al., 2023]. Professional medical coders reannotated each encounter with ICD-10-CM and ICD-10-PCS codes, and annotations were cross-validated across coders to ensure quality. Each encounter contains multiple clinical note types. Previous work has evaluated on individual notes [Motzfeldt et al., 2025, Yuan et al., 2025a], but we concatenate all notes within each encounter into a single document, as we found that codes were frequently missing when individual note types were evaluated in isolation.

3.2.2 Proprietary datasets

To evaluate performance in real-world clinical settings, we include two proprietary datasets reflecting distinct care environments.

Emergency department (ED) The first dataset comprises 563,153 clinical notes containing 11,148 unique ICD-10-CM codes from an emergency department at a large U.S. provider system. The ED setting presents a particularly demanding test for automated coding. Documentation is often fragmented and symptom-driven, produced under time pressure with high diagnostic uncertainty. Coding decisions frequently depend on incomplete or provisional information, which makes coding particularly challenging.

Ambulatory (AMB) The second dataset consists of 2,250,380 ambulatory (outpatient) clinical notes containing 17,561 unique ICD-10-CM codes from the same U.S. provider system. In contrast to the ED setting, ambulatory care reflects scheduled, longitudinal interactions with more structured documentation focused on chronic disease management and preventive care. Coding patterns tend to be denser and more stable, often involving broad sets of comorbidities. This dataset tests the system's ability to handle fine-grained diagnostic distinctions across a large code space.

Neurology (NEURO) The third proprietary dataset comprises 10,675 clinical notes containing 667 unique codes from a neurology department within the UK National Health Service. This dataset uses the NHS version of ICD-10, which differs from the ICD-10-CM system used in the U.S. datasets in both structure and code granularity. We use it specifically to demonstrate Symphony's ability to adapt to an entirely different coding ontology without retraining. It also introduces a subspecialty setting with a concentrated, technically demanding code distribution. Finally, it tests cross-national generalisability, as UK clinical documentation follows different conventions and terminology than its U.S. counterparts.

4 Experiments

4.1 Evaluation setting

Prior work in medical coding often simplifies the task by restricting prediction to a predefined subset of codes [Edin et al., 2023]. In practice, this reduces the output space from the full classification

system to a much smaller label set, often on the order of only 1,000 codes. While this setup enables the use of standard supervised classification approaches, it substantially alters the nature of the problem.

This restriction introduces two key limitations. First, it artificially makes the problem easier by excluding neighboring codes from the evaluation. For example, there are numerous medical codes for *diabetes mellitus*, each specifying the type and any complications. If only one or two of these appear in the filtered test set, the system is never penalised for confusing closely related codes that it would encounter in practice. Reported accuracy, therefore, overstates what could be expected in deployment against the full ontology. Second, it assumes prior knowledge of the relevant label space, which is not always available in deployment settings where the system must operate over the full, evolving classification system. As a result, performance reported under restricted label settings may significantly overestimate real-world utility.

Symphony for Medical Coding can easily adapt to a sub-selection of codes. We evaluate systems under two complementary settings:

- **Restricted code system evaluation:** Models predict over a predefined subset of codes, following common practice in prior work. This setting enables direct comparison to existing benchmarks.
- **Full code system evaluation:** Models predict over the complete coding system without label restrictions, reflecting real-world deployment requirements.

We evaluate each system five times and report its mean and standard deviation. Some of the results we do not reproduce, but instead copy from their respective papers. Most of these studies did not evaluate their models multiple times; therefore, we cannot report the mean and standard deviation.

4.1.1 Evaluation metrics

Similar to prior work, we evaluate all systems using precision, recall, and F1 score [Edin et al., 2023, Zhang et al., 2025].

Precision measures the proportion of predicted codes that are correct, i.e., how much you can trust a prediction:

$$\text{Precision (P)} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

where TP denotes true positives, and FP denotes false positives. A system with low precision frequently assigns codes that do not belong to the encounter, generating noise for downstream billing and analytics.

Recall measures the proportion of true codes that are successfully identified:

$$\text{Recall (R)} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

where FN denotes false negatives. A system with low recall misses codes that should have been assigned, leading to incomplete clinical records and potential under-reimbursement.

The F1 score combines both into a single measure via their harmonic mean:

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

The harmonic mean ensures that a system must perform well on both precision and recall to achieve a high F1; strong performance on one cannot compensate for poor performance on the other.

These metrics can be aggregated across codes in two ways. Micro-averaging pools all true positives, false positives, and false negatives across codes before computing a single score, effectively weighting each prediction equally. Macro-averaging instead computes the metric independently for each code and then takes the unweighted mean, giving equal weight to every code regardless of its frequency. Both are widely reported in the medical coding literature, and the choice between them can substantially affect how system performance is perceived.

We report micro-averaged scores. This choice is motivated by two properties of our evaluation setting. First, macro-averaging has low statistical power when most classes are rare and test sets are small,

because a single example being correctly or incorrectly classified can swing a class-level F1 from 0 to 1, or vice versa. Second, macro-averaging can mask systematic overprediction. If a system predicts I10 (essential hypertension) for every encounter, the precision for that code collapses, but because I10 is only one class out of thousands, its effect on the macro-average is negligible. Micro-averaging, by contrast, counts every false positive equally, making such behaviour immediately visible in the overall score.

These metrics should be interpreted with some caution, as the target labels themselves are not always reliable. Medical coding is a subjective, guideline-driven process, and prior studies have shown that even professional coders often disagree on the correct codes [Cheng et al., 2023]. This implies that the reference annotations used for evaluation are inherently uncertain and may not reflect a single, unambiguous ground truth. As a result, there exists a practical upper bound on achievable F1 scores when evaluating against a single set of annotations, since even expert coders would not consistently agree with that reference. Model performance should therefore be interpreted in the context of this label noise, particularly when disagreements arise from alternative yet clinically plausible coding decisions.

Furthermore, the long-tailed distribution of medical codes complicates the interpretation of aggregate metrics such as F1. A small number of common codes account for a large fraction of encounters, while the majority of codes appear only rarely. In practice, this means that a system can achieve near-perfect accuracy on high-volume, well-defined coding patterns while still receiving a moderate overall F1 score due to errors on rare or highly specific codes. Consequently, F1 alone does not fully capture the extent to which a system can automate real-world coding workloads. High performance on frequent and clinically critical code combinations may enable substantial automation in practice, even if the aggregate F1 score suggests room for improvement.

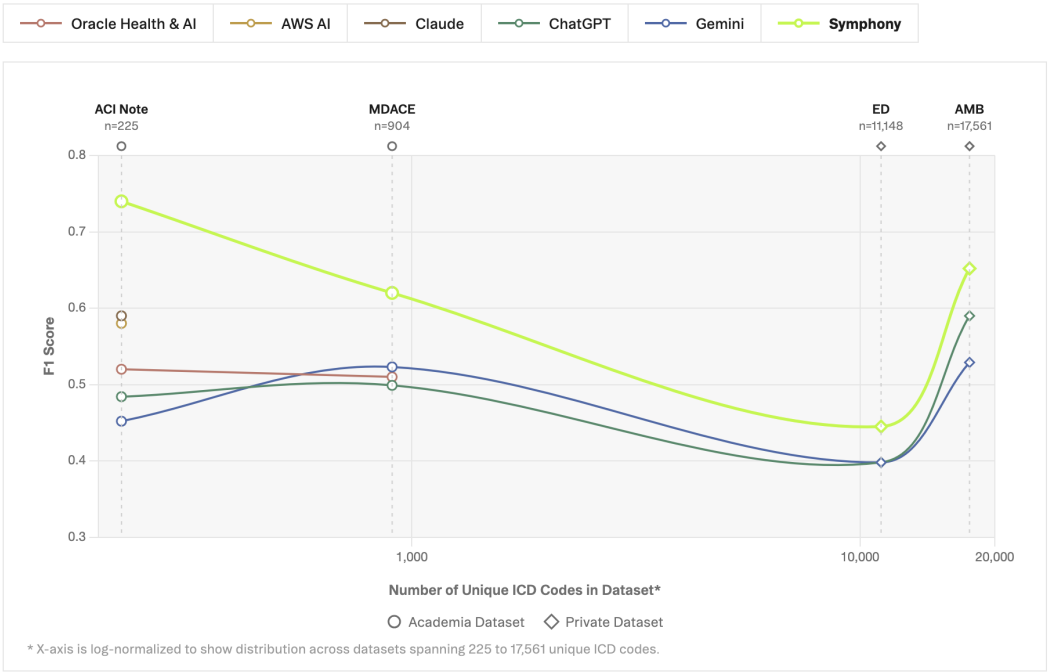

Figure 3: F1 performance as a function of code space size (log-scaled) from Yuan et al. [2025b], Zheng et al. [2025] and experiments in this paper. Each point corresponds to a dataset with a distinct number of unique codes. Symphony demonstrates superior performance across label space complexity.

4.2 Results

4.2.1 Restricted code system evaluation

Table 2 presents results under the restricted code system setting. Across both datasets, Symphony achieves the highest overall performance, significantly outperforming prior approaches.

We group the compared methods into three categories based on their underlying approach. **Fine-tuned models** are supervised approaches trained directly for code prediction over a fixed label set. Claude was fine-tuned on ACI, while PLM-CA and PLM-ICD were on MIMIC-IV [Nguyen et al., 2023]. **Agent-based methods** use large language models with prompting strategies such as chain-of-thought and self-consistency to perform coding through reasoning, without explicit task-specific training. **Workflow-based methods** extend this paradigm by decomposing the coding task into multiple coordinated steps or agents, often combining reasoning, retrieval, and verification in structured pipelines.

On the ACI dataset, Symphony reaches an F1 score of 0.74, improving substantially over the strongest baseline, MedDCR (0.52), developed by Oracle Health & AI on the OpenAI GPT family of models, and over the best fine-tuned model, Claude (0.58), developed by the AWS AI team on the Anthropic Claude model family. This gain is driven by a strong balance between recall (0.81) and precision (0.68), indicating that Symphony can both identify relevant codes and maintain high precision in its predictions.

On MDACE, Symphony similarly achieves the best performance with an F1 score of 0.58, surpassing MedDCR (0.51), developed by Oracle Health & AI on GPT-based models, and fine-tuned baselines such as PLM-ICD (0.48). Notably, Symphony maintains a more balanced precision-recall trade-off compared to prior methods, which often exhibit either high recall with low precision or vice versa.

Across both datasets, prior agentic and workflow-based approaches tend to suffer from low precision, despite achieving moderate recall. In contrast, Symphony consistently improves precision while maintaining competitive or higher recall, resulting in substantially higher F1 scores.

Finally, the low standard deviation across runs indicates that Symphony produces stable and reproducible results.

Table 2: **Restricted code system evaluation**. Symphony in comparison to prior approaches, including fine-tuned models [Huang et al., 2022, Edin et al., 2024, Yuan et al., 2025b] (with Claude based on the Anthropic Claude model family developed by AWS AI), agent-based methods, and workflow-based systems such as MedDCR [Zheng et al., 2025] (built by Oracle Health & AI on the OpenAI GPT model family) and related approaches [Kwan, 2024, Li et al., 2024]. We evaluated Symphony five times. The table shows the mean value and the standard deviation across runs. The results for the other models are taken from their respective papers. The best F1-score is shown in bold.

		ACI			MDACE		
Method	**Model**	**R**	**P**	**F1**	**R**	**P**	**F1**
Fine-tune	PLM-ICD	0.41	0.43	0.42	0.47	0.49	0.48
	PLM-CA	0.42	0.44	0.43	0.45	0.46	0.45
	Claude	-	-	0.58	-	-	-
Agent	CoT	0.50	0.35	0.41	0.31	0.30	0.30
	CoT-SC	0.59	0.36	0.44	0.43	0.39	0.41
	MulDe	0.65	0.16	0.25	0.40	0.21	0.28
	Judge	0.64	0.22	0.33	0.53	0.26	0.35
	ADAS	0.59	0.28	0.43	0.51	0.37	0.43
Workflow	RRS	0.52	0.26	0.35	0.30	0.24	0.27
	MAC	0.50	0.23	0.31	0.31	0.27	0.29
	MedDCR	0.67	0.43	0.52	0.59	0.46	0.51
	Symphony	$0.81_{\pm 0.01}$	$0.68_{\pm 0.01}$	$\mathbf{0.74_{\pm 0.01}}$	$0.66_{\pm 0.02}$	$0.59_{\pm 0.02}$	$\mathbf{0.62_{\pm 0.02}}$

4.2.2 Full code system evaluation

Table 3 and 4 present results under the full code system setting, where models are required to predict over the complete coding space without label restrictions. To the best of our knowledge, such evaluations have not been reported by model providers, despite both Anthropic and OpenAI positioning their models as ready for healthcare applications.

We compare three classes of systems[3]: foundation models without tools, foundation models augmented with tools, and specialized coding systems. The *with tools* setting for Claude leverages the Anthropic Model Context Protocol (MCP) adapted for medical coding, enabling structured interaction with external coding resources. While tool augmentation improves performance in some cases, results remain inconsistent across datasets and models.

Across all datasets, Symphony achieves the strongest and most consistent performance, outperforming both standalone foundation models and tool-augmented variants. In contrast, general-purpose models tend to exhibit high recall but substantially lower precision, particularly in the full-classification system setting, reflecting the challenge of operating over large and highly imbalanced label spaces. These results highlight the gap between general-purpose LLM capabilities and the requirements of real-world medical coding, and demonstrate the importance of structured, ontology-aware systems for scalable deployment.

Table 3: **Full code system evaluation** on public datasets. F1-scores (multiplied by 100) across models: Anthropic Claude (Opus 4.6), OpenAI GPT (5.4), Google Gemini (3.1 Pro), and Corti Symphony. We evaluated each model five times. The table shows the mean value and the standard deviation across runs. The best score for each metric is shown in bold. †Anthropic did not want to give us access to a HIPAA compliant model so we could only evaluate Claude on ACI data.

		ACI			MDACE		
Method	**Model**	**R**	**P**	**F1**	**R**	**P**	**F1**
No tools	Claude†	51.0±4.1	45.3±8.8	47.6±5.2	-	-	-
	ChatGPT	74.2±1.8	42.5±1.2	54.0±1.4	66.0±1.9	43.1±1.3	52.1±1.4
With tools	Claude†	60.3±8.47	58.13±8.47	59.0±6.1	-	-	-
	ChatGPT	72.4±0.9	36.4±0.9	48.4±0.9	56.0±1.5	45.1±1.3	49.9±1.3
	Gemini	76.5±1.1	32.1±0.1	45.2±0.1	60.0±0.6	46.4±0.4	52.3±0.5
Corti	CLH	56.0±1.0	58.5±1.3	57.2±1.0	30.7±0.7	51.9±0.8	38.5±0.5
	Symphony	75.0±1.2	56.0±2.0	**64.1±1.7**	61.8±1.7	49.6±1.0	**55.0±0.2**

In Table 5 we evaluated Symphony on the U.K. NEURO coding dataset. We compare Symphony against GPT-based models, as these were the only foundation models we could evaluate in a GDPR-compliant setting. Given the similar performance patterns observed across ChatGPT, Claude, and Gemini in the experiments above, we consider ChatGPT to be a representative baseline for this setting.

These results highlight Symphony's ability to generalize across coding systems and geographic standards. Despite differences in coding guidelines, terminology, and clinical documentation practices between U.S. and U.K. settings, Symphony maintains strong performance without retraining, achieving a substantial improvement over GPT (32.9 vs. 18.1 F1). This suggests that Symphony's ontology-aware and reasoning-driven approach is not tied to a specific coding system, but instead captures underlying coding principles that transfer effectively across domains. The performance gap further indicates that structured workflows are particularly important when adapting to new coding regimes, where reliance on parametric knowledge alone is insufficient.

A further insight from the U.K. evaluation is the presence of systematic under-specification in the reference annotations. In additional analyses, we observe that a substantial portion of disagreements arises from differences in specificity rather than incorrect clinical interpretation. In many cases,

[3]Despite multiple attempts to obtain a HIPAA-compliant deployment option from Anthropic, we were unable to process PHI-containing datasets with Claude.

Table 4: **Full code system evaluation** on Corti's proprietary datasets. F1-scores (multiplied by 100) across models: Anthropic Claude (Opus 4.6), OpenAI GPT (5.4), Google Gemini (3.1 Pro), and Corti Symphony. We evaluated each model five times. The table shows the mean value and the standard deviation across runs. The best score for each metric is shown in bold. †Anthropic would not give us access to a HIPAA complient model so we could only evaluate Claude on synthetic data.

		AMB			ED		
Method	**Model**	**R**	**P**	**F1**	**R**	**P**	**F1**
No tools	Claude†	-	-	-	-	-	-
	ChatGPT	77.0±0.4	47.6±0.8	58.8±0.6	42.3±0.7	42.8±0.5	42.5±0.5
With tools	Claude†	-	-	-	-	-	-
	ChatGPT	70.4±0.6	50.8±0.4	59.0±0.4	46.6±0.3	34.7±0.4	39.8±0.3
	Gemini	42.6±0.2	67.8±0.7	52.3±0.5	27.1±0.3	35.8±0.4	39.8±0.3
Corti	CLH	51.1±0.6	62.0±0.2	56.0±0.4	26.5±0.5	49.9±0.8	34.6±0.6
	Symphony	64.4±0.3	66.0±0.5	**65.2±0.2**	37.6±0.3	54.3±0.8	**44.5±0.5**

Table 5: **Full code system evaluation for ICD-10 (UK)**. F1-scores (multiplied by 100) on OpenAI GPT (5.4) and Corti Symphony. We evaluated each model five times. The table shows the mean value and the standard deviation across runs. The best score for each metric is shown in bold. The table shows the mean value and the standard deviation across runs. The best score for each metric is shown in bold.

	NEURO		
Model	**R**	**P**	**F1**
ChatGPT	21.3±0.3	15.8±0.2	18.1±0.2
Symphony	49.6±0.4	24.6±0.5	**32.9±0.5**

the reference data assigns broader, unspecified codes where the clinical documentation supports more precise alternatives selected by Symphony. This pattern, commonly observed in NHS coding practice, reflects a tendency toward conservative coding choices rather than maximal specificity. As a result, part of the residual error measured at the code level may stem from limitations in the reference annotations rather than model performance. This further highlights the value of structured, reasoning-based systems in promoting consistent and clinically precise coding.

Table 6: **Full procedure coding system evaluation**. F1-scores (multiplied by 100) on OpenAI GPT (5.4) and Corti Symphony. We evaluated each model five times. The table shows the mean value and the standard deviation across runs. The best score for each metric is shown in bold. †Symphony for CPT is in `beta`. The results may not generalize well to some specialties (updates are coming soon).

	ICD-10-PCS			CPT†					
	MDACE			AMB			ED		
Model	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**
ChatGPT	34.8±2.4	26.4±3.1	30.0±2.9	31.1±0.6	66.0±1.8	42.2±0.6	21.0±1.1	20.5±0.8	20.8±0.9
Symphony	37.5±0.8	37.0±0.6	**37.3±0.4**	51.8±0.4	50.8±0.4	**51.3±0.4**	48.8±0.6	34.2±0.3	**40.2±0.4**

4.2.3 Procedure coding evaluation

While the previous results were focused on diagnosis coding, Table 6 presents results on ICD-10-PCS coding using the MDACE dataset and CPT using the AMB and ED dataset. We compare Symphony against GPT-based models, as these were the only foundation models we could evaluate in a HIPAA-compliant setting for procedure coding tasks. Given the similar performance patterns observed across GPT, Claude, and Gemini in the experiments above, we consider GPT to be a representative baseline for this setting.

Symphony achieves the best overall performance across systems. On ICD-10-PCS (MDACE), Symphony attains an F1 score of 37.3, outperforming GPT (30.0). While GPT achieves moderate recall, it exhibits a pronounced imbalance with substantially lower precision, limiting overall performance. In contrast, Symphony maintains a near-equal balance between recall (37.5) and precision (37.0), resulting in a significantly higher F1 score.

This advantage extends to CPT coding. On the ambulatory dataset, Symphony achieves an F1 score of 51.3 compared to 42.2 for GPT, while maintaining balanced recall (51.8) and precision (50.8). On the ED dataset, Symphony similarly outperforms GPT (40.2 vs. 20.8), again with a more balanced precision-recall profile. These results highlight both the increased difficulty of procedure coding compared to diagnosis coding and the importance of structured reasoning and ontology-aware workflows when operating over large and heterogeneous coding systems such as ICD-10-PCS and CPT.

4.2.4 Evidence-span Explainability Analysis

Each clinical note in MDACE is annotated with evidence spans: spans of text that contain the evidence for a code [Cheng et al., 2023]. We compare Symphony's extracted evidence spans with those annotated in the dataset. Table 7 summarizes the quality of the evidence spans produced by Symphony on the MDACE dataset. The system provides evidence for nearly all correct predictions, achieving a coverage of 98.9%. Among these, 73.7% of predicted spans overlap with human-annotated evidence, with 42.5% showing substantial alignment (IoU > 0.5). At the token level, Symphony attains a character-level F1 of 0.459 and a ROUGE-L F1 of 0.506, indicating that the predicted evidence captures the relevant clinical content while allowing for some flexibility in span boundaries. The average span IoU of 0.471 further supports that the model frequently highlights similar regions of text as human annotators. Note, that the human-annotated evidence spans in MDACE do not cover all mentions of a code, and often do not contain full sentences [Cheng et al., 2023, Beckh et al., 2025a,b, Douglas et al., 2025]. The true scores of Symphony's extracted evidence spans is therefore higher than the scores indicate. Overall, these results show that Symphony produces reliable and well-grounded evidence, with most discrepancies arising from differences in annotation granularity rather than incorrect reasoning.

Table 7: **Evidence span explainability on MDACE**. Key metrics evaluating alignment between predicted and human-annotated evidence spans.

Metric	Value
Coverage (≥ 1 predicted span)	98.9%
Hit rate (IoU > 0)	73.7%
Hit rate (IoU > 0.5)	42.5%
Character-level F1	0.459
ROUGE-L F1	0.506
Span IoU (avg)	0.471

5 Conclusion

We introduced Symphony for Medical Coding, an ontology-aware agentic system for scalable, explainable, and deployment-ready medical coding. Across both restricted-label and full-classification system evaluations, Symphony achieves state-of-the-art performance on public benchmarks and on real-world provider data, while maintaining the high precision required for safety-sensitive clinical workflows. In contrast to prior supervised approaches that operate over fixed subsets of codes, Symphony supports reasoning over full and evolving coding systems without retraining, and provides span-level evidence attribution for transparent and auditable decision support.

These results show that medical coding benefits from structured, ontology-aware reasoning rather than closed-set prediction alone. They also demonstrate that Symphony can serve not only as a

high-performance coding engine, but as a modular component within broader autonomous clinical systems.

Acknowledgements

We thank Casper Christensen, Majed Sharif, Maxwell White, Nicklas Frahm, Maciej Tatarski, Henrik Cullen, and Dan Engel for their valuable contributions, insightful discussions, and support throughout the development and evaluation of this work.

References

Andreas Geert Motzfeldt, Joakim Edin, and Lars Maaløe. Code Like Humans: A Multi-Agent Solution for Medical Coding. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, volume Findings of the Association for Computational Linguistics: EMNLP 2025. Association for Computational Linguistics, 2025.

Chao-Wei Huang, Shang-Chi Tsai, and Yun-Nung Chen. PLM-ICD: Automatic ICD Coding with Pretrained Language Models. In *Proceedings of the 4th Clinical Natural Language Processing Workshop*, pages 10–20, Seattle, WA, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.clinicalnlp-1.2.

Joakim Edin, Alexander Junge, Jakob Drachmann Havtorn, Lasse Borgholt, Maria Maistro, Tuukka Ruotsalo, and Lars Maaløe. Automated Medical Coding on MIMIC-III and MIMIC-IV: A Critical Review and Replicability Study. *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023. doi: 10.48550/arXiv.2312.13533.

Jiyang Zheng, Islam Nassar, Thanh Vu, Xu Zhong, Yang Lin, Tongliang Liu, Long Duong, and Yuan-Fang Li. MedDCR: Learning to Design Agentic Workflows for Medical Coding, November 2025.

Moy Yuan, Han-Chin Shing, Mitch Strong, and Chaitanya Shivade. Toward Reliable Clinical Coding with Language Models: Verification and Lightweight Adaptation. In Saloni Potdar, Lina Rojas-Barahona, and Sebastien Montella, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 173–184, Suzhou (China), November 2025a. Association for Computational Linguistics. ISBN 979-8-89176-333-3. doi: 10.18653/v1/2025.emnlp-industry.12.

Jinghui Liu, Daniel Capurro, Anthony Nguyen, and Karin Verspoor. "Note Bloat" impacts deep learning-based NLP models for clinical prediction tasks. *Journal of Biomedical Informatics*, 133: 104149, September 2022. ISSN 1532-0464. doi: 10.1016/j.jbi.2022.104149.

Ann Barta. ICD-10-CM official coding guidelines. *Journal of AHIMA*, 80(6):70–71, June 2009. ISSN 1060-5487.

Hang Dong, Matúš Falis, William Whiteley, Beatrice Alex, Joshua Matterson, Shaoxiong Ji, Jiaoyan Chen, and Honghan Wu. Automated clinical coding: What, why, and where we are? *npj Digital Medicine*, 5(1):1–8, October 2022. ISSN 2398-6352. doi: 10.1038/s41746-022-00705-7.

Phillip Tseng, Robert S. Kaplan, Barak D. Richman, Mahek A. Shah, and Kevin A. Schulman. Administrative Costs Associated With Physician Billing and Insurance-Related Activities at an Academic Health Care System. *JAMA*, 319(7):691–697, February 2018. ISSN 0098-7484. doi: 10.1001/jama.2017.19148.

Mary H. Stanfill, Kang Lin Hsieh, Kathleen Beal, and Susan H. Fenton. Preparing for ICD-10-CM/PCS Implementation: Impact on Productivity and Quality. *Perspectives in Health Information Management*, 11(Summer):1f, July 2014. ISSN 1559-4122.

Vera Alonso, João Vasco Santos, Marta Pinto, Joana Ferreira, Isabel Lema, Fernando Lopes, and Alberto Freitas. Problems and Barriers during the Process of Clinical Coding: A Focus Group Study of Coders' Perceptions. *Journal of Medical Systems*, 44(3):62, February 2020. ISSN 1573-689X. doi: 10.1007/s10916-020-1532-x.

Aliya Jiwani, David Himmelstein, Steffie Woolhandler, and James G. Kahn. Billing and insurance-related administrative costs in United States' health care: Synthesis of micro-costing evidence. *BMC Health Services Research*, 14(1):556, November 2014. ISSN 1472-6963. doi: 10.1186/s12913-014-0556-7.

E.M. Burns, E. Rigby, R. Mamidanna, A. Bottle, P. Aylin, P. Ziprin, and O.D. Faiz. Systematic review of discharge coding accuracy. *Journal of Public Health (Oxford, England)*, 34(1):138–148, March 2012. ISSN 1741-3842. doi: 10.1093/pubmed/fdr054.

Isabella Friis Jørgensen and Søren Brunak. Time-ordered comorbidity correlations identify patients at risk of mis- and overdiagnosis. *npj Digital Medicine*, 4(1):1–10, January 2021. ISSN 2398-6352. doi: 10.1038/s41746-021-00382-y.

Richárd Farkas and György Szarvas. Automatic construction of rule-based ICD-9-CM coding systems. *BMC Bioinformatics*, 9(3):1–9, April 2008. ISSN 1471-2105. doi: 10.1186/1471-2105-9-S3-S10.

Sharon Campbell and Katrina Giadresco. Computer-assisted clinical coding: A narrative review of the literature on its benefits, limitations, implementation and impact on clinical coding professionals. *Health Information Management Journal*, 49(1):5–18, January 2020. ISSN 1833-3583. doi: 10.1177/1833358319851305.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. Explainable Prediction of Medical Codes from Clinical Text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1101–1111, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1100.

Ali Soroush, Benjamin S. Glicksberg, Eyal Zimlichman, Yiftach Barash, Robert Freeman, Alexander W. Charney, Girish N Nadkarni, and Eyal Klang. Large Language Models Are Poor Medical Coders — Benchmarking of Medical Code Querying. *NEJM AI*, 1(5):AIdbp2300040, April 2024. doi: 10.1056/AIdbp2300040.

Keith Kwan. Large language models are good medical coders, if provided with tools, July 2024.

Joakim Edin, Maria Maistro, Lars Maaløe, Lasse Borgholt, Jakob Drachmann Havtorn, and Tuukka Ruotsalo. An Unsupervised Approach to Achieve Supervised-Level Explainability in Healthcare Records. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4869–4890, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.280.

Joakim Edin, Sedrah Butt Balaganeshan, Annike Kjølby Kristensen, Lars Maaloe, Giannos Louloudis, and Søren Brunak. A medical coding language model trained on clinical narratives from a population-wide cohort of 1.8 million patients.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li-wei H. Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1):160035, May 2016. ISSN 2052-4463. doi: 10.1038/sdata.2016.35.

Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Benjamin Moody, Brian Gow, Li-wei H. Lehman, Leo A. Celi, and Roger G. Mark. MIMIC-IV, a freely accessible electronic health record dataset. *Scientific Data*, 10(1):1, January 2023. ISSN 2052-4463. doi: 10.1038/s41597-022-01899-x.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. Aci-bench: A Novel Ambient Clinical Intelligence Dataset for Benchmarking Automatic Visit Note Generation. *Scientific Data*, 10(1):586, September 2023. ISSN 2052-4463. doi: 10.1038/s41597-023-02487-3.

Hua Cheng, Rana Jafari, April Russell, Russell Klopfer, Edmond Lu, Benjamin Striner, and Matthew Gormley. MDACE: MIMIC Documents Annotated with Code Evidence. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7534–7550, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.416.

Xu Zhang, Kun Zhang, Wenxin Ma, Rongsheng Wang, Chenxu Wu, Yingtai Li, and S. Kevin Zhou. A General Knowledge Injection Framework for ICD Coding, May 2025.

Zhangdie Yuan, Han-Chin Shing, Mitch Strong, and Chaitanya Shivade. Toward Reliable Clinical Coding with Language Models: Verification and Lightweight Adaptation, October 2025b.

Thanh-Tung Nguyen, Viktor Schlegel, Abhinav Kashyap, Stefan Winkler, Shao-Syuan Huang, Jie-Jyun Liu, and Chih-Jen Lin. Mimic-IV-ICD: A new benchmark for eXtreme MultiLabel Classification, April 2023.

Rumeng Li, Xun Wang, and Hong Yu. Exploring LLM Multi-Agents for ICD Coding, August 2024.

Katharina Beckh, Sven Heuser, and Stefan Rüping. Can ensembles improve evidence recall? A case study, December 2025a.

Katharina Beckh, Elisa Studeny, Sujan Sai Gannamaneni, Dario Antweiler, and Stefan Rueping. The Anatomy of Evidence: An Investigation Into Explainable ICD Coding. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Findings of the Association for Computational Linguistics: ACL 2025*, pages 16840–16851, Vienna, Austria, July 2025b. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.864.

James C. Douglas, Yidong Gan, Ben Hachey, and Jonathan K. Kummerfeld. Less is More: Explainable and Efficient ICD Code Prediction with Clinical Entities. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar, editors, *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 30835–30847, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-251-0. doi: 10.18653/v1/2025.acl-long.1489.