JAMIA Open, 2026, **9(1)**, ooaf123
https://doi.org/10.1093/jamiaopen/ooaf123
Research and Applications

Research and Applications

Leveraging generative AI to enhance Synthea model development

Mark A. Kramer, PhD[1,*], Aanchal Mathur, MS[1], Caroline E. Adams, MS[1], Jason A. Walonoski, MS[1]

[1]MITRE Corporation, Bedford, MA 01730, United States

*Corresponding author: Mark A. Kramer, PhD, MITRE Corporation, 202 Burlington Rd., Bedford, MA 01730, United States (mkramer@mitre.org)

Abstract

Objective: To explore the use of large language models (LLMs) to assist in developing new agent-based disease-specific patient journey models.

Materials and Methods: We focus on Synthea, an open-source synthetic health data generator, with the goal of developing models in less time and with reduced expertise, expanding model diversity, and improving synthetic patient data quality. We apply a 4-stage methodology: (1) using an LLM to extract disease information from authoritative medical sources, (2) using an LLM to create an initial Synthea-compatible model, (3) validating that model through 2-level assessment (structural/syntax validation and requirements satisfaction), and (4) using an LLM to iteratively refine the model based on validation feedback.

Results: Using hyperthyroidism as an example, we tested Claude 3.5 Sonnet, GPT-4o, and Gemini 1.5 Pro. While the LLMs generated initial models that varied widely in quality, all 3 demonstrated significant improvement in requirement fulfillment scores through successive iterations, with final requirement fulfilment scores approaching 100% for Claude and Gemini. However, evaluation by human experts revealed various structural deficits in final models.

Discussion: LLMs can assist in creating patient journey models when combined with structured methodology and authoritative medical knowledge sources. Iterative improvement was shown to be essential in creating models meeting stated requirements. Limitations include frequent medical code inaccuracies, model isolation without comorbidity considerations, and remaining requirements for clinical expertise and human oversight.

Conclusion: LLMs can serve as valuable assistive tools for synthetic health data model development when used within structured, iterative frameworks, although at the time of this writing (mid-2024) LLMs require continued human expertise and validation rather than fully autonomous operation. In principle, this conclusion is not limited to Synthea and could be applied to other agent-oriented patient journey frameworks.

Lay Summary

Artificial-but-realistic patient records are important because they support education and experimentation without introducing privacy risks. Synthea is a computer program that generates synthetic patient records by simulating individual patient journeys through health, sickness, and treatment. Currently, building these disease simulation models requires extensive collaboration between medical and modeling experts. This study explores whether artificial intelligence language models (like ChatGPT) could help speed up this process. We developed a 4-step approach: first, an AI extracts disease information from medical sources; second, the AI creates an initial disease model; third, the model is checked for errors; and fourth, the AI uses feedback to improve the model through multiple rounds of revision. Testing this approach on hyperthyroidism, we found that while the AI could create functional models that improved with feedback, the final results still contained significant flaws. Our conclusion is that AI can serve as a valuable assistant in creating patient simulation models, potentially reducing development time from months to days, but human medical expertise remains essential for ensuring accuracy and clinical validity.

Key words: Synthea; FHIR; synthetic health data; large language models; disease model.

Introduction

Synthetic health data provide a source of realistic-but-not-real patient data that protect patient privacy, help to overcome data scarcity and silos, and enable efficient development and testing of new health applications and models without risk of exposure of personal health data.[1]

Synthea is an open-source software tool for generating synthetic patient data.[2] It creates artificial patient health records that span lifetimes, designed to be realistic while not representing any actual individuals. A Synthea model is a JSON-based configuration file that defines a specific disease or condition pathway that the synthetic patient generator uses

Received: July 21, 2025; **Revised:** September 25, 2025; **Accepted:** September 26, 2025

to simulate patient experiences and generate realistic health records. Each model describes the progression of a medical condition through a series of states, transitions, and clinical events, allowing Synthea to produce clinically accurate disease trajectories and reproduce real-life treatment patterns. A model is comprised of:

- States, representing a point in the progression of a disease or a patient's health status.
- Transitions, defining the movement between states based on certain conditions, such as age, gender, or the presence of specific symptoms.
- Distributions, used to model the variability in patient attributes, such as the age of onset for a disease or the duration spent in a particular state, or probability of an event.
- Codes, representing diagnoses, procedures, and medications, typically based on standardized healthcare vocabularies like SNOMED CT and ICD-10.
- Lab results or other observed values, typically paired with LOINC codes.

A simple Synthea model and its graphical equivalent are shown in Figure 1.

Synthea uses its collection of models to generate patient journeys through various health states, such as the onset of conditions, healthcare encounters, diagnoses, procedures, and medications. The final output is a set of synthetic patient records that include a variety of health data points across the simulated lifetime in formats that comply with healthcare data standards, for example, HL7® FHIR® (Fast Healthcare Interoperability Resources).

Currently, Synthea model design relies on co-development by clinicians and Synthea experts. While this approach ensures understandability, clinical suitability, and reliability, it is time-consuming and may result in limited treatment variations, reflecting clinical practices of the collaborators. This paper explores the use of large language models (LLMs) to assist in developing new agent-based disease-specific patient journey models. If successful, LLM-aided development could help expand Synthea's model library, increase model diversity, and improve the overall quality and variety of synthetic patient data.

Prior work

A comprehensive survey of synthetic data in healthcare by Gonzales et al. provides 89 references in this field through 2023.[1] Prior work can be placed into 4 categories of progressive flexibility and detail: correlational models, probabilistic models, causal models, and agent-based models.

1) **Machine learning** (ML) techniques learn the entire joint distribution of the data simultaneously. Generative adversarial networks (GANs) have emerged as a leading method.[4–6] Variational autoencoders (VAEs) have also shown effectiveness in synthetic health data generation.[7–9] More recent work has involved transformer architectures and hybrid architectures for synthetic EHR data and synthetic clinical notes.[10,11] Federated ML approaches have been explored.[12]
2) **Bayesian networks** (BN) go a step beyond ML techniques by capturing probabilistic dependency in a transparent graphical model, a key advantage for interpretability.[13,14]
3) **Causal frameworks,** by enforcing causal structure, support interventional reasoning (predicting what happens when we change the system), overcoming a key limitation of ML and BN approaches.[15,16]
4) **Agent-based models** create and follow artificial individuals over a period of time, modeling their health states, their decision processes, and interactions. This concept significantly pre-dates LLMs.[17] Agents can possess a wide range of differing characteristics, such as age, gender, socioeconomic status, health conditions, and personal preferences. This allows for exploration of entirely new scenarios, for example, exploring how health insurance coverage impacts breast cancer survival rates.[18] Medical education is a common use case for patient simulation.[19–21] Process mining techniques have been used to derive clinical pathways from real-world data[22,23] In addition to Synthea, popular agent simulation toolkits include AnyLogic[24] and FRED.[25]

Methodology

We developed a 4-stage methodology for creating Synthea models using LLMs:

1) Generating disease information in a useful form,
2) Creating an initial disease model from that information,
3) Validating the model, and
4) Iteratively improving the model based on validation results.

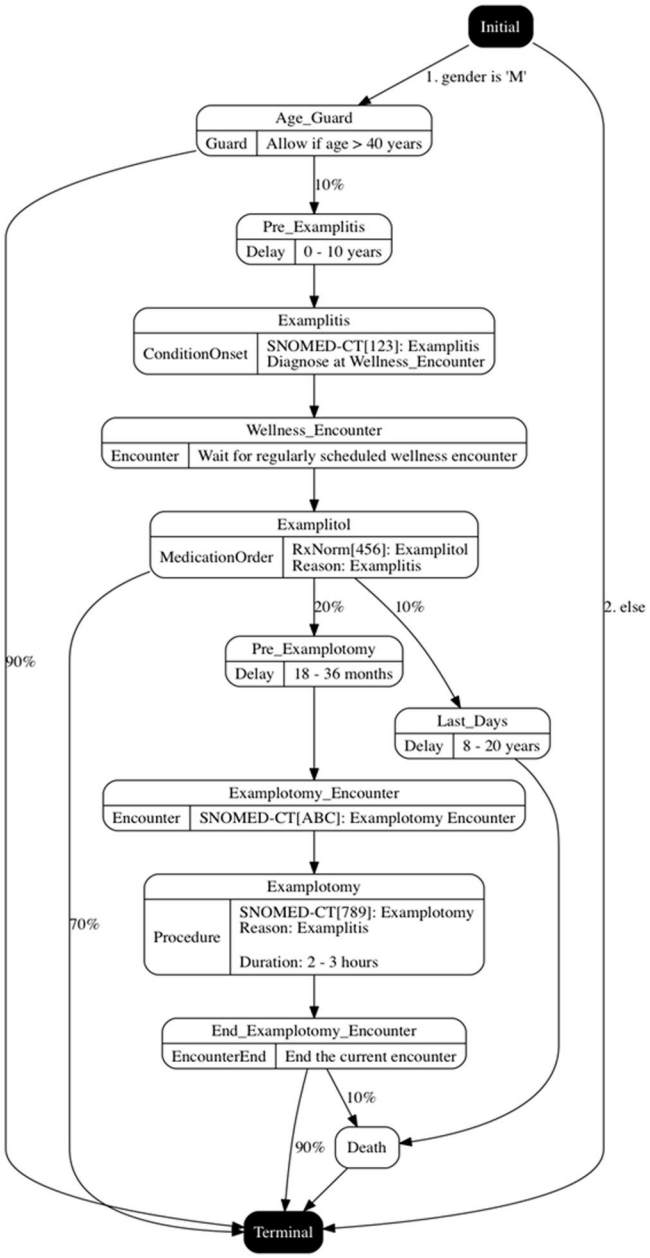
Human input can be inserted at any step of this process, for example, to add or modify information about the disease and its treatments, to manually modify a disease model, or to direct the LLM to make certain changes to the model.

Phase 1: Develop functional requirements

In software engineering, functional requirements are a list of declarative statements that specify the behavior and actions a system must perform to meet user needs. Since models are software systems, it makes sense to organize the information used to create the model in the form of functional requirements. Using an explicit list of functional requirements has several benefits: it anchors the model development in medical reality and enables the LLM in the following phases to focus solely on translating requirements into the final format required for synthetic patient data generation. The requirements also serve as a reference point for reviewing the generated model, enabling iterative refinement and improvement of the model in subsequent steps of the model generation process.

For modeling patient journeys, a wide range of information can be required, including but not limited to:

- Risk factors and demographics
- Screening recommendations
- Incidence and prevalence
- Disease progression
- Symptoms
- Diagnostic criteria
- Treatment options
- Treatment outcomes




```
{
  "name": "Examplitis",
  "remarks": ["Examplitis is a painful condition that
affects only males. Most patients can be cured with
Examplitol or an Examplotomy but some never recover."],
  "states": {
    "Initial": {
      "type": "Initial",
      "conditional_transition": [
        {"condition": {"condition_type": "Gender",
"gender": "M"}, "transition": "Age_Guard"},
        {"transition": "Terminal"}
      ]
    },
    "Age_Guard   ": {
      "type": "Guard",
      "allow": {"condition_type": "Age", "operator": ">",
"quantity": 40, "unit": "years"},
      "distributed_transition": [
        {"distribution": 0.10, "transition":
"Pre_Examplitis"},
        {"distribution": 0.90, "transition": "Terminal"}
      ]
    },
    "Pre_Examplitis": {
      "type": "Delay",
      "range": {"low": 0, "high": 10, "unit": "years"},
      "direct_transition": "Examplitis"
    },
    "Examplitis": {
      "type": "ConditionOnset",
      "target_encounter": "Wellness_Encounter",
      "codes": [{"system": "SNOMED-CT", "code": "123",
"display": "Examplitis"}],
      "direct_transition": "Wellness_Encounter"
    },
    "Wellness_Encounter": {
      "type": "Encounter",
      "wellness": true,
      "direct_transition": "Examplitol"
    },
    "Examplitol": {
      "type": "MedicationOrder",
      "reason": "Examplitis",
      "codes": [{"system": "RxNorm", "code": "456",
"display": "Examplitol"}],
      "distributed_transition": [
        {"distribution": 0.2, "transition":
"Pre_Examplotomy"},
        {"distribution": 0.1, "transition": "Last_Days"},
        {"distribution": 0.7, "transition": "Terminal"}
      ]
    }
(…truncated…)
```


Figure 1. Example of a Synthea disease model. Graphical rendering (L) and source JSON (R) (taken from Ref. 3).

- Follow-up care
- Complications
- Prognosis

Prior to generative AI, the only way to create the functional requirements that shape the model was to engage medical experts on each of these topics or compile relevant information from the literature by hand (typically both). Unfortunately, it can be difficult to recruit and engage physicians with relevant clinical and epidemiologic expertise for a sufficient period of time (which is one of the primary motivations for this research). Now, this process can be streamlined by using an LLM to create at least a first draft of the requirements.

The LLMs tested in this research are general-purpose models whose internal knowledge proved insufficient to produce requirements at the required level of detail. As such, we provided authoritative knowledge sources and did not rely solely on the LLMs' internal knowledge. A sample prompt used to convert the source material into functional requirements is given in the Supplemental Material.

Phase 2: Initial model generation

The second stage involves generating an initial, first-draft model. Again, an LLM is used, this time with a prompt that includes the Phase 1 requirements, and instructs the LLM to create a model in the desired format (eg, Synthea JSON). For validation purposes, the prompt should also ask the LLM to provide traceability back to requirements. The prompt used in this study is provided in the Supplemental Material.

While LLMs typically have extensive knowledge of common programming languages like Python, they cannot be assumed to have expert-level knowledge of niche modeling

frameworks like Synthea. Therefore, information about the target modeling language, including definitions, key words, operators, schemas, and grammatical rules, should be explicitly provided in the LLM's context, along with examples that provide canonical patterns for the LLM to follow.

Phase 3: Model validation

Validation, the third stage of the method, is crucial for ensuring the generation of high-quality synthetic patient data that accurately represents real-world health scenarios. In general, any model can be evaluated on 2 levels:

- **Model syntax (Level 1)**: Does the model follow rules governing the structure of valid statements in the selected modeling language, with structures that conform to schemas and grammatical rules? The output can be binary pass/fail, or quantified in terms of the number of syntactic violations.
- **Model semantics (Level 2)**: Does the model express or implement our intended meaning or function, implementing the functional requirements and any implicit "common sense" rules that are often understood but unstated (for example, the duration of an outpatient encounter is less than 1 day)?

Our implementation of the Level 2 semantic review uses an LLM as the judge.[26] The LLM reviews each requirement and determines if the requirement has been met by the model. It applies a scoring system, as follows:

- 0.0: Requirement not implemented
- 0.25: Implemented incorrectly
- 0.50: Partially implemented
- 0.75: Fully implemented but with minor issues
- 1.00: Fully and correctly implemented

The prompt (see the Supplemental Material) also asks the LLM to explain its reasoning behind the score, and suggest changes or improvements, if necessary.

Phase 4: Progressive refinement via iterative generation

The fourth stage of the process employs a feedback loop that leverages validation results to guide subsequent improvements in the patient journey model. This process has several steps:

1) After model generation, a Level 1 review is conducted, and errors are collected in a narrative form suitable for inclusion in a prompt instructing the LLM to correct its errors. The prompt includes a list of errors, suggestions to correct the errors generated by the LLM, and the model to correct.
2) After one opportunity to correct Level 1 errors, an LLM performs the Level 2 review, returning a score representing how well each requirement is implemented.
3) Targeted requirements: The system next identifies requirements with the lowest scores and selects a limited number of these to be the focus for the next iteration. This approach allows for targeted improvement efforts, focusing on requirements most in need of improvement. This approach allowed the LLM to concentrate on key areas for improvement without being overwhelmed by excessive feedback. As the lowest scores were addressed, attention would automatically shift to higher scoring requirements, thus addressing increasingly nuanced aspects of the model.
4) Continuous validation: After each regeneration step, the new model undergoes another round of review and scoring.
5) Termination: The process continues iteratively until a satisfactory average score across all requirements is achieved or until the improvement between iterations becomes negligible.

Several aspects are worth noting:

1) Optimization objective: The iterative process maximizes the average Level 2 score, which gives equal weight to each requirement.
2) Improvement trajectories: Initial iterations may see rapid improvements as major issues are addressed, followed by a plateau as more nuanced refinements are needed. Monotonic improvement is not guaranteed, since improvements in one requirement may affect the implementation of others.
3) Stochasticity: Because LLM outputs have a degree of randomness, there is no guarantee of convergence to a unique structure.

Example

Hyperthyroidism (HT) was chosen as an example. Synthea currently lacks a model for HT, and treatment and management of HT involves multiple diagnostic pathways, various treatment modalities, and complex decision trees based on patient characteristics and disease etiology,[27] which showcases the potential benefits of our automated approach. Also, the disease and its treatments are relatively standardized, which allowed for objective evaluation of the results.

The model generation algorithm was implemented in Python, using programmatic interfaces to Claude 3.5 Sonnet, GPT-4o, and Gemini 1.5 Pro. The use of proprietary, closed source models reflected our discovery that only the most powerful models were able to generate answers of sufficient detail to be useful for Synthea model generation.

The final models are too lengthy for inclusion in this paper but can be found at https://github.com/syntheticahealth/synthea-llm/tree/main/modules.

Example Phase 1: Functional requirements

The first stage involved prompting one of the LLMs (Claude) to create functional requirements for hyperthyroidism based on diagnosis and treatment guidelines from American Thyroid Association,[28] Society for Endocrinology,[29] and the Journal of the American Medical Association.[27] The resulting 45 requirements (included in the Supplemental Material) covered incidence rates (eg, "Risk of overt Graves' disease in women aged 15-60: 1.35%"), clinical presentation (eg, "Heart palpitations affect 80% of hyperthyroidism cases"), and treatment pathways (eg, "If Graves' Disease is present, antithyroid drugs are first-line therapy except in cases of severe toxic nodular goiter or severe ophthalmopathy, in which case surgical intervention is the preferred first-line therapy"). The requirements were reviewed by 3 practicing physicians employed by MITRE, none experts in hyperthyroidism. One MD questioned the probabilities and

percentages, leading to manual recalculation of incidence rates. In addition, the review led to addition of several narrative sections to the list of requirements: general background on hyperthyroidism, assumptions, and a list of acronyms.

Example Phase 2: Initial model generation

Armed with functional requirements, the next phase was to generate the initial Synthea model. To inform this process, we included the following contextual information:

- Background: A narrative introduction to Synthea, including rules that must be followed to create valid Synthea models.
- Reference Material: Technical documentation describing types of Synthea states and transitions.
- Model Examples: A document containing several examples of Synthea models in JSON format, taken from the existing model set.

The prompt for initial model generation is given in the Supplemental Material. We re-ran this phase several times and noted a wide range of quality (as measured by Level 1 warnings and Level 2 scores) and sophistication (measured by the number of Synthea states). After several repetitions, we selected the best run for each LLM as a starting point for iterative improvement process. An example of an initial model is shown in Figure 2.

Example Phase 3: Validation

Level 1 validation

Level 1 (syntactic) validation focuses on ensuring model integrity. Customized unit tests were developed to check the overall structure and adherence to guidelines in state definitions, transitions, logic, and other elements. The validation included these Synthea-specific checks:

- Path integrity: Ensuring a direct path exists from the Initial state to every other state, maintaining a continuous progression through the model.
- State and attribute validity: Checking if all states are initialized and all attributes have values before they are referenced.
- Transition completeness: Verifying that all states, except the end (Terminal) state, have an output transition, preventing dead-ends in patient trajectories.
- Temporal logic: Confirming realistic start conditions and time progression with respect to disease onset, symptom development, and healthcare interactions.
- Care delivery sequence: Ensuring all clinical actions occur during an encounter, reflecting proper care delivery workflows.
- Probabilistic integrity: Verifying that all probabilities sum to 100% at choice points.

Level 1 validation generates a set of error messages that are fed back to the LLM in Phase 4, ensuring that the models maintain a basic level of integrity, without delving into conformance to the requirements.

Level 2 validation

As described above, Level 2 validation was implemented using an LLM as judge. Prior to full-scale implementation, we ran a series of tests to ensure that the Level 2 reviews could be relied upon to accurately reflect the quality of the model. To evaluate reproducibility, we generated review ratings using the 3 initial models, 3 different LLMs as judges, using 5 runs per model-judge combination. Overall, Claude showed the best precision (lowest standard deviation between runs). Although accuracy was not directly measured, Gemini and Claude delivered similar scores, while ChatGPT's scores were outliers. On this basis, we selected Claude to be the judge for subsequent experiments.

As discussed above, the LLM judge was asked to review each requirement separately (for the exact prompt, see the Supplemental Material). An excerpt from a typical Level 2 review is shown in Table 1.

Example Phase 4: Iterative refinement

The process of iterative refinement was begun after reviewing the first generated model. As discussed above, 10 requirements with lowest scores were identified at each iteration, and the LLM was prompted to improve the implementation of those requirements, using the prompt given in the Supplemental Material.

Figure 3 shows the progress of the overall Level 2 score as a function of the iteration. While not monotonic, the scores trended upwards, reaching 100% for Claude and Gemini. GPT-4o performed relatively poorly in this experiment. However, 100% accuracy only indicates that the LLM judge determined all requirements were met. This does not guarantee that the model actually implemented the requirements correctly or that it was constructed as a human expert would design it.

Discussion

To assess the quality of LLM-generated Synthea models, a Synthea expert provided an in-depth analysis of the models created by Claude and Gemini at the end of the iterative improvement phase. The evaluations revealed that while the LLMs generated runnable models, **they did not produce production-ready models.** The Claude-generated model required improvements in areas such as pairing encounter beginnings and endings, duplication of condition onsets, and elimination of unnecessary states. The Gemini-generated model showed a better foundational understanding of symptom probability chains and incorporation of high-risk patient designation within the model. Still, the Gemini model had more structural problems, including missing transitions, incorrect attribute usage, and references to values within observation states that were not defined. Both models consistently hallucinated formal medical codes. Overall, the Synthea expert concluded that even with these flaws, the resulting models exhibited a high level of sophistication in terms of technical implementation detail and in overall construction, design, and organization.

This proof-of-concept study prioritized technical feasibility of LLM-assisted model generation over production of clinically validated models. While our Level 2 LLM-based validation demonstrated automated iterative improvement capabilities, **Level 2 requirement fulfillment scores should not be interpreted as clinical validation.** The current LLM-generated models would require additional development before clinical validation would be appropriate. Future work should implement systematic clinical expert review processes once technical quality issues are resolved, perhaps leveraging future generations of LLMs.

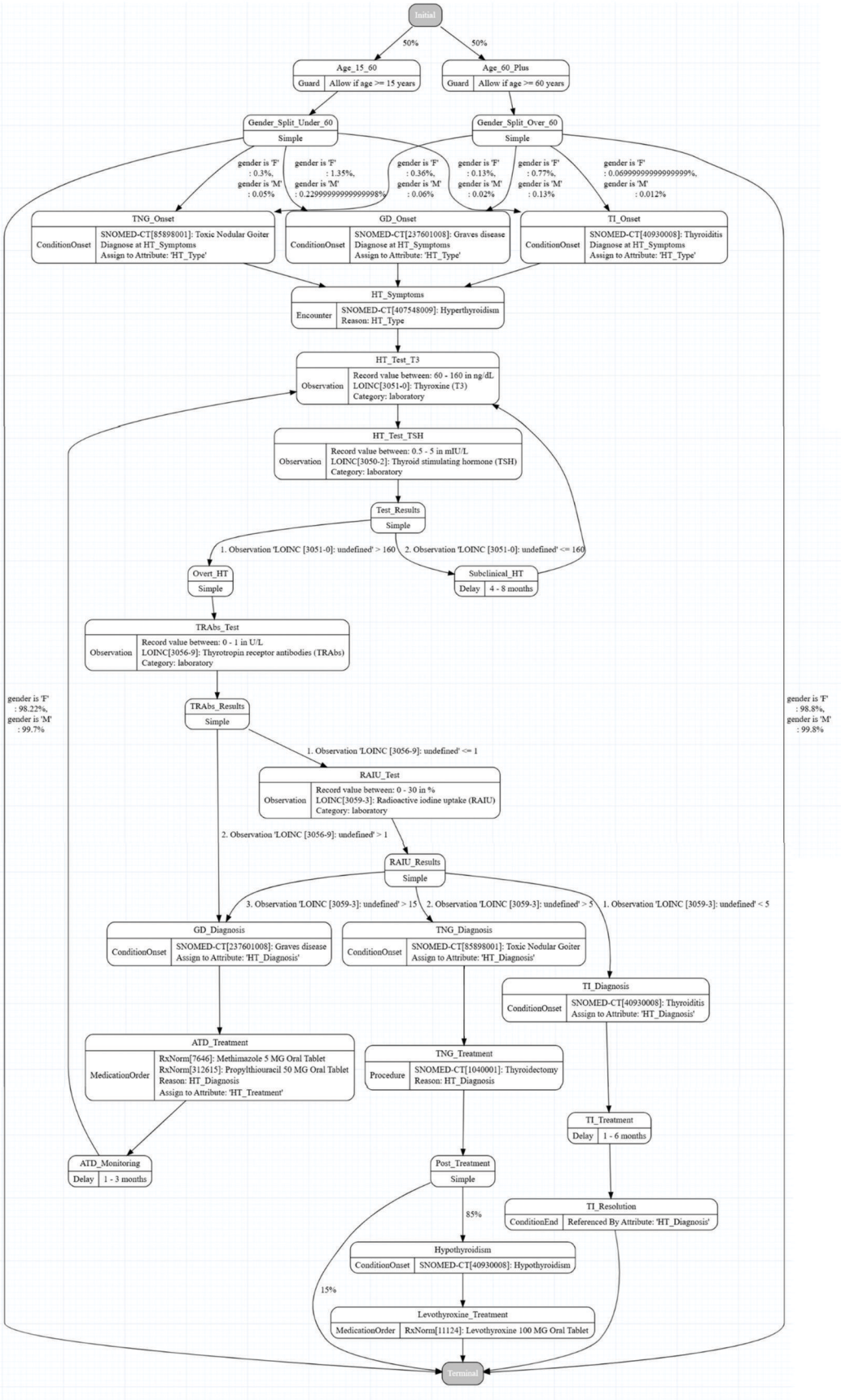


Figure 2. Example of initial model for hyperthyroidism generated by GPT-4o.

Table 1. Excerpt from Level 2 review of initial model produced by GPT-4o.

Req. no.	Requirement	Explanation	Recommended change	Score
23	If initial or follow-up testing reveals overt HT, and palpable thyroid nodules are present, or physiologic signs of GD are unclear, then TRAbs should be measured next.	The model partially implements this requirement. After the Test_Results state determines overt HT, it transitions to the TRAbs_Test state. However, it does not consider the presence of palpable thyroid nodules or unclear physiologic signs of GD.	Modify the Test_Results state to include checks for palpable thyroid nodules and unclear GD signs before transitioning to TRAbs_Test.	0.75
24	However, if initial or follow-up testing reveals overt HT and there are no palpable thyroid nodules and there are clear physiologic signs of GD, TRAbs testing is not necessary, and a diagnosis of GD is confirmed.	The model does not implement this requirement. It always proceeds to TRAbs testing for overt HT cases without considering the absence of thyroid nodules or presence of clear GD signs.	Modify the Test_Results state to include a direct transition to GD_Diagnosis when overt HT is present, there are no palpable thyroid nodules, and there are clear GD signs.	0.00
25	If TRAbs are elevated, a GD diagnosis is confirmed. TRAbs will be elevated in 98% of cases of GD.	The model partially implements this requirement. The TRAbs_Results state transitions to GD_Diagnosis if TRAbs are elevated. However, it does not accurately reflect the 98% probability of elevated TRAbs in GD cases.	Adjust the TRAbs_Test state to ensure that 98% of GD cases have elevated TRAbs. This may require restructuring the model to determine the underlying condition before the test.	0.75
26	Alternatively, if TRAbs are normal, radioactive iodine uptake test (RAIU) should be conducted. However, RAIU is contraindicated in pregnancy and lactation, and a thyroid ultrasound with color-flow Doppler procedure should be substituted.	The model partially implements this requirement. If TRAbs are normal, it proceeds to the RAIU_Test. However, it does not consider contraindications for RAIU or provide an alternative thyroid ultrasound option.	Add a check for pregnancy and lactation before the RAIU_Test state. If contraindicated, add a transition to a new Thyroid_Ultrasound state.	0.50
27	If GD is present, RAIU will reveal diffusely increased uptake in 95% of cases, and then GD diagnosis is confirmed.	The model partially implements this requirement. The RAIU_Results state transitions to GD_Diagnosis if RAIU is high. However, it does not accurately reflect the 95% probability of increased uptake in GD cases.	Adjust the RAIU_Test and RAIU_Results states to ensure that 95% of GD cases have diffusely increased uptake (>15%).	0.75
28	Alternatively, if TNG is present, RAIU will reveal focal areas of increased uptake, and then TNG diagnosis is confirmed. Nodules revealed by ultrasound likewise indicate the presence of TNG.	The model partially implements this requirement. The RAIU_Results state transitions to TNG_Diagnosis if RAIU is in a middle range. However, it does not explicitly model focal areas of increased uptake or consider ultrasound results.	Modify the RAIU_Test and RAIU_Results states to better model focal areas of increased uptake. Add a Thyroid_Ultrasound state that can also lead to TNG_Diagnosis if nodules are present.	0.50
29	Alternatively, if TI is present, RAIU will reveal low or absent uptake, and a diagnosis of TI can be confirmed.	The model implements this requirement. The RAIU_Results state transitions to TI_Diagnosis if RAIU is low.	none	1.00

While the literature lacks a simple, quantitative answer regarding the time it takes to develop a Synthea model, it provides ample evidence of a complex, resource-intensive process.[30] The total effort is contingent on a confluence of factors, including the clinical complexity of the disease, the quality of available data, the maturity of platform documentation, and the blended expertise of the development team. Informally, MITRE experts estimated 40 hours of clinical expert time and 80 or more hours of Synthea expert time are needed to develop a model manually. Using the proposed approach, a single iterative model generation run required only 20-30 minutes of computation. Allowing several hours of human time to validate requirements and inspect the final output, and with several repetitions to refine the results, we speculate that the entire process might be completed in a day or 2, a substantial reduction in development effort.

Limitations

Several other important limitations should be noted:

1) Medical code accuracy: While the LLMs can create structurally valid models with plausible clinical pathways, the codes generated for diagnoses, procedures, and medications are frequently incorrect. A follow-on step to detect and replace hallucinated codes is necessary.

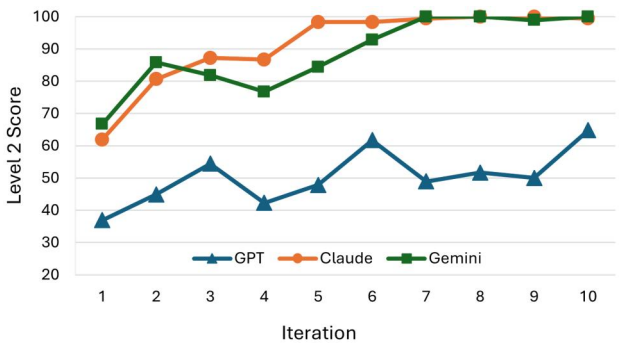


Figure 3. Overall Level 2 (requirement fulfillment) score as a function of iteration.

2) Disease interactions: The current approach treats each disease model in isolation, without considering interactions with other conditions or comorbidities. This limits the realism of the synthetic patient data, as real patients often have multiple interacting conditions that affect treatment decisions and outcomes. This is not a fundamental limitation. In Synthea, comorbidities are handled through parametric interactions between models, for example, adjusting the incidence of one condition or the selection of treatment based on the existence of another disease in the same patient.[31] Extending the current technique to include probabilities conditioned on the entire patient state should be addressed in future research. However, establishing a baseline methodology with a single condition was necessary before tackling more complex interactions.
3) Population-level validation: An important step in model validation is to verify whether the synthetic populations accurately reflect real-world epidemiological patterns. Because of the shortcomings of the models produced here, we did not attempt to validate them at the population level. It should be noted that this issue exists broadly in the domain of synthetic data generation and is not exclusively a problem related to the use of LLM workflows. When the time comes, several methods for evaluating synthetic data quality are available.[32,33]
4) Clinical expertise requirements: While LLMs can assist in model creation, the process still requires significant clinical expertise, particularly in validating the functional requirements and reviewing generated models.
5) Negative requirements: The functional requirements used in the example did not instruct the LLM on what *not* to do when constructing the model. Because of this omission, we observed certain unrealistic behaviors, for example, encounters lasting for many months over entire courses of treatment.
6) State proliferation: Through successive iterations, models tended to grow and become more complex. While this can reflect real-life patient pathways, it may also indicate opportunities for optimization and simplification.

Conclusion

This research demonstrates that LLMs can serve as valuable tools in the development of synthetic health data generators when used within a structured, iterative methodology. Several key findings emerge from this research:

- The progressive refinement methodology proved effective at improving model quality. All 3 tested LLMs showed significant improvement in Level 2 (requirement fulfillment) scores through successive iterations, with requirement fulfillment score approaching 100% for 2 of the LLMs. This suggests that even when initial generations are flawed, systematic feedback and refinement can increase the quality of the results. However, 100% only means that the LLM judge determined that all requirements were met, not that the model *actually* implemented the requirements, nor that the model is clinically accurate.
- The choice of LLM impacted the refinement trajectory and final outcome. While GPT-4o produced consistently valid JSON, it failed to improve Level 2 scores as effectively as Gemini and Claude. We suspect future LLMs will improve all phases, from requirements generation to iterative updates.
- The use of curated knowledge sources significantly improved the quality of the functional requirements compared to relying solely on the LLMs' internal knowledge. This highlights the importance of combining LLM capabilities with authoritative medical sources rather than treating LLMs as standalone knowledge bases.
- The automated review process proved reliable and consistent, particularly when using Claude as the reviewer, with standard deviations in generated scores typically below 5%. This enables tracking of improvements across iterations.

The approach demonstrates value in handling complex disease pathways with multiple treatment options and decision points. The hyperthyroidism case study illustrates how LLMs can manage intricate patient journey logic while maintaining adherence to both medical accuracy and Synthea's structural requirements.

However, the limitations identified indicate that, currently in mid-2024, LLMs should be viewed as assistive tools rather than autonomous model creators. The need for clinical expertise, manual code verification, and careful validation remains critical. Future work should focus on addressing these limitations, particularly in areas such as medical code accuracy, comorbidity handling, and population-level validation.

The methodology introduced here represents a significant step forward in synthetic health data generation, offering a systematic approach to leveraging LLM capabilities to satisfy complex requirements. As LLM technology evolves, this framework provides a foundation for further advancement in synthetic health data generation, ultimately supporting broader access to realistic but non-identifiable patient data for healthcare innovation.

Finally, this research suggests the potential utility of using LLM workflows more broadly within modeling and simulation, beyond the scope of disease modeling.

Acknowledgments

The authors wish to thank Karl Davis and Cynthia Miles of CMS OIT for their support and guidance during this work.

Author contributions

Mark Kramer (Conceptualization, Investigation, Methodology, Software, Supervision, Validation, Writing—original draft, Writing—review & editing), Aanchal Mathur (Conceptualization, Investigation, Methodology, Software, Validation, Writing—original draft, Writing—review & editing), and Caroline Elizabeth Adams (Conceptualization, Investigation, Methodology, Software, Writing—original draft, Writing—review & editing), and Jason Walonoski (Investigation, Visualization, Writing—review & editing)

Supplementary material

Supplementary material is available at *JAMIA Open* online.

Funding

This research was supported by the U.S. Government Centers for Medicare & Medicaid Services (CMS) Office of Information Technology (OIT) under Contract Number 75FCMC18D0047, Task Order 75FCMC23D0004, and is subject to Federal Acquisition Regulation Clause 52.227-14, Rights in Data-General.

Conflicts of interest

The authors are employed by MITRE Corporation, which initiated and maintains Synthea as a non-profit public interest open-source project; none receives direct financial benefit from its use or adoption.

Data availability

The data underlying this article, such as LLM prompts, are available in the Supplemental Material accompanying this article. Synthea model JSON files are available at: https://github.com/synthetichealth/synthea-llm/tree/main/modules. A preliminary version of this article which includes some additional data is available at: https://arxiv.org/abs/2507.21123.

References

1. Gonzales A, Guruswamy G, Smith SR. Synthetic data in health care: a narrative review. *PLoS Digit Health*. 2023;2:e0000082.
2. Walonoski J, Kramer M, Nichols J, et al. Synthea: an approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record. *J Am Med Inform Assoc*. 2018;25:230-238. https://doi.org/10.1093/jamia/ocx079
3. Synthea Generic Model Framework: Complete Example. Accessed September 21, 2025. https://github.com/synthetichealth/synthea/wiki/Generic-Model-Framework%3A-Complete-Example
4. Choi E, Biswal S, Malin B, Duke J, Stewart WF, Sun J. Generating multi-label discrete patient records using generative adversarial networks. *Proc Mach Learn Healthcare Conf*. 2017;68:286-305.
5. Yale A, Dash S, Dutta R, et al. Privacy preserving synthetic health data. ESANN 2019 - European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Apr 2019, Bruges, Belgium. ⟨hal-02160496⟩. Available online at: https://inria.hal.science/hal-02160496v1/document
6. Baowaly MK, Lin CC, Liu CL, Chen KT. Synthesizing electronic health records using improved generative adversarial networks. *J Am Med Inform Assoc*. 2019;26:228-241.
7. Dash S, Gunluk O, Wei D. Boolean decision rules via column generation. *Adv Neural Inf Process Syst*. 2019;32:4655-4665.
8. Nikolentzos G, Vazirgiannis M, Xypolopoulo C, Lingma M, Brandt E. Synthetic electronic health records generated with variational graph autoencoders. *NPJ Digit Med*. 2023;6:83. https://doi.org/10.1038/s41746-023-00822-x
9. Biswal S, Kulas J, Sun H, et al. SLEEPNET: automated sleep staging system via deep learning. 10.48550/arXiv.1707.08262. 2017, preprint: not peer reviewed.
10. Goncalves A, Ray P, Soper B, Stevens J, Coyle L, Sales AP. Generation and evaluation of synthetic patient data. *BMC Med Res Methodol*. 2020;20:108-140.
11. Chen J, Chun D, Patel M, Chiang E, James J. The validity of synthetic clinical data: a validation study of a leading synthetic data generator (Synthea) using clinical quality measures. *BMC Med Inform Decis Mak*. 2021;21:1-10.
12. Xiao C, Choi E, Sun J. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. *J Am Med Inform Assoc*. 2018;25:1419-1428. https://doi.org/10.1093/jamia/ocy068
13. Tucker A, Wang Z, Rotalinti Y, Myles P. Generating high-fidelity synthetic patient data for assessing machine learning healthcare software. *NPJ Digit Med*. 2020;3:147.
14. Martins LNA, Gonçalves FB, Galletti TP. Generation and analysis of synthetic data via Bayesian networks: a robust approach for uncertainty quantification via Bayesian paradigm. arXiv:2402.17915, 2024, preprint: not peer reviewed.
15. Hernán MA, Robins JM. Using big data to emulate a target trial when a randomized trial is not available. *Am J Epidemiol*. 2016;183:758-764.
16. Amad H, Qian Z, Frauen D, Piskorz J, Feuerriegel S, van der Schaar M. Generation and evaluation of synthetic data containing treatments. *International Conference on Learning Representations, ICLR*, 2025.
17. Christiansen J, Campbell A. HealthSim: an agent-based model for simulating health care delivery. 2003. Accessed August 3, 2024. https://www.researchgate.net/publication/237482705_HealthSim_An_Agent-Based_Model_for_Simulating_Health_Care_Delivery
18. Scalfani R, Bhada SV. Health insurance and its impact on the survival rates of breast cancer patients in Synthea. *Risk Manage Insurance Rev*. 2020;23:7-29. https://doi.org/10.1111/rmir.12138
19. Li Y, Zeng C, Zhong J, et al. Leveraging large language model as simulated patients for clinical education. arXiv:2404.13066v2 [cs.CL], 2024, preprint: not peer reviewed. https://arxiv.org/html/2404.13066v2
20. Chen S, Wu M, Zhu K et al. LLM-empowered chatbots for psychiatrist and patient simulation: application and evaluation. arXiv:2305.13614 [cs.CL], 2023, preprint: not peer reviewed. https://arxiv.org/abs/2305.13614
21. Holderried F, Stegemann-Philipps C, Herschbach L, et al. A generative pretrained transformer (GPT)–powered Chatbot as a simulated patient to practice history taking: prospective, mixed methods study. *JMIR Med Educ*. 2024;10:e53961.
22. Rojas E, Munoz-Gama J, Sepúlveda M, Capurro D. Process mining in healthcare: a literature review. *J Biomed Inform*. 2016;61:224-236.
23. Guzzo A, Rullo A, Vocaturo E. Process mining applications in the healthcare domain: a comprehensive review. *WIREs Data Min Knowl Discov*. 2022;12. https://doi.org/10.1002/widm.1442
24. AnyLogic. Accessed August 3, 2024. https://www.anylogic.com/
25. Grefenstette JJ, Brown ST, Rosenfeld R, et al. FRED (a framework for reconstructing epidemic dynamics): an open-source software system for modeling infectious diseases and control strategies using census-based populations. *BMC Public Health*. 2013;13:940. https://doi.org/10.1186/1471-2458-13-940
26. Zheng L, Chiang W-L, Sheng Y, et al. Judging LLM-as-a-judge with MT-bench and Chatbot Arena. arXiv:2306.05685 [cs.CL], 2023, preprint: not peer reviewed. https://arxiv.org/abs/2306.05685

27. Lee SY, Pearce EN. Hyperthyroidism: a review. *JAMA*. 2023;330:1472-1483. https://doi.org/10.1001/jama.2023.19052
28. Hyperthyroidism—StatPearls—NCBI Bookshelf. n.d. Accessed December 10, 2024. https://www.ncbi.nlm.nih.gov/books/NBK537053/
29. Society for Endocrinology. The Latest Guidance in Managing Thyroid Disease. 2022;144. Accessed December 10, 2024. https://www.endocrinology.org/endocrinologist/144-summer-22/features/the-latest-guidance-in-managing-thyroid-disease/
30. Office of the National Coordinator for Health Information Technology, U.S. Department of Health and Human Services. Synthetic health data generation to accelerate Patient-Centered Outcomes Research (PCOR): final report. 2022. Accessed Aug 3 2024. https://www.healthit.gov/sites/default/files/page/2022-03/20220314_Synthetic%20Data%20Final%20Report_508.pdf
31. Agosta J-M, Horton R, Dummitt B, et al. Virtual generalist: modeling co-morbidities in Synthea[TM]. 2022. Accessed August 3, 2024. https://www.healthit.gov/sites/default/files/page/2021-09/Generalistas_Synthetic_Data_Solution.pdf
32. Jordon J, Yoon J, van der Schaar M. PATE-GAN: generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2019. https://openreview.net/pdf?id=S1zk9iRqF7
33. Esteban C, Hyland SL, Rätsch G. Real-valued (medical) time series generation with recurrent conditional GANs. arXiv:1706.02633, 2017, preprint: not peer reviewed.

JAMIA Open, 2026, 9, 1–10
https://doi.org/10.1093/jamiaopen/ooaf123
Research and Applications