# Leveraging Open-Source Large Language Models to Identify Undiagnosed Patients with Rare Genetic Aortopathies

**Pankhuri Singhal, PhD[1*], Zilinghan Li, MS[2*], Ze Yang, MS[3*], Tarak Nandi, PhD[2], Colleen Morse, PT[1], Zachary Rodriguez, PhD[1], Alex Rodriguez, PhD[2], Volodymyr Kindratenko, DSc[3,4], Giorgio Sirugo, MD, PhD[1], Reed E. Pyeritz, MD, PhD[1], Theodore Drivas, MD[1], Ravi Madduri, MS[2,5+], and Anurag Verma, PhD[1+]**

[1]Department of Medicine, Division of Translational Medicine and Human Genetics, University of Pennsylvania Perelman School of Medicine, Philadelphia, PA, 19104, USA
[2]Data Science and Learning Division, Argonne National Laboratory, Lemont, IL, 60439, USA
[3]The Grainger College of Engineering, Department of Electrical and Computer Engineering, University of Illinois Urbana-Champaign, Urbana, IL, 61801, USA
[4]National Center for Supercomputing Applications, University of Illinois Urbana-Champaign, Urbana, IL, 61801, USA
[5]Consortium for Advanced Science and Engineering, The University of Chicago, Chicago, IL, 60637, USA
[*]these authors contributed equally to this work
[+]madduri@anl.gov, Anurag.Verma@pennmedicine.upenn.edu

## ABSTRACT

Rare genetic aortopathies are frequently undiagnosed due to phenotypic heterogeneity, and delayed diagnosis can lead to fatal cardiac outcomes. While genetic testing can enable early proactive interventions, it relies on primary care physicians to recognize a genetic basis for symptoms and then refer patients to clinical genetics. Broad-scale screening methods are needed to identify cases that do not fit an obvious diagnostic pattern. Clinical notes, rich in narrative details, may support the automated flagging of patients for genetic testing. Given the strength of Large Language Models (LLMs) in processing unstructured text, we developed an open-source LLM-enabled genetic testing recommendation pipeline, which leverages retrieval augmented generation (RAG) on curated genetic aortopathy-related corpora to utilize relevant clinical knowledge for identifying patients likely to benefit from genetic testing. The pipeline was validated using 22,510 patient progress notes from 500 individuals (250 cases, 250 controls) in the Penn Medicine BioBank, and successfully categorized 425 out of 499 patients, with one case requiring further clinician evaluation due to incomplete information. The pipeline achieved a patient-level recommendation accuracy of 0.852, precision of 0.889, sensitivity of 0.803, F1-score of 0.844, and F3-score of 0.811. Our LLM-enabled workflow that integrates RAG showed strong performance in recommending genetic testing for patients with rare genetic aortopathies, demonstrating its potential to support undiagnosed patient identification from free-text clinical notes, thereby automating early disease identification and improving patient outcomes.

## Introduction

Rare genetic aortopathies, such as Marfan syndrome and Loeys-Dietz syndrome, are inherited connective tissue disorders that weaken the walls of the aorta, increasing the risk of life-threatening complications like aneurysm or dissection. These conditions are frequently undiagnosed until a major cardiac event, such as aortic dissection, occurs.[1–3] The heterogeneity of symptoms across the affected patient population makes timely diagnosis a challenge.[4] Many adults with genetic disorders are reported to have taken more than five years from experiencing initial symptoms to receiving a genetic disease diagnosis,[3] and these rates are likely even higher for rare genetic conditions. Early screening of at-risk individuals can enable timely medical management and surgical prophylaxis to improve survival rates. Currently, screening methodologies rely on specialist referral to clinical genetics, manual chart review, and primary care physicians' recognition of subtle phenotypic patterns.[5] The lack of adaptable tools for broad-scale flagging of patients across healthcare settings has resulted in the underdiagnosis of affected patients.

Free-text clinical notes contain observations and narratives regarding a patient's health history that can provide evidence to support recommendations for genetic testing. Large language models (LLMs) have shown promise in extracting clinically relevant insights from these unstructured notes.[6–9] Additionally, unlike encoder-only models such as BERT and Med-BERT,[10,11]

which are suitable for classification-only tasks, LLMs offer generative capabilities that enable the generation of reasoning for their outputs, potentially enhancing clinical decision-making. While proprietary models like OpenAI's GPT series demonstrate strong performance across various domains, including medical question answering, their limited adaptability, generalizability, and deployability pose challenges for integration into healthcare systems. Critically, these proprietary models often require data to be sent outside healthcare system firewalls, creating significant privacy concerns in sensitive healthcare environments. Open-source LLMs, when tailored with domain-specific medical knowledge and optimized for local deployment, offer a compelling alternative that supports both scalability and data privacy. This is particularly crucial for rare disease screening in closed and secure clinical infrastructure focused on preserving patient privacy and ensuring computational efficiency.

In this work, we developed an open-source LLM-enabled recommendation pipeline to refer individuals at risk for genetic aortopathy to genetic testing by providing clinical reasoning based on an analysis of the patient's clinical notes. Our pipeline utilized retrieval augmented generation (RAG)[12] technique to leverage additional information within relevant clinical guidelines and disease-specific literature for more accurate recommendations on less-confident clinical notes. By developing a generalizable, end-to-end pipeline scaled for different compute requirements, and customizable for any other disease domain, we demonstrated how AI-driven methodologies can bridge the gap between prognostics and improved patient outcomes.

## Results

### Evaluation framework and model configuration

We evaluated our pipeline on a dataset of 500 patients (250 cases and 250 controls) with 22,510 clinical notes in total within the Penn Medicine's health system. Each patient's record comprised a variable number of notes, reflecting diverse clinical patterns across individuals. The primary objective was to minimize missed diagnoses (false negatives), a particularly critical task in rare disease detection. Thus, we selected the F3-score as our primary metric to evaluate performance, prioritizing sensitivity over precision. Llama 3.1-8B-Instruct[13] was used as the base LLM for the recommendation pipeline. All experiments were conducted using an LLM temperature of 0.7, selected to balance response flexibility and reliability. Additionally, we evaluated the LLM's recommendation confidence based on the probabilities of the LLM outputs. If the confidence for a given clinical note fell below a threshold of 0.5, a RAG module was automatically triggered to retrieve relevant clinical information from curated aortopathy-related literature, augment the additional information with the clinical notes, and generate a refined prediction.

### Note-level performance

Given the temporality of clinical notes data, the pipeline first generates an evaluation for each note, accounting for notes that may not be relevant to rare disease symptoms. At the note level, the pipeline achieved an accuracy of 0.753, sensitivity of 0.630, precision of 0.242, F1-score of 0.350, and F3-score of 0.543 (Figure 1a-b). 500 patients were comprised of 22,510 notes, of which 20,720 (92.05%) successfully generated valid recommendations, while 1,790 (7.95%) failed to generate the output in the desired format for result parsing and were dropped from further analysis. Note-level performance showed relatively high false positive and false negative rates, which were expected as isolated notes may lack comprehensive diagnostic context, and not every note contains symptoms of an aortopathy. No preprocessing filters were used when selecting notes, and all notes were utilized; thus, we expect some of them to be irrelevant or sparse. Diversity in note relevance may arise depending on the clinician's style of writing clinical notes, preset templates offered in the physician portal may vary by department or health system, and some notes may be autogenerated by an administrative action, such as ordering a lab test, without inclusion of other relevant clinical information pertaining to the patient.

### Patient-level performance

A final recommendation for each patient was determined by aggregating predictions across all notes per patient using a consensus-based strategy: if more than 50% of a patient's clinical notes are classified as recommended for genetic testing, the final output is set to recommended. At the patient level, the pipeline achieved an accuracy of 0.852, sensitivity of 0.803, precision of 0.889, F1-score of 0.844, and F3-score of 0.811 (Figure 1a-b). Only one out of 500 individuals (0.2%) failed to receive a prediction because that individual had only one incomplete and uninformative note, which did not produce a valid recommendation output in the desired format. 200 patients were correctly determined by the model to require genetic testing (true positives), 25 were incorrectly thought to need testing when they did not (false positives), 225 were correctly identified as not needing testing (true negatives), and 49 were incorrectly classified as not needing testing when in fact they did (false negatives). Total end-to-end processing time for all 22,510 patient notes on Databricks was 18 hours (Databricks cluster specs: 15.4 LTS, standard NC24ads A100 v4 220 GB, 1 GPU).

Patient-level aggregation of notes significantly improves the overall sensitivity and precision of predictions. The increase in sensitivity from 0.630 (note-level) to 0.803 (patient-level) suggests that aggregating multiple notes enhances diagnostic robustness by mitigating deviations in individual note predictions. This improvement aligns with clinical decision-making processes, where a diagnosis is rarely based on a single clinical entry, which could be incomplete or less informative, but

rather on a holistic assessment of multiple records. Furthermore, the F3-score in patient-level evaluation (0.811) highlights the pipeline's strong performance in prioritizing sensitivity while maintaining diagnostic confidence.

The confidence-based RAG-rerun in our pipeline was motivated by the output probability distribution of the base LLM's recommendations on a randomly sampled development dataset (50 cases and 50 controls), where misclassifications tend to have a wider probability distribution range (Figure 1c). To address this, we aimed to provide additional relevant information from aortopathy-related literature to help the model correct these errors via RAG. We found that a confidence threshold of 0.5 yields optimal diagnostic performance (Figure 1d), as RAG enhances the LLM's ability to generate more accurate recommendations for those low-confidence notes. However, applying RAG to high-confidence notes leads to performance degradation. This is likely because the additional information from RAG makes it more challenging for smaller models, such as Llama 3.1-8B-Instruct, to focus on the most relevant clinical details, ultimately hindering the decision-making process.

### Interpretability analysis

Model interpretability is essential in AI-based medical applications as it helps build trust and ensures that the outputs are grounded in meaningful clinical evidence. We built an interpretability pipeline that leverages Captum's LLM attribution algorithms to enhance the transparency of model predictions.[14] Figure 2 provides an overview of the interpretability pipeline on a synthetic case example. Both the LLM inputs, including the system prompt and clinical note, and the generated outputs, comprising the genetic testing recommendation and its supporting rationale, are fed into the attribution algorithms to compute the importance scores for input tokens. Various LLM attribution algorithms are supported to analyze the importance of input tokens to the LLM predictions. For example, perturbation-based attribution algorithms systematically ablate individual input tokens - removing or altering them one at a time - and measure the change in output probability to assign an attribution score that reflects each token's influence. Gradient-based attribution algorithms compute the derivative of the model's output with respect to each input token embedding to assign importance scores to input tokens.[15] These scores are then visualized using a blue gradient for the corresponding input tokens, with darker hues indicating a higher impact. Finally, an LLM-enabled filter is applied to retain only medically relevant terms within the patient note, ensuring that the visualization highlights key clinical features for more informed decision-making. More interpretability examples for both synthetic rare disease cases and controls and the required computing time for such analyses are provided in the Supplementary Note 3.

## Discussion

In this study, we present an open-source LLM-enabled framework that provides clinical recommendations for genetic testing in patients with rare aortopathies. While current approaches rely on proprietary models such as GPTs for simplistic question-answering setups,[16] our method prioritizes portability and transparency in evaluating patient clinical notes while providing a generalizable framework that can be applied to any other disease and integrated in any health system in a privacy-compliant way. Instead of employing RAG in a generic manner without tailoring it to specific clinical scenarios,[17] our method, employing a RAG-based approach with tailored domain knowledge, provided a genetic testing evaluation for each patient with a 0.852 accuracy and 0.811 F3-score on 22,510 patient notes that did not undergo any preprocessing or filtering step. Our framework can be deployed in a general physician setting as a first-line filter to flag patients who could benefit from genetic testing. The traditional model of genetic testing involves the patient first receiving an assessment from their healthcare provider with reasons, a reference to a geneticist, an order for a genetic test, and possibly a reference to a genetic counselor. However, this entire cycle is contingent on the initial provider recognizing the symptoms as being associated with a genetic condition, which are chronically underdiagnosed.[18] Our approach seeks to provide an automated supplemental recommendation system to the primary care physician that can dynamically process a patient's risk as new data from visits is updated.

We tested a variety of parameters in designing this framework, including open-source model types, model parameters such as temperature, prompting text, corpus chunking and indexing strategies, fine-tuning, and RAG optimization. We found that varying any of these parameters slightly might have a large impact on performance. For example, RAG performance varies when patient notes are formatted as their original note text compared to when formatted as an extracted list of clinical terms. The format of the desired response specified in the prompt also matters, i.e., requesting a JSON-formatted output produces more consistent results compared to open-ended questions. Furthermore, providing few-shot notes of both cases and controls in the prompt significantly elevates false positive rates in the performance. For these reasons, we selected a reasonable configuration of parameters evaluated on a subset of notes and then tested our framework on a larger set of test notes to assess performance. While future directions for this work may include exploring larger open-source models with diverse architectures, this study focused on performance with portable small models that could easily be integrated into health system computing environments.

In order to provide accessible and relevant insights to a physician, our diagnostic framework provided reasoning for each genetic testing recommendation. In this study, we were limited in the ability to do a manual review of the 22,510 reasoning outputs; however, we demonstrated the utility of attribution algorithms for the interpretability of the responses. Figure 2 shows an example of the attribution algorithm demonstrating how each word in a patient note influences the output prediction. In

this case, we see relevant symptoms such as "chronic abdominal pain", "intermittent diarrhea", "recurrent hernias", and "fatal ruptured artery" score highly. This approach could help a physician quickly scan a patient's history and notes to identify relevant factors that support a diagnostic recommendation by our algorithm. Additional work is needed to validate the relevance of each reasoning in context. Time and computing resources were a limitation of the interpretation analysis (Supplementary Table 2).

The framework presented in this study is highly adaptable and deployable across diverse healthcare institutions for any given clinical diagnostic or prognostic question, such as screening for signs of neurodegenerative conditions or cardiac adverse event risk, where only the corpus requires tailoring to the domain knowledge, and the prompt must be changed accordingly. We provide a comprehensive foundation for one to optimize and parameterize if desired. By demonstrating a transparent, portable, and domain-adaptable framework for LLM-assisted clinical decision support, this work lays the foundation for scalable and privacy-compliant integration of generative AI into routine care, enabling earlier detection and intervention for rare conditions.

## Methods

We developed an LLM-enabled recommendation pipeline to flag patients who would benefit from genetic testing for rare genetic diseases and provide corresponding reasoning. We utilize techniques such as retrieval augmented generation (RAG) and fine-tuning on a curated genetic aortopathy-related corpus to inject relevant medical knowledge into the model to bolster performance in providing more reasonable and accurate recommendations.

### Patient data

Clinical notes from patients in Penn Medicine's health system between 2019 and 2024 were utilized to develop and test the pipeline. The case cohort pool contained 273 individuals referred by a physician for genetic testing for a connective tissue disorder. In our definition of the case, the outcome of the test (whether positive or negative for a genetic condition) was irrelevant. Rather, the suspicion of having symptom(s) that necessitated a physician's recommendation for testing was the case criterion. For each case, clinical progress notes from all visits within the 12 months prior to the test order date were evaluated. The control cohort pool consisted of 1,200 age-and-sex-matched individuals from the Penn Medicine BioBank[19] who had no prior genetic testing or predicted loss-of-function variants in genetic aortopathy-related genes as per genetic sequencing (Supplementary Table 1). All available progress notes for control individuals were utilized for analysis. The development cohort for parameterization experiments consisted of a randomly sampled set of 50 cases and 50 controls from each pool, comprising 4489 notes. The final testing cohort, which consisted of 250 cases and 250 controls, was also randomly sampled from the pool, composed of 22,510 patient notes. Genetic testing referral status by physicians functioned as our ground truth label across all experiments. Figure 3 provides a detailed overview of the statistics of the selected testing clinical note cohort, which indicates that different patients have different numbers of notes, and different clinical notes have varying lengths.

### Model selection

In the healthcare domain, data privacy and regulatory constraints impose significant limitations on model selection. Medical institutions usually have strict policies governing data access, making reliance on proprietary models like ChatGPT, which are challenging for local deployment and often require external API calls for inference, less viable despite their strong performance. To ensure data privacy and enable on-premise deployment, we opted for open-source models that can be seamlessly deployed within a local secure computing environment without external dependencies. To determine the suitable base model for this study, we evaluated several popular open-source models, including Llama 2-7B-Chat,[20] Mistral-7B-Instruct,[21] Llama 3-8B-Instruct, Llama 3.1-8B-Instruct, and Llama 3.1-70B-Instruct.[13] We compared these models on the development dataset with 50 cases and 50 controls and evaluated their performance at both the note and patient levels. Our analysis revealed that Llama 3 and Llama 3.1 models consistently outperformed the others (Figure 4a-c). In contrast, Llama 2-7B-Chat and Mistral 7B-Instruct-v0.3 exhibited an overly conservative bias, frequently misclassifying patients as positive (Figure 4c). This high false positive rate makes them unsuitable for clinical applications.

In addition to performance on the development dataset, another crucial factor in our model selection was the context window length of the LLM, which refers to the maximum number of tokens an LLM can process before collapsing and generating uncontrollable content. Llama 3.1 models support a 128K-token context window, while Llama 3 models are limited to 8K tokens. In our datasets, some patient notes exceed 8K tokens (Figure 3b), making a longer context window essential for accurate analysis. Additionally, techniques such as RAG used in our pipeline introduce additional tokens to the LLM. The 128K-token context window of Llama 3.1 allows for full integration of the lengthy patient notes as well as the retrieved knowledge.

When comparing 8B and 70B variants of Llama 3.1, Llama 3.1-70B-Instruct generally has stronger performance across many metrics but is significantly less computationally efficient. Inference with the 70B model requires around 140GB of VRAM, typically necessitating a compute node with four to eight GPUs. In contrast, Llama 3.1-8B-Instruct is much more computationally efficient and capable of running on a single 24GB GPU, making it feasible and accessible for a broader range

of medical institutions. Furthermore, at the patient level, the 8B model outperformed the 70B model in sensitivity and F3-score, two important metrics in rare disease identification tasks. Considering these factors, we finally selected Llama 3.1-8B-Instruct as the base LLM in our study to strike a balance between data privacy, computational efficiency, and clinical relevance, ensuring scalability and accessibility for real-world medical usage. Additionally, if resources permit, the base model can be seamlessly upgraded to a larger and more powerful variant within the same pipeline.

### LLM temperature selection

Temperature is a crucial hyperparameter in LLM inference, particularly in medical applications: a lower temperature encourages more deterministic and stable outputs, which is desirable for clinical reasoning, while a slightly higher temperature introduces variability that may enhance flexibility without compromising reliability. Given these considerations, we conducted experiments using two temperature values: 0.3 and 0.7, aiming to assess their impact on pipeline performance. The choice of 0.3 and 0.7 was guided by the need to maintain stability while allowing a degree of flexibility. A temperature lower than 0.3 would result in overly rigid and deterministic responses, potentially limiting the model's ability to generalize across diverse clinical narratives. Conversely, a temperature significantly higher than 0.7 could introduce excessive randomness, leading to inconsistencies in medical reasoning and factual accuracy.

For evaluation, we tested our pipeline on the development dataset. We observed that at the note level, temperature 0.7 outperformed 0.3, while at the patient level, both settings produced comparable results (Figure 4d-e). Beyond performance accuracy, we also analyzed the number of successfully processed notes. We defined a failed note processing instance as any output that did not conform to the expected structured format and was therefore unparseable. The results showed that temperature 0.7 successfully processed 4,080 notes, whereas temperature 0.3 processed 3,991 notes, resulting in an additional note processing ratio of 2.2% for temperature 0.7. We hypothesize that this is due to the model's increased flexibility at 0.7, which allows it to escape rigid output patterns that could contribute to systematic parsing failures. In contrast, at a lower temperature (0.3), the model tends to generate more deterministic and repetitive outputs, making it susceptible to strict formatting constraints and increasing the likelihood of failure when slight variations occur in input structures. The added variability at 0.7 may have helped prevent repeated failure patterns, ultimately improving output consistency across different cases. Given these findings, we selected temperature 0.7 as the final setting for our pipeline. This decision was based on its ability to maintain comparable performance at the patient level while improving processing efficiency by reducing formatting-related failures. This balance between reliability and robustness is critical for deploying LLMs in clinical settings, ensuring both interpretability and operational efficiency in real-world applications.

### Knowledgebase curation

To improve the capability of general-purpose LLMs in aortopathy-related domains for more accurate recommendations, we curated a specialized medical knowledgebase for injecting domain-specific knowledge into the models. This knowledgebase includes full-text PubMed Central papers on genetic aortopathies published within the last 5 years. NCBI's clinical guidelines for genetic diseases, GeneReviews, were also included, in addition to relevant aortopathy-related medical textbooks. We leveraged MinerU,[22] a state-of-the-art PDF parsing tool, to accurately convert PDF files into raw text files. This process resulted in a comprehensive yet compact rare genetic aortopathy knowledge corpus comprising a total of 1,064 PDF documents.

### Retrieval augmented generation

The curated knowledgebase can enhance the performance of LLM through retrieval-augmented generation (RAG). RAG retrieves relevant text chunks from the knowledgebase based on the patient's clinical notes and appends them to the input of the LLM. The additional contextual information helps improve the LLM's ability to generate more accurate predictions. Efficient retrieval in RAG is essential for LLM performance improvement; however, standard text-chunking and indexing strategies often fail to maintain semantic coherence in medical texts.

To address this, we adopted a strategy inspired by MedRAG,[23], which combines semantic-aware chunking for the knowledgebase with multi-retriever fusion during indexing. Standard chunking methods, which split long texts into chunks with a certain number of characters or words, often disrupt the contextual integrity of the split chunks, thus reducing retrieval performance. Therefore, we implemented a semantic double-pass merging (SDPM) approach: first grouping semantically similar sentences, then merging adjacent similar chunks to preserve context and minimize redundancy. In our SDPM, we utilized a similarity threshold of 0.5, a chunk size of 512 tokens, and a minimum of one sentence per chunk. In the second merge pass, we applied a skip window size of one, allowing the algorithm to bypass a single intervening chunk when evaluating semantic similarity. This ensured that each final chunk encapsulates a complete and meaningful unit of information.

For indexing, we adopted a hybrid retrieval system that integrates both lexical and dense retrieval models. BM25 was used for lexical matching,[24] enabling precise keyword-based search, while the domain-specific embedding model MedCPT[25] facilitated semantic retrieval based on the embedding similarity. These two methods were combined using reciprocal rank fusion to balance lexical precision with semantic depth.[26] Additionally, given that clinical notes often exceed the 512-token limit of

the MedCPT embedding model, long clinical notes were segmented into smaller, context-preserving chunks, and each of which was retrieved independently. The retrieved segments were then aggregated to maintain comprehensive and relevant contextual information. We used a token-based chunking strategy for the clinical notes, which splits input text into segments based on a predefined token limit and overlap. This ensures readability and keeps words intact, while also minimizing the number of sub-notes, thereby reducing redundant retrieval. Each sub-note chunk has a fixed length of 512 tokens with a 25-token overlap to maintain contextual continuity across chunks. To improve retrieval performance, we introduce two additional parameters: (1) sub_k, the number of retrieved chunks per sub-note chunk, and (2) total_k, the total number of retrieved chunks across all sub-note chunks. We chose these parameters through experiments on the development cohort with 50 cases and 50 controls. We found that setting sub_k = 4 and total_k = 25 strikes the best balance between sensitivity and precision (Figure 4f).

### Model fine-tuning

Fine-tuning is also considered effective to inject domain-specific knowledge into LLMs.[27] Traditional fine-tuning typically refers to supervised fine-tuning, where labeled question-answer pairs are used to update the LLM for better performance on specific tasks. However, in this task, we required the LLM to generate both genetic testing recommendations and corresponding reasoning to aid physicians in decision-making. Collecting a sufficiently large, expert-annotated notes training set for this purpose is challenging, and furthermore, training the model on notes limits generalizability when evaluating new patient histories and symptoms. In other words, we did not want to limit the model's exposure to only a certain set of patient profiles through training on notes. Therefore, we opted for continual pre-training on the collected unlabeled aortopathy corpus to enhance the model's understanding of genetic aortopathy. We employed Low-Rank Adaptation (LoRA), a parameter-efficient tuning method proven effective in low-data scenarios.[28] However, our tests on the development dataset indicated that such continual pre-training did not improve the LLM's performance in our tasks, likely due to limited data size and catastrophic forgetting. Consequently, fine-tuned models were not utilized in our final pipeline. Further details and explanations are provided in the Supplementary Note 2.

### Pipeline Implementation

The pipeline first produces predictions at the note-level and then the patient-level with two main components: (a) a recommendation for genetic testing ("recommended" or "not recommended"), and (b) a corresponding reasoning for each prediction. The pipeline is composed of the following main steps:

(i) LLM recommendation for each patient note (substeps 1 and 2 in Figure 5);
(ii) Confidence evaluation for the recommendation (substep 3 in Figure 5);
(iii) RAG-based re-inference for low-confidence notes with retrieved relevant domain knowledge (substeps 4, 5, and 6 in Figure 5);
(iv) Consensus-based aggregation of note-level recommendations to generate final patient-level recommendations (substeps 7 and 8 in Figure 5);

In step (i), each clinical note for an individual is passed to the LLM independently for inference to generate predictions composed of two parts as JSON-formatted strings: a binary classification and an explanatory reasoning output. No preprocessing was done on the clinical notes, though the pipeline is modular in that one could pass notes through filters if desired. Given stochasticity in response generation, the number of rerun iterations for a poorly-formatted or null-response note can be set (default = 3). Responses for notes that do not meet the desired format will be dropped after several rerun iterations. To enhance the recommendation reliability, in step (ii), a confidence score is derived from the log probability of the tokens for "recommended" or "not recommended" in the model output. If the probability is below a certain threshold, the model response is considered uncertain, and a RAG-based re-inference will be triggered by incorporating relevant medical knowledge retrieved from the curated database (step iii). This confidence-based RAG-rerun approach was informed by an initial run on the development cohort with 50 cases and 50 controls, which demonstrated that false predictions tend to have lower confidence values, and RAG can improve LLM's performance on low-confidence notes (Figure 1c-d). To derive a final patient-level recommendation, in step (iv), we apply a consensus-based aggregation strategy across all available notes for each patient. If more than 50% of a patient's clinical notes are classified as "recommended", the final output is set to "recommended". This approach reduces the impact of uninformative and incomplete individual notes while preserving sensitivity to relevant diagnostic signals, and mitigates noise introduced by clinical visits for concerns unrelated to a potential suspected aortopathy. The final system output includes both the recommendation and a reasoning summary to bolster clinician interpretation.

The pipeline is designed to be adaptable across healthcare systems by adjusting the base model, prompt, retrieval database, and confidence threshold. When deploying on a new task, parameters can be tuned iteratively to optimize performance before full-scale implementation. Additionally, as the pipeline is used, high-quality cases can also be incorporated into the retrieval database, progressively enhancing its performance.

### Prompting

Prompt is crucial for LLM performance as it defines the context, scope, and intent of the output. Our prompt consists of general guidelines outlining the scope of the evaluation, clinical task of recommendation and reasoning, and output format specifications as a JSON-formatted string for easy and automated result parsing (Figure 6).

## Data availability

Patient notes data used in this study will not be made available to anyone due to HIPAA compliance and privacy restrictions for Penn Medicine.

## Code availability

The code for this study will be made available through a public GitHub repository upon publication.

## References


1. Seo, G. H. *et al.* The phenotypic heterogeneity of patients with marfan-related disorders and their variant spectrums. *Medicine* **97**, e10767 (2018).
2. Robertson, E. N., Bannon, P. G. & Jeremy, R. W. Long-term outcomes in heritable thoracic aortic disease. *Front. Cardiovasc. Medicine* **9**, 1009947 (2022).
3. Kim, J. *et al.* Patterns of genetic mutations explored by systematic screening of patients with aortopathy and their family members. *JTCVS open* **15**, 27–37 (2023).
4. Phillips, C. *et al.* Time to diagnosis for a rare disease: managing medical uncertainty. a qualitative study. *Orphanet J. Rare Dis.* **19**, 297 (2024).
5. Faghfoury, H. & Guerin, A. Ensuring timely genetic diagnosis in adults. *CMAJ* **195**, E413–E414 (2023).
6. Clusmann, J. *et al.* The future landscape of large language models in medicine. *Commun. medicine* **3**, 141 (2023).
7. Behari, N. *et al.* A decision-language model (dlm) for dynamic restless multi-armed bandit tasks in public health. *Adv. Neural Inf. Process. Syst.* **37**, 3964–4002 (2024).
8. Thirunavukarasu, A. J. *et al.* Large language models in medicine. *Nat. medicine* **29**, 1930–1940 (2023).
9. Huang, J. *et al.* A critical assessment of using chatgpt for extracting structured data from clinical notes. *NPJ digital medicine* **7**, 106 (2024).
10. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 4171–4186 (2019).
11. Rasmy, L., Xiang, Y., Xie, Z., Tao, C. & Zhi, D. Med-bert: pretrained contextualized embeddings on large-scale structured electronic health records for disease prediction. *NPJ digital medicine* **4**, 86 (2021).
12. Lewis, P. *et al.* Retrieval-augmented generation for knowledge-intensive nlp tasks. *Adv. neural information processing systems* **33**, 9459–9474 (2020).
13. Grattafiori, A. *et al.* The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).
14. Kokhlikyan, N. *et al.* Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896* (2020).
15. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328 (PMLR, 2017).
16. Chimirri, L. *et al.* Consistent performance of gpt-4o in rare disease diagnosis across nine languages and 4967 cases. *medRxiv* (2025).
17. Zelin, C., Chung, W. K., Jeanne, M., Zhang, G. & Weng, C. Rare disease diagnosis using knowledge guided retrieval augmentation for chatgpt. *J. Biomed. Informatics* **157**, 104702 (2024).
18. Pederson, H. J. & Narod, S. A. Commentary: Why is genetic testing underutilized worldwide? the case for hereditary breast cancer. *BJC reports* **2**, 73 (2024).
19. Verma, A. *et al.* The penn medicine biobank: towards a genomics-enabled learning healthcare system to accelerate precision medicine in a diverse population. *J. Pers. Medicine* **12**, 1974 (2022).

**20.** Touvron, H. *et al.* Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288* (2023).

**21.** Jiang, A. Q. *et al.* Mistral 7b. *arXiv preprint arXiv:2310.06825* (2023).

**22.** Wang, B. *et al.* Mineru: An open-source solution for precise document content extraction. *arXiv preprint arXiv:2409.18839* (2024).

**23.** Xiong, G., Jin, Q., Lu, Z. & Zhang, A. Benchmarking retrieval-augmented generation for medicine. In *Findings of the Association for Computational Linguistics ACL 2024*, 6233–6251 (2024).

**24.** Robertson, S., Zaragoza, H. *et al.* The probabilistic relevance framework: Bm25 and beyond. *Foundations Trends Inf. Retr.* **3**, 333–389 (2009).

**25.** Jin, Q. *et al.* Medcpt: Contrastive pre-trained transformers with large-scale pubmed search logs for zero-shot biomedical information retrieval. *Bioinformatics* **39**, btad651 (2023).

**26.** Cormack, G. V., Clarke, C. L. & Buettcher, S. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, 758–759 (2009).

**27.** Ovadia, O., Brief, M., Mishaeli, M. & Elisha, O. Fine-tuning or retrieval? comparing knowledge injection in llms. *arXiv preprint arXiv:2312.05934* (2023).

**28.** Hu, E. J. *et al.* Lora: Low-rank adaptation of large language models. *ICLR* **1**, 3 (2022).

## Acknowledgements

P.S. is supported by Genomic Medicine T32 Training Grant 5T32HG009495-08. This research uses resources of the Argonne Leadership Computing Facility, a U.S. Department of Energy (DOE) Office of Science user facility at Argonne National Laboratory and is based on research supported by the U.S. DOE Office of Science-Advanced Scientific Computing Research Program, under Contract No. DE-AC02-06CH11357. This research also utilizes computing resources provided by the National Artificial Intelligence Research Resource (NAIRR) Pilot, supported by award NAIRR240008.

## Author contributions

P.S., Z.L., and Z.Y. conceptually designed the study, conducted the analysis, and drafted the manuscript. T.N., A.R., V.K., G.S., R.P., T.D., R.M., and A.V. provided guidance and feedback in study design and analysis. P.S., C.M., Z.R., and A.V. had full access to all the data in the study and were responsible for the integrity and accuracy of the data analysis. C.M. and Z.R. deidentified the clinical data and P.S. and A.V. accessed and utilized the data. All authors critically edited, read, and approved the final manuscript. P.S., Z.L., and Z.Y. contributed equally and shared co-first authorship. R.M. and A.V. jointly supervised this work.

## Competing interests

All authors declare no financial or non-financial competing interests.

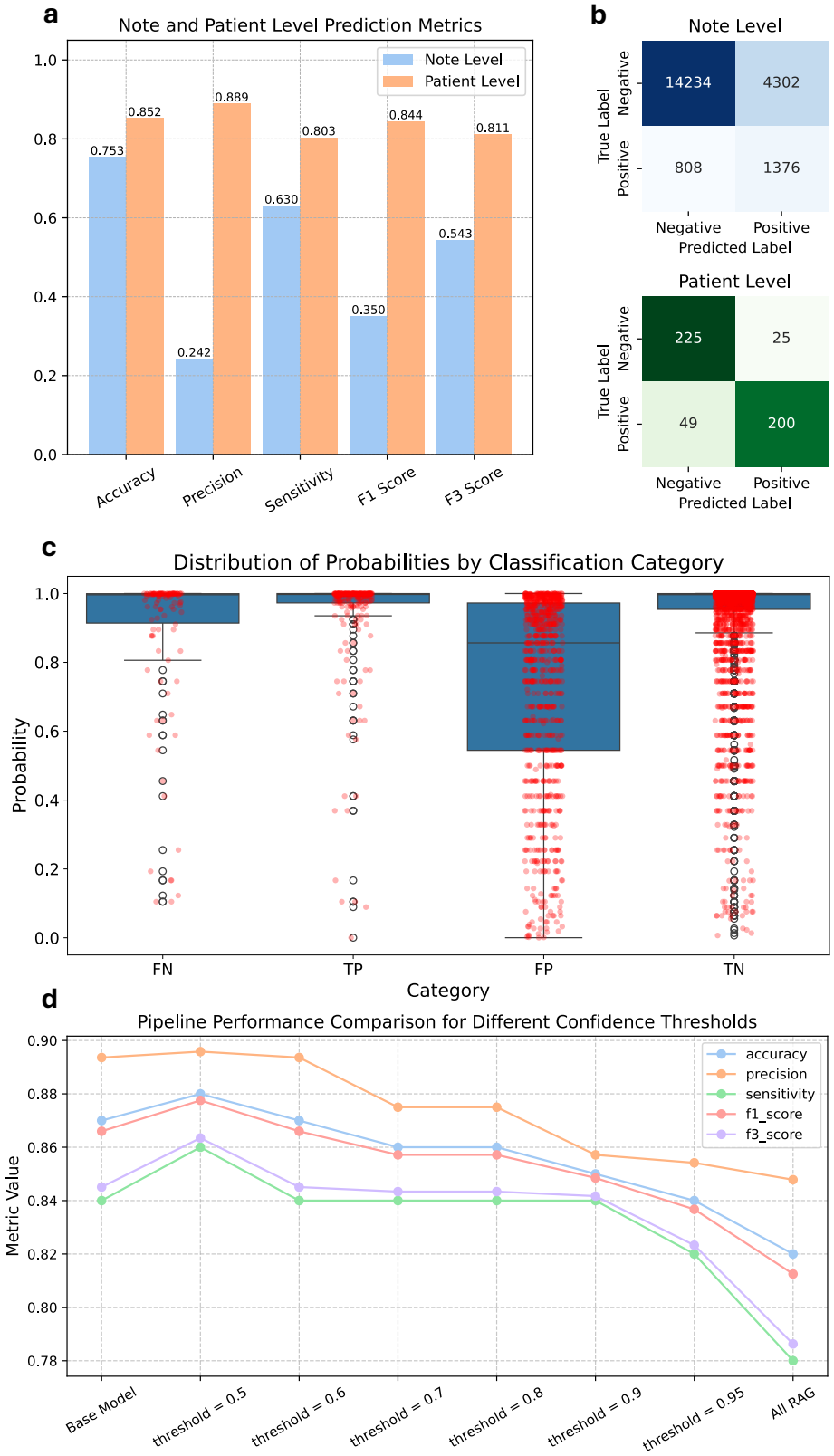


**Figure 1. Overview of the pipeline performance. a** Note and patient-level performance of the pipeline with an optimal confidence threshold of 0.5 on the testing dataset (250 cases and 250 controls). **b** Note and patient-level confusion matrices for the pipeline on the testing dataset. **c** Distribution of the LLM-generated output probabilities across different classification categories on the development dataset (50 cases and 50 controls). **d** Performance comparison of the base model-only, RAG-augmented pipeline with various confidence thresholds, and all-RAG run on the development dataset.

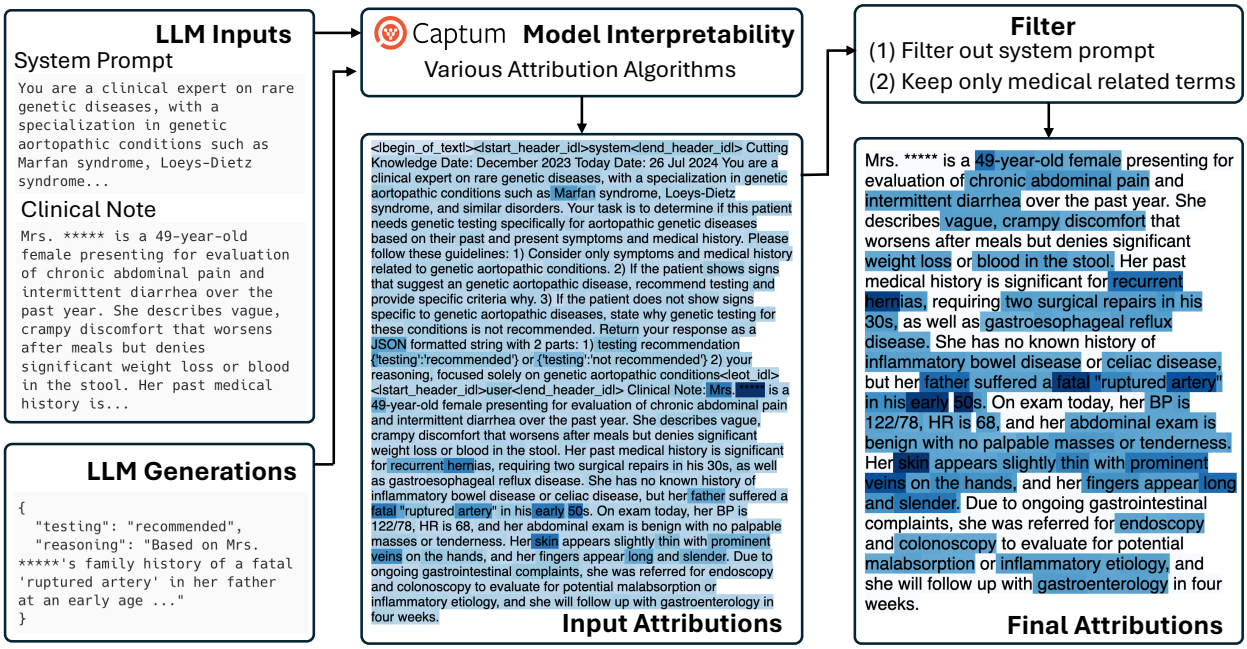


**Figure 2. Overview of the interpretability pipeline on a synthetic case example.** Both the inputs and outputs of the LLM are fed into Captum's various LLM attribution algorithms to compute attribution scores for each input token. A filter is then applied to retain only the scores for medically relevant terms, enhancing visualization clarity.

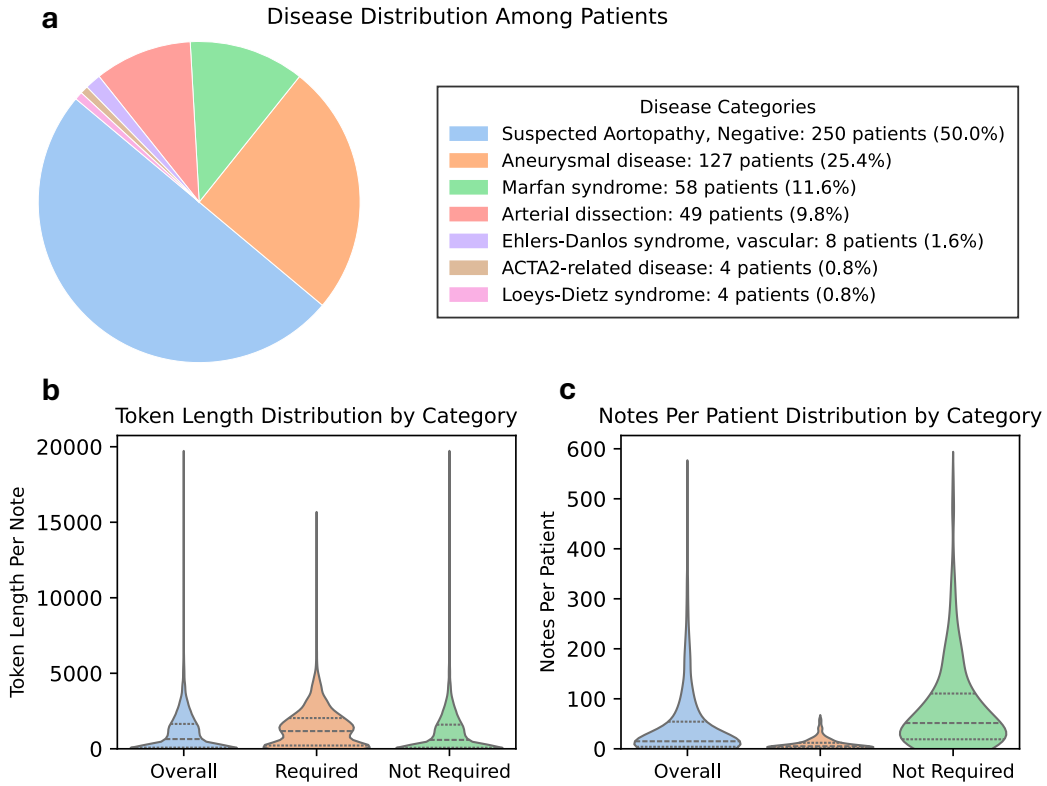


**Figure 3. Clinical note statistics for the randomly sampled testing patient cohort. a** Disease distributions of the 500 individuals within the randomly sampled case/control testing cohort. **b** Token length distributions for the patient notes by category (all patient notes, notes of patient required testing, and notes of patients not required testing). **c** The distributions of the total number of notes per patient by category.

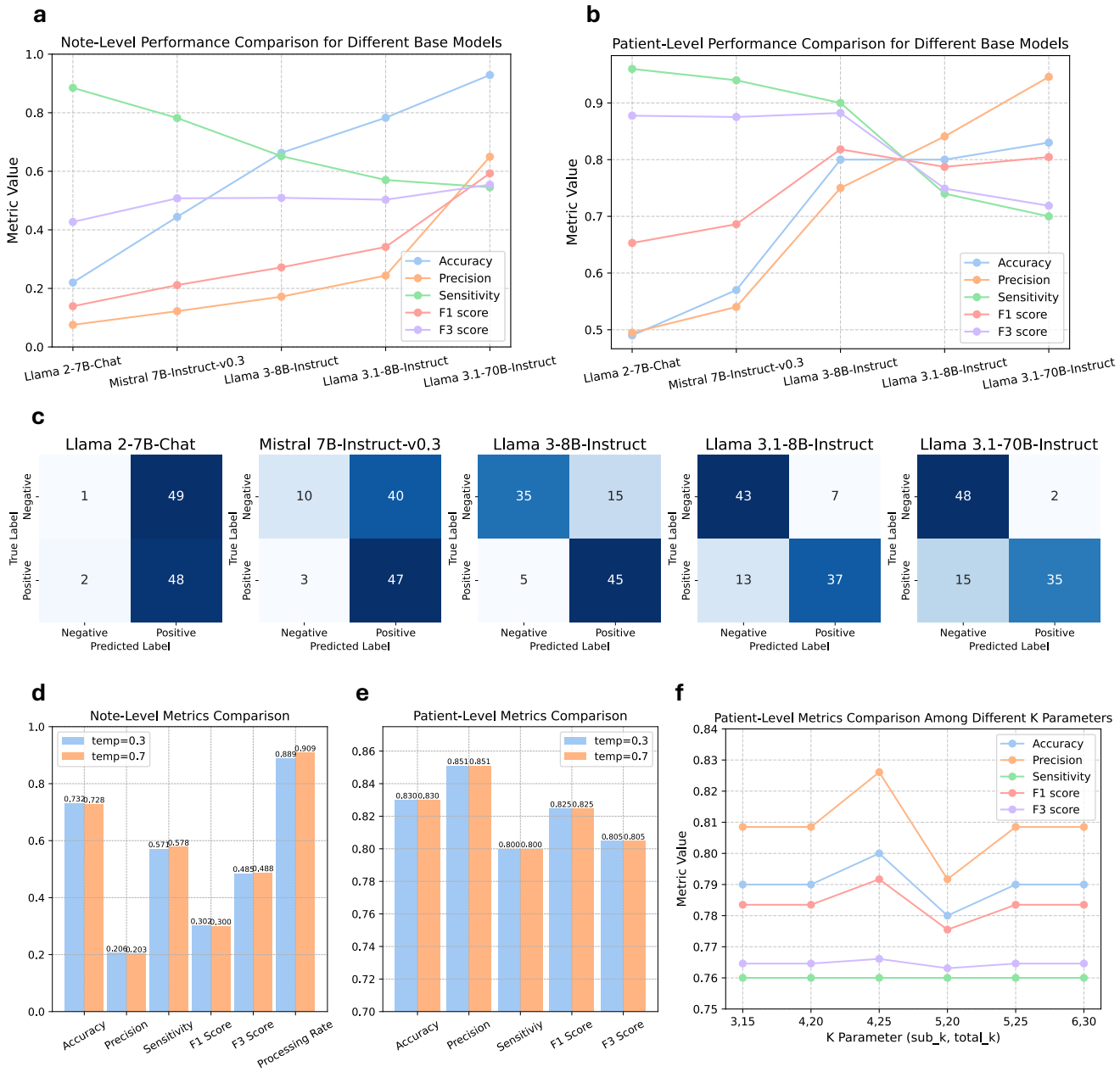


**Figure 4. Performance comparisons for various pipeline settings. a** Note-level performance comparison across multiple LLMs on the development dataset (50 cases and 50 controls). **b** Patient-level performance comparison across multiple LLMs on the development dataset. **c** Patient-level confusion matrices for multiple LLMs on the development dataset. **d** Note-level performance comparison between temperatures 0.3 and 0.7 on the development dataset. **e** Patient-level performance comparison between temperatures 0.3 and 0.7 on the development dataset. **f** Model performance comparison among different K parameters in RAG on the development dataset.

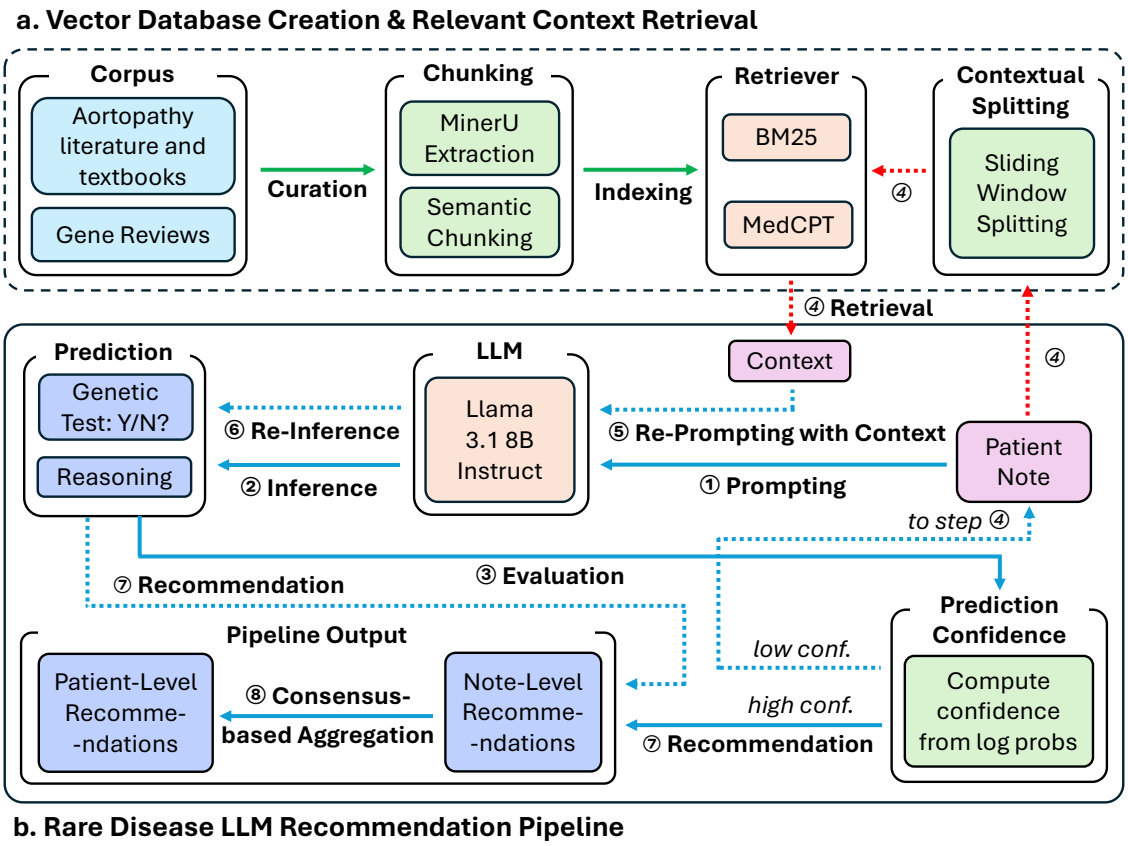


**Figure 5. Overview of the LLM-enabled genetic testing recommendation pipeline for rare genetic aortopathies. a** The raw aortopathy corpora are exacted via MinerU and split into semantically meaningful chunks, which are then indexed to vector databases for further retrieval. **b** Each patient note is passed to a Llama 3.1-8B-Instruct model for generating genetic testing recommendations, with the generation confidence evaluated. Low-confidence notes retrieve relevant contextual information from the vector database for relevant context and re-prompt the LLM for a final note-level recommendation. All note-level recommendations for an individual patient will be evaluated using a consensus-based aggregation approach to generate the final patient-level recommendation for genetic testing.

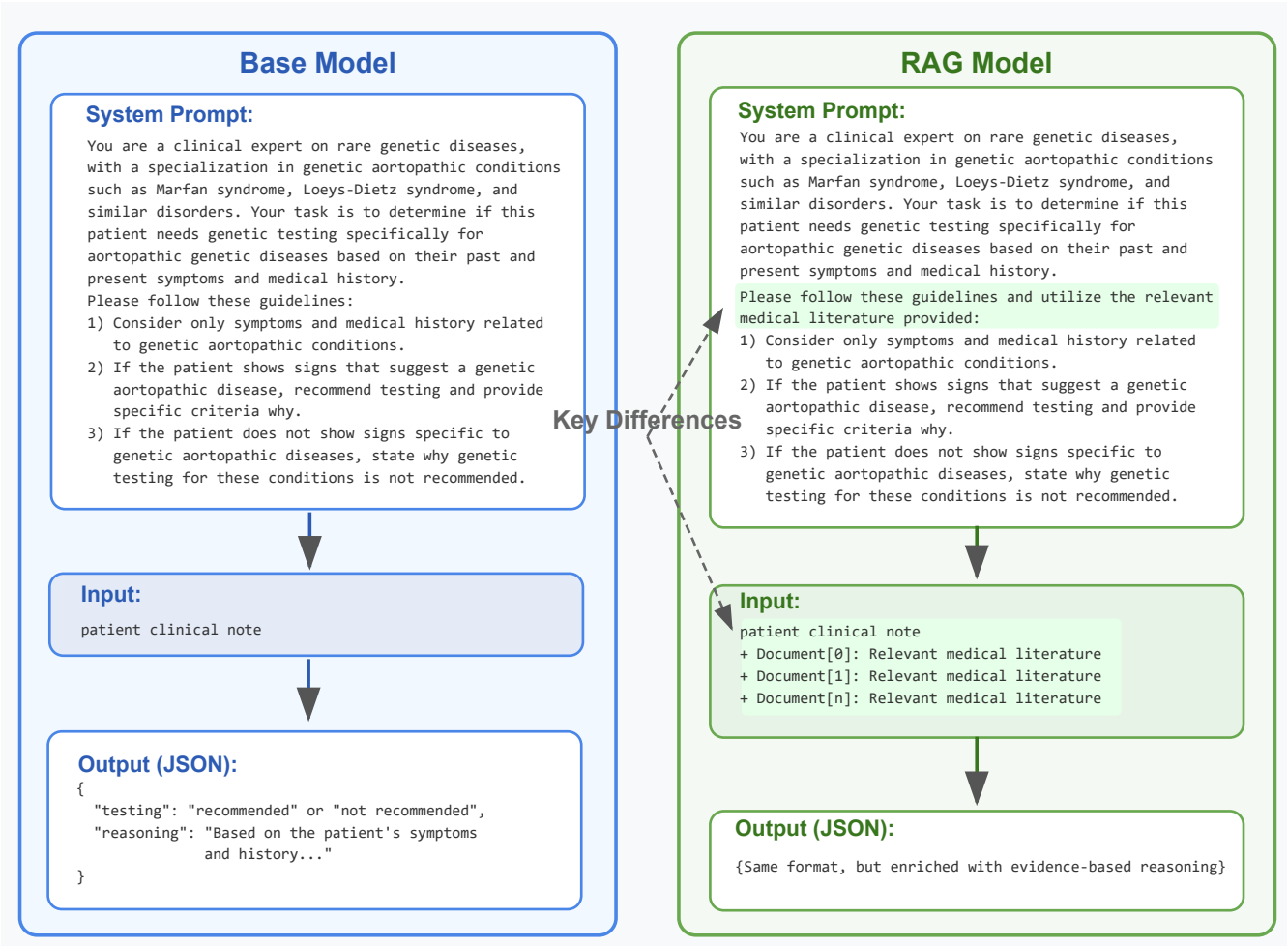


**Figure 6. Prompting schematic.** In the base LLM inference, the left prompt is utilized. Upon inference with RAG on low-confidence notes, the right prompt, together with the retrieved relevant knowledge are provided to the LLM.