Prediction-Augmented Generation for Automatic Diagnosis Tasks

Chan-Yang Ju*, **and Dong-Ho Lee**
Department of Applied Artificial Intelligence, Hanyang University, Korea
{karunogi,dhlee72}@hanyang.ac.kr

Abstract

Most Large language models (LLMs) adopt an autoregressive architecture, predicting the next word token based on the preceding context. While this approach is robust for language generation tasks such as writing and summarization, it has limitations for high-level reasoning tasks, such as prediction and decision-making. To overcome these limitations, we introduce a new method called Prediction-Augmented Generation (PAG). PAG can improve the generation quality and predictive accuracy of large language models in inference-driven tasks by integrating task-specific predictive models as external tools, enabling more structured and precise reasoning. Moreover, our method does not simply copy the inferences of a predictive model, but improves the inference results with knowledge from the large language model to create better predictions. We comprehensively evaluate our proposed method on diverse datasets for automatic diagnosis tasks requiring extensive domain knowledge and advanced reasoning.

1 Introduction

Large language models (LLMs) have gained attention for their ability to perform human's tasks like writing, summarizing, and translation. These models are trained on vast text datasets encompassing books, articles, and web content (Brown, 2020; Radford et al., 2019), enabling them to understand context, syntax, and semantics. LLMs are becoming an important factor in both industry and academia due to its potential to assist and efficientize human work with large amounts of knowledge and language understanding ability.

However, LLMs use an autoregressive structure that predicts the next word token based on the previous tokens, which is less similar to human thinking and relies more on generating natural linguistic flow through memorization (Bender et al., 2021).

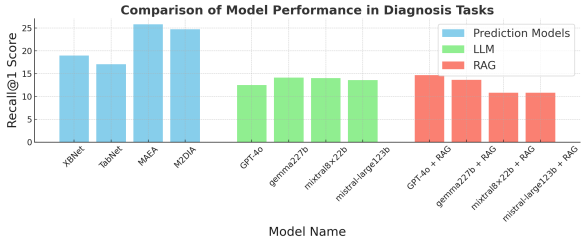

Figure 1: Performance comparison of Predictive models (light blue), single LLMs (light green), and RAG (light red) on the PolyMed dataset (Ju and Lee, 2023).

In addition, unlike prediction-based models that directly output the correct label in inference tasks, LLMs rely on indirectly predicting the word tokens that collectively represent the correct answer, which can degrade their predictive performance overall. This limitation often leads to "hallucinations" where the model generates unsupported text (Jiang et al., 2024b; McKenna et al., 2023). Moreover, even under identical conditions, small changes in the input words can lead to variations in the generated token probabilities and the resulting conclusions, highlighting the system's inherent instability. This problem is present in most language models, making generative models less consistent and reliable in their answers.

Retrieval-augmented generation (RAG) tries to solve this issue by providing rationale and reducing the generation scope to minimize inaccuracies (Lewis et al., 2020; Nakano et al., 2021). However, the tasks expected of LLMs require a huge amount of experience and predictive ability, such as disease diagnosis, judgment, and strategic decisions, beyond knowledge-intensive tasks. LLMs are not expected to perform these advanced tasks adequately due to their limitations (Huang et al., 2023; Ullah et al., 2024), even with RAG. Figure 1 compares the performance of predictive models, generative models, and RAG in disease prediction tasks, showing that predictive models generally per-

* Major in Bio Artificial Intelligence

Findings of the Association for Computational Linguistics: ACL 2025, pages 14225–14246

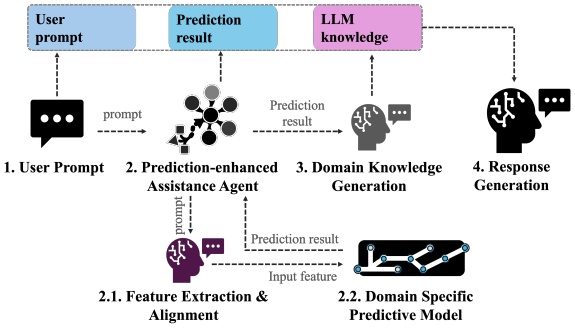

Figure 2: Key Concept of PAG

form better. Despite LLMs being trained on vast general data and medical knowledge, we found that this knowledge was not effectively utilized in the prediction task.

To overcome this limitation, we propose Prediction-Augmented Generation (PAG), illustrated in Figure 2, which can facilitate LLM's prediction task beyond RAG. Inspired by RAG's strategy of supplementing LLMs with external knowledge, we propose a novel method that leverages predictive models to make label predictions and then integrates these results with the knowledge provided by LLMs. Instead of merely copying the predicted labels, our method refines the results by leveraging the LLM's extensive knowledge, leading to improved predictive accuracy. We also propose a novel method for evaluating generative models for traditional classification tasks, overcoming the difficulty of evaluating LLMs on label-centric prediction problems. Our method enables automated evaluation of LLMs even in conventional label prediction tasks by matching generated text with label text. We evaluate PAG in a comprehensive disease diagnosis setting, covering both single and multiple tasks to enable a thorough performance analysis from multiple perspectives.

Our contributions are summarized as follows:

- To our knowledge, our method is the first to introduce the concept of prediction-augmented generation, providing analysis and groundwork for future research.
- Our proposed PAG achieves state-of-the-art performance on the disease diagnosis task, surpassing existing predictive models and LLMs.
- We introduce a novel evaluation method, facilitating a more quantitative and objective evaluation of language models rather than relying solely on abstract assessments.

2 Related Work

External Tools for Enhanced LLMs Early research on LLMs focused on their proficiency in language understanding and generation tasks such as machine translation, summarization, and question and answer (Kenton and Toutanova, 2019; Brown, 2020). While these models demonstrate remarkable abilities in generating contextually relevant text, they also face challenges related to the cost of retraining for updated knowledge and the occurrence of hallucinations in their outputs. To overcome these limitations, previous work has explored the use of external tools to compensate for the weaknesses of LLMs, such as retrieval (Guu et al., 2020; Lewis et al., 2020), extensive external tool support via API calls (Schick et al., 2024), and computational assistance via code executors (Gao et al., 2023). These external tools can extend the context processing capabilities using external modules, thus moving away from the pure language modeling paradigm (Mialon et al., 2023).

Evaluation of LLMs Evaluating the performance of LLMs is an important topic in the natural language processing community. Early methods relied on word-count-based metrics (Papineni et al., 2002; Lin, 2004) that compare surface overlap with reference text, failing to capture deeper meaning. Vector-based similarity measures (Papineni et al., 2002) later improved semantic evaluation by embedding words in continuous spaces, though they remain dependent on the quality of the underlying embedding models.

To evaluate LLMs automatically, some researchers developed multiple-choice question and answering datasets (Clark et al., 2018; Hendrycks et al., 2020; Rein et al., 2023). While this approach can automatically evaluate LLMs, it requires constructing a dedicated dataset for evaluation. In addition, LLMs may still rely on statistical patterns without genuine comprehension or deep reasoning.

In recent years, there has been a shift toward using LLMs themselves as evaluation tools (Chiang and Lee, 2023; Shi et al., 2023). This approach involves using one LLM to evaluate the outputs of another, based on pre-defined criteria. While this method allows for more flexible and nuanced assessments, it introduces new challenges that LLMs tend to produce median scores or neutral judgments (Liu et al., 2023).

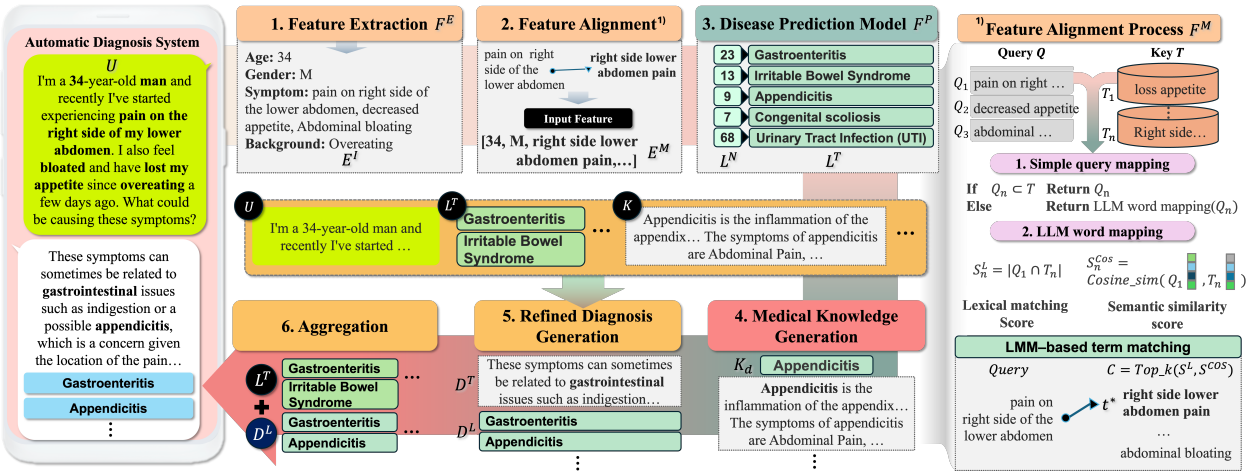

Figure 3: Overall process of Prediction-Augmented Generation for automatic diagnosis system

3 Preliminaries

3.1 Challenges of ADS

Automatic diagnosis systems (ADS) have gained attention for predicting diseases using a patient's self-report, such as age, gender, and symptoms. One challenge in ADS research is the difficulty of collecting diagnostic data, especially for statistically less informed diseases. Existing research has tried to solve this problem by combining independent knowledge inference result with prediction (Ju et al., 2023) or injecting knowledge into neural networks to help improve performance (Xu et al., 2019a; Lin et al., 2021; Zhang et al., 2023; Zhao et al., 2022; Tiwari et al., 2022). Motivated by this challenge, we expected that PAG could contribute to this work by combining LLM's vast knowledge with predictive models to solve the problem. Our method is elaborated in the next section.

3.2 Problem Formulation

Figure 3 illustrates the overall process of PAG for ADS. In the input stage, unstructured patient utterances U are processed by the large language model $\mathcal{F}^{\mathcal{E}}$ for feature extraction to produce structured data E^I. Next, during the feature alignment stage, the extracted data E^I is mapped to standardized features E^M using the medical term dictionary T and the mapping function $\mathcal{F}^{\mathcal{M}}$. In the alignment stage, the mapped features E^M are transformed into numerical representations X for the predictive model $\mathcal{F}^{\mathcal{P}}$. The predictive model then operates on these numerical representations in the prediction stage, predicting disease labels L^N as outputs, which also have corresponding textual labels L^T. Following this, the knowledge generation stage uses large language models to enrich the predicted disease text L^T with relevant medical knowledge K. In the diagnosis stage, the patient utterances U, predicted disease text L^T, and generated knowledge K are integrated to produce an initial diagnosis.

$$P(D^T|U) = LLM(U, L^T, K) \quad (1)$$

$$P(D^L|D^T, U) = LLM(U, L^T, K, D^T) \quad (2)$$

where D^L denotes the disease labels about textual diagnosis D^T. To derive the final diagnosis, an aggregation mechanism combines the predictive model's outputs and the LLM-generated diagnosis:

$$D = \underset{d \in L^N, D^L}{\text{argmax}} \sum_{s \in L^N, D^L} S_s(d) \quad (3)$$

where $S_s(d)$ represents the score assigned to disease d by source of s.

4 Prediction-Augmented Generation

4.1 Feature Extraction and Alignment

The pipeline begins with Feature Extraction (Step 1), where relevant medical concepts are extracted from a patient utterance U according to a predefined schema matching the predictive model's input. LLMs perform this extraction, filling in missing attributes and normalizing the concepts to match the schema. Since the extracted medical concepts are generated in natural language by LLMs, they require transformation into a predefined token suitable for the predictive model. To address this problem, a feature alignment (Step 2) function, denoted as F^M, maps the extracted

medical concepts E^I to a predefined medical term dictionary $T = \{t_1, t_2, ..., t_m\}$. Each of the extracted information is inserted into a query set $Q = \{q_1, q_2, ..., q_n\}$ for word matching. The n is the number of extracted query terms, and m is the number of predefined medical terms in the dictionary. To match the query terms to medical terminology, a two-stage matching approach is employed. First, an exact match search is conducted. If a query term q_n is found in T, it is directly assigned. If no exact match is found, the process moves to the second stage, which ranks term candidates based on lexical-based and semantic-based similarity scores. The lexical-based scoring method measures the overlap between query words and medical terms. The score for a given query-key pair is computed as:

$$\text{Lexical}(q_n, t_m) = |W(q_n) \cap W(t_m)| \quad (4)$$

where $W(q_n)$ and $W(t_m)$ represent the sets of words in a query term and a predefined medical term, respectively. For example, if the query is "pain on right side of the lower abdomen" and the key is "right side lower abdomen pain", the lexical score of this key would be 5. The semantic-based scoring method measures conceptual similarity using word embeddings. Each query q_n and key t_m is transformed into a word embedding representation, denoted as $\mathbf{v}_{q_n}$ and $\mathbf{v}_{t_m}$, respectively. The semantic similarity score is calculated as:

$$\text{Semantic}(q_n, t_m) = \frac{\mathbf{v}_{q_n} \cdot \mathbf{v}_{t_m}}{\|\mathbf{v}_{q_n}\| \|\mathbf{v}_{t_m}\|} \quad (5)$$

This method captures conceptual similarities, such as recognizing "headache" and "pain in the head" as semantically equivalent. The final ranking score for each query-key pair is computed by combining lexical and semantic similarity scores:

$$\text{Score}(q_n, t_m) = \text{R}_L(q_n, t_m) + \text{R}_S(q_n, t_m) \quad (6)$$

where $\text{R}_L(q_n, t_m)$ and $\text{R}_S(q_n, t_m)$ denote the ranking positions of t_m in the lexical and semantic similarity rankings, respectively. The top-ranked term receives a score of k, the second-ranked term is assigned $k-1$, and this pattern continues decrementally until the lowest-ranked term within the top k receives a score of 1. The top-k candidate terms, denoted as $C = \{c_1, c_2, \ldots, c_k\}$, are provided to the LLM through a structured prompt, where the model selects the most contextually appropriate medical term as:

$$t^* = \text{LLM}(q_n, C) \quad (7)$$

where q_n represents the query term, and C is the set of candidate terms. The selected term t^* is then used for word matching, ensuring consistency with the predefined terminology schema. The matched medical information is subsequently utilized as input for the predictive model. To prepare the input X for the predictive model, the selected medical term can be transformed using one-hot encoding, word embeddings, or statistical representations.

4.2 Prediction and Knowledge Integration

Once the input X is prepared, it is fed into the predictive model (Step 3), yielding k prediction results. The predicted numerical disease labels are represented as L^N, while the corresponding textual labels are denoted as L^T. To enhance the explainability and provide context, L^T is further processed by an LLM (Step 4), which generates detailed medical knowledge for each predicted disease. This generated knowledge for a disease is represented as K_d.

Subsequently, the initial diagnosis is formed by integrating the patient's original utterances U, the predictive results L^T, and the generated medical knowledge K_d. This integrated information is provided to the LLM to produce the refined diagnosis text D^T and the corresponding disease label D^L (Step 5).

The final step (Step 6) involves an aggregation mechanism to combine the predictive model's output and the LLM-generated diagnosis. The predicted disease labels L^N and the LLM-generated labels D^L are ranked based on their likelihood. For each source, the most probable disease is assigned the highest score k, the second most probable receives $k-1$, and this continues until the k-th disease, which receives a score of 1.

The score for each disease is computed as:

$$\text{Vote}(d) = \sum_{s \in L^N, D^L} S_s(d) \quad (8)$$

where $S_s(d)$ represents the score assigned to disease d by source s, reflecting the confidence or rank provided by each model. The final diagnosis D is the disease with the highest aggregated score:

$$D = \underset{d \in L^N, D^L}{\text{argmax}} \, \text{Vote}(d) \quad (9)$$

This final diagnosis represents a comprehensive conclusion that integrates insights from both the predictive model and the LLM.

Models	MDD	MZ	DXY	Mean
GPT-4o	87.40	71.69	**96.88**	84.58
GPT-3.5	81.38	65.44	92.66	79.72
gemma2$_{27b}$	85.61	69.19	93.21	82.31
llama3.1$_{8b}$	79.79	67.96	89.21	78.12
llama3.1$_{70b}$	86.21	**75.14**	95.10	82.19
mixtral$_{8\times22b}$	85.96	68.72	94.19	83.11
mistral-large$_{123b}$	**90.12**	74.19	95.06	**86.08**

Table 1: Feature alignment recall (FAR) scores of LMMs across MDD, MZ, and DXY datasets.

5 Experiments

5.1 Experimental Settings

Dataset We evaluate our method on two diagnostic tasks. For single disease prediction, we employ three publicly available datasets—MDD[1], Muzhi (MZ) (Wei et al., 2018), and Dxy (Xu et al., 2019b)—using standard train/test splits that combine both implicit and explicit symptom information. Additionally, the PolyMed dataset (Ju and Lee, 2023) challenges the model under three scenarios: Single test (predicting diseases seen during training), Unseen test (predicting diseases absent from training), and Multi test (predicting multiple diseases simultaneously). Further details are provided in Appendix A.2.

Metrics We evaluate our model using Recall@K, Precision@K, and NDCG (Normalized Discounted Cumulative Gain)@K—which measures ranking quality by assigning lower importance to correct predictions at lower ranks. To aggregate the three PolyMed tests, we use the Weighted arithmetic mean (WAM) score, detailed in Appendix A.3.

Baselines We employ Linear Discriminant Analysis (LDA), TabNet (Arik and Pfister, 2021), XBNet (Sarkar, 2022), M2DIA (Ju et al., 2023), and MAEA (Ju and Lee, 2024) as predictive models. For LLMs, we use closed-source models `gpt-4o-2024-08-06`(GPT-4o) (OpenAI, 2024) and `gpt-3.5-turbo-0125`(GPT-3.5-turbo) (OpenAI, 2023), alongside quantized open-source models from the llama3.1 family (Dubey et al., 2024) (llama3.1:8b, llama3.1:70b), gemma2:27b (Team et al., 2024), mixtral:8x22b (Jiang et al., 2024a), and mistral-large (Mistral AI Team, 2024). Further details on each model and RAG are in Appendix A.1 and Appendix A.4.

[1]https://competitions.codalab.org/competitions/29706

Base LLM	Models	MDD	MZ	DXY
GPT-4o	LDA	67.78	54.92	70.19
	PAG(LDA)	**79.07**	**58.45**	**76.92**
	XBNet	69.87	59.15	67.30
	PAG(XBNet)	**74.89**	**64.08**	**72.11**
	TabNet	78.66	50.00	75.96
	PAG(TabNet)	**82.42**	**62.67**	**76.92**
mixtral $_{8\times22b}$	LDA	67.78	54.92	73.07
	PAG(LDA)	**78.66**	**67.60**	**75.96**
	XBNet	69.03	58.45	69.23
	PAG(XBNet)	**73.22**	**64.78**	**72.11**
	TabNet	76.98	51.40	**76.92**
	PAG(TabNet)	**81.59**	**65.49**	**76.92**
mistral-large $_{123b}$	LDA	60.25	52.81	65.38
	PAG(LDA)	**66.94**	**64.78**	**70.19**
	XBNet	61.92	57.04	64.42
	PAG(XBNet)	**67.78**	**63.38**	**75.96**
	TabNet	69.45	57.74	68.26
	PAG(TabNet)	**71.96**	**66.90**	**72.11**

Table 2: Recall@1 score of PAG and competitors on restored MDD, MZ, and DXY datasets, aligned as in Table 1. The Base LLM is used for data restoration and the PAG process.

5.2 Data Restoration Test

We conducted a data restoration experiment to assess the extraction of structured data from unstructured patient text and its mapping to predefined terms. Virtual patient utterances were generated from structured symptom data in the MDD, MZ, and DXY datasets, and then processed through our alignment system to verify accurate reconstruction, detailed in Appendix A.1. Table 1 shows the Feature alignment recall (FAR) scores for each dataset and model, as defined in Equation 10. The results indicate that the mistral large model performs best, followed by GPT-4o. The performance of each dataset is influenced by the quality of the predefined terminology dictionary. Redundant terms (e.g., "poor appetite" versus "loss of appetite") sometimes caused contextual confusion, hindering accurate reconstruction. Moreover, smaller models tended to exhibit text generation errors, such as typos and inaccurate token outputs, which lowered restoration performance. Detailed generated case are provided in Table 12 and Table 13 in Appendix. Table 2 presents the results of the comparison between the performance of a single predictive model and PAG on the restored data with each large lan-

	Models	MDD			MZ			DXY		
		R@1	R@3	ND@3	R@1	R@3	ND@3	R@1	R@3	ND@3
PM	LDA	75.73	84.93	81.10	67.60	**98.59**	86.78	79.80	97.11	90.34
	gemma2$_{27b}$ + PAG(LDA)	**82.00**	**89.95**	**86.75**	**71.83**	**98.59**	**88.43**	**81.73**	**98.07**	**91.41**
	XBNet	77.82	90.37	85.19	69.71	97.18	86.67	71.15	98.07	87.51
	gemma2$_{27b}$ + PAG(XBNet)	**80.75**	**92.05**	**87.71**	**71.12**	**99.29**	**88.43**	**76.92**	**99.03**	**90.75**
	TabNet	**86.61**	92.46	90.03	72.53	**100**	89.67	81.73	95.19	89.72
	gemma2$_{27b}$ + PAG(TabNet)	**86.61**	**94.97**	**91.39**	**73.94**	**100**	**89.83**	**82.69**	**97.11**	**91.28**
LLM	GPT-4o	39.33	83.26	65.57	38.02	97.18	73.41	31.73	86.53	63.54
	GPT-4o + RAG	65.27	89.95	79.53	66.19	95.07	83.58	66.34	**98.07**	84.60
	GPT-4o + PAG(TabNet)	**86.19**	**95.39**	**91.28**	**69.71**	**99.29**	**87.91**	**79.80**	97.11	**90.22**
	GPT-3.5	37.65	81.17	63.35	34.50	95.77	71.41	31.73	87.50	64.90
	GPT-3.5 + RAG	60.25	83.68	74.15	**66.19**	92.95	82.80	**66.34**	95.19	**83.28**
	GPT-3.5 + PAG(TabNet)	**75.73**	**94.14**	**86.96**	61.26	**98.59**	**84.17**	62.50	**96.15**	82.60
	gemma2$_{27b}$	33.89	71.12	55.79	23.94	79.57	55.72	21.15	55.76	40.47
	gemma2$_{27b}$ + RAG	58.15	82.42	72.21	60.56	95.07	81.59	42.30	90.38	69.74
	gemma2$_{27b}$ + PAG(TabNet)	**86.61**	**94.97**	**91.39**	**73.94**	**100**	**89.83**	**82.69**	**97.11**	**91.28**
	llama3.1$_{8b}$	29.70	74.89	56.24	40.84	92.95	72.43	31.73	81.73	60.12
	llama3.1$_{8b}$ + RAG	58.99	84.51	74.00	58.45	**95.07**	80.17	65.38	93.26	81.71
	llama3.1$_{8b}$ + PAG(TabNet)	**84.51**	**93.72**	**89.94**	**68.30**	92.95	**83.03**	**71.15**	**95.19**	**85.18**
	llama3.1$_{70b}$	41.84	82.42	65.85	41.54	95.77	74.00	43.26	85.57	67.94
	llama3.1$_{70b}$ + RAG	63.59	89.53	78.86	61.26	95.07	82.04	64.42	98.07	83.64
	llama3.1$_{70b}$ + PAG(TabNet)	**83.26**	**94.14**	**89.68**	**70.42**	**98.59**	**88.10**	**71.15**	**99.03**	**87.99**
	mixtral$_{8\times22b}$	36.40	71.96	57.58	39.43	90.84	70.02	17.30	71.15	48.76
	mixtral$_{8\times22b}$ + RAG	63.59	86.61	77.29	64.08	92.95	81.93	68.26	95.19	83.87
	mixtral$_{8\times22b}$ + PAG(TabNet)	**84.51**	**94.56**	**90.58**	**72.53**	**99.29**	**89.05**	**76.92**	**98.07**	**89.51**
	mistral-large$_{123b}$	18.82	59.41	42.84	28.16	83.80	61.88	25.00	68.27	49.90
	mistral-large$_{123b}$ + RAG	62.76	87.86	77.50	63.38	94.36	82.19	71.15	97.11	86.65
	mistral-large$_{123b}$ + PAG(TabNet)	**78.66**	**94.56**	**88.25**	**71.12**	**99.29**	**88.62**	**75.00**	**98.07**	**88.93**

Table 3: Experimental results of Predictive Models (PM), LLMs, and PAG on MDD, Muzhi, and DXY datasets, using Recall(R), Precision(P), and NDCG(ND) metrics. Extended results are presented in Table 10 of the Appendix.

guage model used in Table 1, using the Recall@1 score. We have conducted experiments on the three large language models that performed the highest in Table 1 and found that PAG significantly improves the diagnosis performance. The PAG with GPT-4o and LDA combination recorded the largest performance improvement, and the GPT-4o-based PAG-applied model performed the best overall. The Mistral-large model had the highest data restoration performance in Table 1, but it performed relatively poorly compared to GPT-4o and Mixtral in Table 2. This suggests that data restoration performance does not ensure final prediction performance. Since PAG involves a variety of tasks, including word matching, knowledge generation, and prediction refinement, a model that maintains balanced performance at each stage is likely to be more suitable for our method.

5.3 Single Diagnosis Task

We evaluated the performance of disease prediction using original data from MDD, MZ, and DXY to validate the performance of pure PAG, excluding the impact of data restoration quality. We used the prediction performance of Recall@1 and 3, and the NDCG@3 metrics. The experimental results in Table 3 show that PAG significantly improves prediction performance across all datasets. Although some older and smaller models exhibit performance degradation, PAG still yields an overall enhancement in predictive accuracy.

We evaluated the LLM's performance by aligning its generated diagnostic text with the correct disease labels using our feature alignment process. In the single LLM performance, models with poor prediction performance, such as Mistral-large, showed significant performance improvement after applying PAG. This suggests that PAG can be utilized

	Models	R@1	R@3	R@5	P@1	P@3	P@5	ND@1	ND@3	ND@5
PM	XBNet	18.96	38.47	49.59	**27.36**	17.33	13.07	21.31	33.71	38.75
	GPT-4o + PAG(XBNet)	**20.00**	**45.76**	**57.30**	26.77	**19.92**	**14.84**	**21.87**	**38.02**	**43.19**
	MAEA	**25.78**	47.56	58.82	**34.65**	20.75	15.22	**28.21**	41.90	46.96
	GPT-4o + PAG(MAEA)	24.65	**50.13**	**63.72**	32.07	**21.56**	**16.34**	26.66	**42.64**	**48.73**
	M2DIA	**24.73**	45.75	56.11	**33.19**	19.95	14.55	**27.06**	40.24	44.89
	GPT-4o + PAG(M2DIA)	24.09	**49.18**	**61.38**	31.55	**21.17**	**15.73**	26.12	**41.93**	**47.42**
LLM	GPT-4o	12.50	33.96	43.28	16.01	14.45	11.04	13.45	26.77	30.95
	GPT-4o + RAG	14.68	31.39	42.13	18.55	13.27	10.70	15.66	25.95	30.82
	GPT-4o + PAG(M2DIA)	24.09	**49.18**	**61.38**	31.55	**21.17**	**15.73**	26.12	41.93	**47.42**
	GPT-4o + PAG(M2DIA) w/o K_d	**25.20**	48.20	59.03	**33.55**	20.86	15.22	**27.50**	**41.99**	46.87
	GPT-3.5	12.46	28.50	36.21	15.92	12.14	9.24	13.39	23.33	26.78
	GPT-3.5 + RAG	15.12	29.68	37.49	19.16	12.57	9.53	16.17	25.28	28.78
	GPT-3.5 + PAG(M2DIA)	**22.53**	**46.84**	**57.71**	29.40	**20.26**	**14.88**	24.40	**39.69**	**44.58**
	GPT-3.5 + PAG(M2DIA) w/o K_d	22.51	44.76	56.10	**29.94**	19.45	14.51	**24.57**	38.53	43.62
	gemma2$_{27b}$	14.13	35.17	44.22	18.06	14.96	11.28	15.17	28.14	32.20
	gemma2$_{27b}$ + RAG	13.65	30.37	40.39	17.40	12.84	10.26	14.59	25.10	29.64
	gemma2$_{27b}$ + PAG(M2DIA)	22.93	**49.78**	**62.22**	30.18	**21.36**	**15.97**	24.90	**41.77**	**47.34**
	gemma2$_{27b}$ + PAG(M2DIA) w/o K_d	**23.57**	45.89	56.18	**31.39**	19.98	14.56	**25.73**	39.80	44.43
	llama3.1$_{8b}$	9.10	21.74	30.69	11.58	9.24	7.81	9.77	17.64	21.69
	llama3.1$_{8b}$ + RAG	12.87	29.63	38.68	16.20	12.56	9.84	13.71	24.11	28.21
	llama3.1$_{8b}$ + PAG(M2DIA)	21.53	**44.47**	**57.86**	28.40	**19.27**	**14.91**	23.40	**37.81**	**43.81**
	llama3.1$_{8b}$ + PAG(M2DIA) w/o K_d	**21.98**	43.51	55.17	**29.23**	18.96	14.30	**23.99**	37.50	42.73
	llama3.1$_{70b}$	11.91	27.91	36.91	15.26	11.87	9.41	12.80	22.53	26.58
	llama3.1$_{70b}$ + RAG	14.09	30.82	40.53	17.74	13.03	10.30	15.02	25.44	29.81
	llama3.1$_{70b}$ + PAG(M2DIA)	**21.67**	**47.35**	**60.16**	**28.59**	**20.44**	**15.48**	**23.55**	**39.66**	**45.40**
	llama3.1$_{70b}$ + PAG(M2DIA) w/o K_d	21.19	44.83	57.42	28.25	19.47	14.84	23.12	37.86	43.55
	mixtral$_{8\times22b}$	14.02	32.13	40.61	17.90	13.65	10.37	15.05	26.09	29.88
	mixtral$_{8\times22b}$ + RAG	10.83	24.93	34.00	13.84	10.59	8.66	11.58	20.39	24.56
	mixtral$_{8\times22b}$ + PAG(M2DIA)	22.63	**47.37**	**59.21**	29.46	20.35	**15.26**	24.47	39.99	45.34
	mixtral$_{8\times22b}$ + PAG(M2DIA) w/o K_d	**23.46**	46.97	57.74	**31.50**	**20.42**	14.90	**25.69**	**40.50**	**45.36**
	mistral-large$_{123b}$	13.58	31.85	40.89	17.43	13.55	10.42	14.60	25.77	29.85
	mistral-large$_{123b}$ + RAG	10.83	27.61	36.06	13.83	11.69	9.17	11.58	21.96	25.81
	mistral-large$_{123b}$ + PAG(M2DIA)	22.68	**48.59**	**60.12**	29.35	**20.92**	**15.45**	24.48	40.78	45.97
	mistral-large$_{123b}$ + PAG(M2DIA) w/o K_d	**24.89**	47.96	58.11	**33.13**	20.76	15.00	**27.14**	**41.69**	**46.29**

Table 4: Overall WAM score for Predictive Models (PM), LLMs, and our PAG with three tests of PolyMed. Metrics "R", "P", and "ND" denote Recall, Precision, and NDCG respectively. "w/o K_d" denotes PAG that omits the knowledge generation module. Extended results are presented in Table 11 of the Appendix.

as a robust external tool to complement the performance of LLMs in prediction tasks.

We compared PAG with a Retrieval-augmented generation (RAG) approach and found that PAG outperformed RAG overall. RAG retrieves the top five disease documents by calculating similarity between the user's symptoms embedding and disease information from medical organizations and uses these documents to generate diagnosis text. In the MZ and DXY datasets having less than five disease labels, PAG still outperformed RAG in prediction, even though we provided information for all disease labels to exclude the impact of retrieval performance. This suggests that PAG can improve the accuracy of generation over RAG in the prediction task.

5.4 Multiple Diagnosis Task

Table 4 shows experimental results in multiple diagnosis task of PolyMed. We measured WAM score (Equation 13) to aggregate the single, unseen, and multi test results. The experimental results show that the model with PAG improves the prediction performance in all experiments. Especially in Top-5 predictions, PAG significantly improves scores, demonstrating superior alignment of multiple disease candidates compared to predictive models. Predictive models typically trained for a single label output, so they often fail to produce multiple candidates. By leveraging LLM knowledge, PAG refines predictions and enhances the candidate pool, which is especially beneficial for automatic diagnosis systems offering multiple suspected diseases.

Models	A@1	R@5	P@5	ND@5
M2DIA	12.63	31.04	6.20	22.03
GPT-4o	11.15	**46.09**	**9.21**	29.79
gemma2$_{27b}$	**15.61**	45.91	9.18	**31.94**
mistral-large$_{123b}$	13.94	43.68	8.73	29.56
PAG(GPT-4o,M2DIA)	**16.17**	**43.86**	**8.77**	**30.66**
PAG(GPT-4o,M2DIA) w/o K_d	14.12	35.68	7.13	25.62
PAG(gem2,M2DIA)	**14.86**	**44.79**	**8.95**	**30.77**
PAG(gem2,M2DIA) w/o K_d	12.63	31.22	6.24	22.57
PAG(mistral,M2DIA)	**17.28**	**40.70**	**8.14**	**29.74**
PAG(mistral,M2DIA) w/o K_d	14.31	34.38	6.87	25.12

Table 5: Experiments result of the unseen test dataset. Metrics "R", "P", and "ND" denote Recall, Precision, and NDCG respectively. A@1 denotes the top-1 score for all metrics. In PAG(a,b), a and b are LLMs and predictive models respectively.

The RAG approach uses association rules to rank diseases by counting the overlap between PolyMed knowledge data and patient symptoms, then retrieves the top 20 disease information to generate diagnoses. However, the RAG-based diagnosis generation method showed overall lower performance than PAG. This suggests that in disease diagnosis, inference-based tools are more crucial for performance than knowledge retrieval.

5.5 LLM Knowledge Impact

Table 4 also evaluates the effect of LLM knowledge by comparing the full PAG model to a variant without its knowledge generation component (w/o K_d). The full model outperforms in Top-3 and Top-5 metrics, while the variant shows an advantage in Top-1, suggesting that LLM knowledge enhances the overall quality of the prediction pool. Table 5 focuses on the PolyMed unseen test, where diseases absent from the training data must be inferred using medical knowledge. In this test, the PAG without the knowledge generation, which is more reliant on the predictive model, performs worse. By integrating the predictive model with LLM-derived knowledge, the full PAG model demonstrates enhanced performance, even surpassing the standalone performance of the LLM on some metrics. Moreover, LLM knowledge generation improves both predictive accuracy and explainability. This is especially valuable in applications like disease prediction, where understanding the rationale behind a prediction is critical. Examples of LLM-generated knowledge and prediction cases are provided in Table 14 in the Appendix.

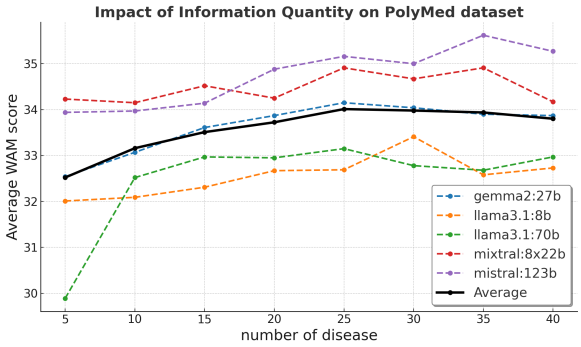

Figure 4: Graph showing PAG performance as the number of disease candidates varies. The dotted lines represent individual open-source LLM (with M2DIA) performance, while the black line shows the average.

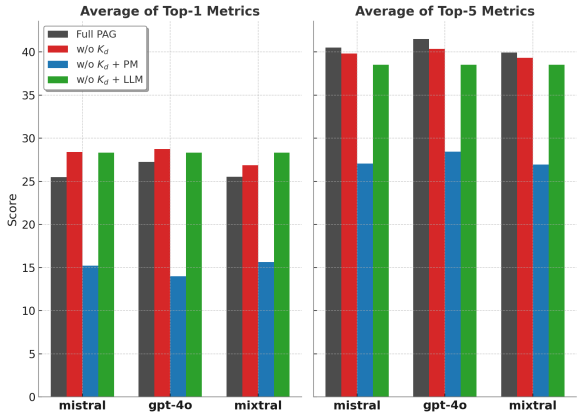

Figure 5: Ablation study presenting average Top-1 and Top-5 metrics for three model groups (Mistral$_{123b}$, GPT-4o, Mixtral$_{8\times22b}$). Bars compare the full PAG model with variants excluding LLM knowledge (K_d), the predictive model (PM), or the large language model (LLM).

5.6 Impact of Information Quantity

We evaluated performance by varying the number of disease predictions fed to the open-source LLM with PAG(M2DIA) using the mean of all metrics' WAM scores on the PolyMed dataset (see Figure 4). We excluded the LLM's knowledge generation of PAG (w/o K_d) to solely assess its ability to process increasing amounts of disease candidates. While performance initially improves, gains plateau or even decline beyond a certain point. Optimal performance is achieved with 25 to 35 disease predictions. This behavior likely reflects the LLM's limited capacity to process excessive information and inherent diagnostic constraints, indicating that the number of predictions in PAG should be treated as a hyperparameter to optimize.

5.7 Ablation Study

We conducted an ablation study to evaluate the contributions of components in the PAG framework. We compare the full PAG model with three ablated variants: (1) removing the knowledge generation module (w/o K_d), (2) using only the LLM without the predictive model and knowledge(w/o K_d, PM), and (3) using only the predictive model without LLM and knoweldge (w/o K_d, LLM). Performance was measured using the average of Recall, Precision, and NDCG at Top-1 and Top-5 ranks.

As shown in Figure 5, the full PAG model achieves better Top-1 performance than the LLM-only variant but is below the predictive model alone and the version without knowledge. This is expected, as predictive models are trained to optimize single-label accuracy, while LLMs are better suited for broader reasoning. In Top-5 metrics, however, full PAG consistently outperforms all other variants. This demonstrates that integrating LLM-generated knowledge enhances the overall quality of candidate predictions.

5.8 Case Study

To illustrate how PAG refines disease prediction through step-wise reasoning, we present a case in Table 6. The input includes structured symptom data indicating a burning sensation behind the breastbone and stomach ache, with the absence of nausea and acid reflux. Although the predictive model identifies several potential conditions, including coronary heart disease and esophagitis, the final diagnosis is refined through knowledge and reasoning powered by large language models.

Using retrieved knowledge about symptom profiles, the LLM identifies esophagitis as the most plausible diagnosis, based on the specificity of the chest-burning symptom and lack of cardiac-related indicators. This demonstrates the ability of PAG to incorporate both statistical predictions and contextual medical reasoning. The refined prediction matches the ground truth, highlighting the value of LLM knowledge in narrowing down ambiguous or overlapping diagnostic candidates. Additionally, our PAG enhances explainability by providing the rationale behind its reasoning, based on the knowledge it has generated about each condition, thereby clarifying the cognitive process that leads to the final diagnosis. These strengths demonstrate the potential of PAG for real-world deployment in medical tasks where trust and reliability are critical.

Component	Content
Meta Data	`{ 'symptoms': { 'Burning sensation behind the breastbone': True, 'Nausea': False, 'Acid reflux': False, 'Stomach ache': True }, 'disease_tag': 'Esophagitis' }`
Target Disease	Esophagitis
PM Prediction	[Coronary heart disease, Esophagitis, Rhinitis, Pneumonia, Enteritis ...]
LLM K_d	Coronary heart disease: [Chest pain, Shortness of breath, Fatigue, Nausea, Lightheadedness, Sweating, Pain in arms or shoulders] Esophagitis: [Difficulty swallowing, Chest pain, Painful swallowing, Heartburn, Acid regurgitation, Sore throat, Hoarseness, Cough, Nausea, Vomiting, Food getting stuck in the esophagus] ...
LLM Reasoning	The symptoms of a burning sensation behind the breastbone and a stomach ache, in the absence of nausea and acid reflux, align most closely with esophagitis. While nausea and acid reflux are common in esophagitis, the central symptom of chest pain or a burning sensation (heartburn) and stomach ache are highly indicative of inflammation or irritation of the esophagus lining, characteristic of esophagitis. Other related conditions, such as coronary heart disease, are less likely due to the absence of other critical symptoms like shortness of breath, pain in arms or shoulders, and lightheadedness. This diagnosis fits the noted symptom pattern without the presence of symptoms more indicative of other diseases on the list.', '1': 'Esophagitis'
Refined Result	Esophagitis

Table 6: Case study illustrating step-wise reasoning from prediction to LLM-based refinement in PAG.

6 Conclusion

We proposed the Prediction-Augmented Generation (PAG) technique, which utilizes predictive models as an external tool to support large language models in inference-driven tasks. We evaluated our method on a wide range of disease diagnosis tasks and found that it can effectively improve not only the diagnosis generation of large language models, but also the prediction performance of predictive models. In particular, we found that the intervention of large language models effectively improves the quality of the disease prediction pool. As a result, PAGs can be utilized as a key technology to effectively combine prediction models and LLMs to improve the reliability of medical diagnosis, and can contribute as a powerful tool for practical medical applications. Our approach is based on recent research trends suggesting that generation through multi-step reasoning can enhance the predictive performance of large language models, and we demonstrated that integrating the output of a predictive model into this process enables more accurate predictions. In future work, we will refine the integration method and evaluate PAG's effectiveness across various prediction tasks.

Limitations

Knowledge Integration. Our PAG effectively combines predictive models with LLM-derived knowledge to enhance predictive capability and overall performance. However, further research is needed to optimize the fusion of prediction results with LLM knowledge. Our experiments show that while PAG excels at generating multiple disease candidates, its ability to pinpoint the single

most probable diagnosis is comparatively weaker. This trade-off indicates a need for improved fusion strategies, and future work should focus on developing knowledge integration methods that can simultaneously optimize both Top-1 and Top-5 performance.

Controllability. Despite the robust language generation capabilities of LLMs, they often struggle with controllability. Our case studies in Table 12 show that LLM-generated text sometimes fails to follow the rules and structure provided in the prompts, and smaller, quantized models can make errors like typos. Addressing these issues could lead to more robust and reliable performance.

Explainability. PAG offers meaningful explainability by incorporating explicit knowledge generation and rationale into its diagnostic output. However, this explainability is derived not from a direct analysis of the model's internal computations, but from conventional post-hoc methods that generate coherent textual explanations. Thus, while PAG's explainability enhances trust and transparency in its predictions, it does not fully elucidate all internal processes. To overcome this limitation, further integration of inherently explainable AI techniques is required.

Acknowledgments

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155885, Artificial Intelligence Convergence Innovation Human Resources Development (Hanyang University ERICA)) and the MSIT(Ministry of Science and ICT), Korea, under the Convergence security core talent training business support program(IITP-2024-RS-2024-00423071) supervised by the IITP(Institute of Information & Communications Technology Planning & Evaluation)

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, page 2623–2631, New York, NY, USA. Association for Computing Machinery.

Sercan Ö Arik and Tomas Pfister. 2021. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6679–6687.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Cheng-Han Chiang and Hung-Yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Pal: Program-aided language models. In *International Conference on Machine Learning*, pages 10764–10799. PMLR.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024a. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Che Jiang, Biqing Qi, Xiangyu Hong, Dayuan Fu, Yang Cheng, Fandong Meng, Mo Yu, Bowen Zhou, and Jie Zhou. 2024b. On large language models' hallucination with regard to known facts. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics:*

Human Language Technologies (Volume 1: Long Papers), pages 1041–1053.

Chan-Yang Ju and Dong-Ho Lee. 2024. Model-agnostic confidence measurement for aggregating multimodal ensemble models in automatic diagnostic systems. *Methods*, 231:103–114.

Chan-Yang Ju, Ji-Sung Park, and Dong-Ho Lee. 2023. M2-dia: Enhancing diagnostic capabilities in imbalanced disease data using multimodal diagnostic ensemble framework. In *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1194–1201. IEEE.

ChanYang Ju and Dongho Lee. 2023. PolyMed: A Medical Dataset Addressing Disease Imbalance for Robust Automatic Diagnosis Systems.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2. Minneapolis, Minnesota.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Shuai Lin, Pan Zhou, Xiaodan Liang, Jianheng Tang, Ruihui Zhao, Ziliang Chen, and Liang Lin. 2021. Graph-evolving meta-learning for low-resource medical dialogue generation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 13362–13370.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: Nlg evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522.

Nick McKenna, Tianyi Li, Liang Cheng, Mohammad Javad Hosseini, Mark Johnson, and Mark Steedman. 2023. Sources of hallucination by large language models on inference tasks. *arXiv preprint arXiv:2305.14552*.

Grégoire Mialon, Roberto Dessì, Maria Lomeli, Christoforos Nalmpantis, Ram Pasunuru, Roberta Raileanu, Baptiste Rozière, Timo Schick, Jane Dwivedi-Yu, Asli Celikyilmaz, et al. 2023. Augmented language models: a survey. *arXiv preprint arXiv:2302.07842*.

Mistral AI Team. 2024. Large enough: Mistral large 2 announcement. Accessed: 19-Sept-2024.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

OpenAI. 2023. Gpt-3.5 turbo. Documentation for GPT-3.5-turbo, optimized for performance and cost-efficiency.

OpenAI. 2024. Hello gpt-4o. Introducing GPT-4o, a variant of GPT-4o optimized for efficiency and cost.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2023. Gpqa: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*.

Tushar Sarkar. 2022. Xbnet: An extremely boosted neural network. *Intelligent Systems with Applications*, 15:200097.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2024. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36.

Xiaoming Shi, Jie Xu, Jinru Ding, Jiali Pang, Sichen Liu, Shuqing Luo, Xingwei Peng, Lu Lu, Haihong Yang, Mingtao Hu, et al. 2023. Llm-mini-cex: Automatic evaluation of large language model for diagnostic conversation. *arXiv preprint arXiv:2308.07635*.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Abhisek Tiwari, Sriparna Saha, and Pushpak Bhattacharyya. 2022. A knowledge infused context driven dialogue agent for disease diagnosis using hierarchical reinforcement learning. *Knowledge-Based Systems*, 242:108292.

Ehsan Ullah, Anil Parwani, Mirza Mansoor Baig, and Rajendra Singh. 2024. Challenges and barriers of using large language models (llm) such as chatgpt for diagnostic medicine with a focus on digital pathology–a recent scoping review. *Diagnostic pathology*, 19(1):43.

Zhongyu Wei, Qianlong Liu, Baolin Peng, Huaixiao Tou, Ting Chen, Xuan-Jing Huang, Kam-Fai Wong, and Xiang Dai. 2018. Task-oriented dialogue system for automatic diagnosis. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 201–207.

Lin Xu, Qixian Zhou, Ke Gong, Xiaodan Liang, Jianheng Tang, and Liang Lin. 2019a. End-to-end knowledge-routed relational dialogue system for automatic diagnosis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7346–7353.

Lin Xu, Qixian Zhou, Ke Gong, Xiaodan Liang, Jianheng Tang, and Liang Lin. 2019b. End-to-end knowledge-routed relational dialogue system for automatic diagnosis. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7346–7353.

Jing Zhang, Xiaokang Zhang, Daniel Zhang-Li, Jifan Yu, Zijun Yao, Zeyao Ma, Yiqi Xu, Haohua Wang, Xiaohan Zhang, Nianyi Lin, et al. 2023. Glm-dialog: Noise-tolerant pre-training for knowledge-grounded dialogue generation. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 5564–5575.

Yu Zhao, Yunxin Li, Yuxiang Wu, Baotian Hu, Qingcai Chen, Xiaolong Wang, Yuxin Ding, and Min Zhang. 2022. Medical dialogue response generation with pivotal information recalling. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 4763–4771.

A Experimental Details

We run all quantized open-source LLMs using the Ollama framework, which provides streamlined deployment and execution of large language models with quantized weights. Experiments are conducted on a workstation equipped with three NVIDIA A6000 GPUs (48 GB each). All LLMs are executed using Ollama's default inference parameters, including decoding strategy, temperature, and token limits.

A.1 Implementations Details

Data Restoration Test. We constructed a patient symptom table by integrating implicit and explicit symptom attributes from the MDD, MZ, and DXY datasets. We generated synthetic patient utterance data using the GPT-4o model based on this consolidated information. To ensure data quality, we employed a dual-validation process. Initially, the GPT-4o model generated patient utterances based on the integrated symptom data. Subsequently, we prompted GPT-4o to review its own output by comparing them with the original data, verifying that no symptom information was omitted. Only the utterances that passed this validation process were used in our experiments. The prompt used for data generation is provided in Table 15, and sample utterances are shown in Table 13. Additionally, OpenAI's `text-embedding-ada-002` was employed for obtaining word embeddings during the Feature Alignment process.

Structured Data Evaluation. To evaluate PAG on the original data of MDD, MZ, DXY, and PolyMed, we replaced the patient utterance information with structured patient data. Using these original data, a predictive model generated 30 disease predictions, which were then used by the LLM to produce related knowledge. Finally, we generate the diagnosis of PAG based on structured patient data, inferences from the predictive model, and the disease knowledge generated by LLM.

LLM Prediction. For the label prediction task of all evaluation datasets, we utilized our proposed feature alignment module. Patient information was provided as a prompt to the LLM, which then generated five candidate disease texts in a structured format. The generated disease texts were compared with a predefined disease label dictionary to retrieve the 15 most similar terms. Consequently, the retrieved candidate terms and the target disease text are combined in the prompt, enabling the LLM to select the most appropriate term. OpenAI's `text-embedding-ada-002` was also used for word embedding.

RAG Prediction. We employed two Retrieval-augmented generation (RAG) methods in this study. The first method is based on a document embedding search and is applied to the MDD, MZ, and DXY datasets, which do not provide explicit disease knowledge. In this approach, we first integrate the implicit and explicit symptom attributes from the datasets to construct a comprehensive representation of patient symptoms, which is then embedded. Concurrently, we collect documents containing disease overviews and symptom information from medical institutions (See Table 8) and embed these as well. By comparing the patient symptom embedding with the document embeddings, we retrieve the five most similar documents and provide them as input to the LLM for diagnosis. The embedding model used is OpenAI's

`text-embedding-ada-002`. The second method employs association rule-based retrieval on the PolyMed dataset, which consists of predefined medical terms for each disease. This approach constructs all combinations of patient symptoms and disease symptoms, calculates the intersection count between these sets to rank the diseases, and then supplies the top 20 disease knowledge entries to the LLM prompt.

A.2 Dataset

Dataset	#Train	#Test	#Diseases	#Symptoms
MDD	1912	223	12	118
MZ	568	142	4	66
DXY	423	104	5	44

Table 7: Statistic information of MDD, MZ, DXY. "#Train" and "#Test" are the total number of train and test data, and "#Diseases" and "#Symptoms" indicate the number of unique diseases and symptoms.

Source
https://www.hopkinsmedicine.org/health/conditions-and-diseases
https://my.clevelandclinic.org/health/diseases
https://www.mayoclinic.org/diseases-conditions
https://my.clevelandclinic.org/health/articles
https://www.msdmanuals.com/professional/ear-nose-and-throat-disorders
https://www.ninds.nih.gov/health-information/disorders
https://www.stanfordchildrens.org/en/topic
https://kidshealth.org/en/parents

Table 8: Knowledge source of MDD, MZ, DXY for RAG.

Dataset	#Data	#Diseases	#Symptoms
Train	3636	57	352
Single test	909	57	248
Unseen test	538	74	290
Multi test	867	53	179
Total	5950	131	433

Table 9: Statistic information of PolyMed-case. "#Data" is the total number of data, and "#Diseases" and "#Symptoms" indicate the number of unique diseases and symptoms.

MDD, MZ, DXY We evaluated disease diagnosis capability in a single-task setting using the public datasets MDD, MZ, and DXY. These datasets were originally designed to assess conversational automatic diagnosis systems, including tasks that extract implicit symptoms from explicit symptoms. However, to measure PAG's diagnostic performance, we employ a symptom set that integrates both explicit and implicit symptoms. Detailed information about each dataset is provided in Table 7.

PolyMed The PolyMed dataset is used to evaluate automatic diagnosis systems from three perspectives.

The first is a Single test to evaluate the disease prediction performance of the model on diseases learned from the Train dataset. This evaluation is equivalent to evaluating the Train-test split data, as is the evaluation of a typical learning-based model.

The second is the Unseen test, which requires the model to make predictions on diseases that it has not learned from the Train data. This task is solved by utilizing disease knowledge data contained in PolyMed data, which is derived from the challenge of supplementing the lack of disease data with knowledge information to overcome the limitation that training data for many common ADS diseases are not available.

The last task is Multi test, which is a test for cases where a patient is diagnosed with more than one suspected disease, and requires prediction for multiple related diseases.

These tasks can evaluate ADS from different angles and allow for a comprehensive assessment. Table 9 provides statistical information about the PolyMed data.

A.3 Metrics

Recall@K Recall@K measures the fraction of correct diseases retrieved in the top K predictions relative to the total number of correct diseases. It is defined as:

$$\text{Recall@K} = \frac{\sum_{i=1}^{K} \mathbf{1}\{\text{disease}_i \text{ is correct}\}}{\text{Total number of correct diseases}} \quad (10)$$

where $\mathbf{1}\{\cdot\}$ is an indicator function. In the data restoration test, "diseases" can be replaced with "symptoms".

Precision@K Precision@K quantifies the proportion of correct diseases among the top K predictions:

$$\text{Precision@K} = \frac{\sum_{i=1}^{K} \mathbf{1}\{\text{disease}_i \text{ is correct}\}}{K} \quad (11)$$

NDCG@K The Normalized Discounted Cumulative Gain at K (NDCG@K) evaluates the ranking quality of predicted diseases. In our setting, every

correct disease is assigned the same relevance score (i.e., $rel_i = 1$). Consequently, the Discounted Cumulative Gain at K (DCG@K) simplifies to:

$$\text{DCG@K} = \sum_{i=1}^{K} \frac{\mathbf{1}\{\text{disease}_i \text{ is correct}\}}{\log_2(i+1)},$$

The Ideal DCG at K (IDCG@K) is computed by placing all correct diseases in the top n positions:

$$\text{IDCG@K} = \sum_{i=1}^{n} \frac{1}{\log_2(i+1)},$$

where $n = \min(K, \text{Total number of correct diseases})$. NDCG@K is then given by:

$$\text{NDCG@K} = \frac{\text{DCG@K}}{\text{IDCG@K}} \tag{12}$$

Weighted Arithmetic Mean To collectively evaluate the three assessments of PolyMed data, we used the Weighted Arithmetic Mean method to calculate a composite score. WAM score can calculated by:

$$\begin{aligned} R &= \{Single, Unseen, Multi\}, \\ w_i &= 1 - \frac{R_i}{\sum_{i \in R} R_i}, \\ WAM &= \frac{\sum_{i \in R} w_i R_i}{\sum_{i \in R} w_i} \end{aligned} \tag{13}$$

where R_i is the score of each $Recall@k$, $Precision@k$, and $NDCG@k$ results for the three test datasets. This weighting system penalizes imbalanced improvements, leading to a decrease in the WAM score if there is an unequal enhancement in the scores. This approach prevents an exclusive focus on specific task optimization to improve model performance.

A.4 Details of baselines

A.4.1 Predictive Models

LDA, XBNet, TabNet For LDA, we used a Singular Value Decomposition solver to fit the data. XBNet (Sarkar, 2022) was trained with the Adam optimizer at a learning rate of 0.01 over 500 epochs using a two-linear-layer architecture, and the best-performing model was selected. TabNet (Arik and Pfister, 2021) was optimized using Optuna (Akiba et al., 2019) with a maximum of 3000 epochs and a batch size of 256; from 1000 training runs, the model yielding the highest performance was chosen.

M2DIA M2DIA (Ju et al., 2023) is a predictive model for the PolyMed task, utilizing a multimodal ensemble approach that combines predictive modeling with knowledge inference techniques to achieve superior diagnostic performance. In our implementation, we adopted the original model's ensemble structure, which integrates six prediction models with their corresponding weights.

MAEA MAEA (Ju and Lee, 2024) is a state-of-the-art model for the PolyMed task that refines the aggregation approach of M2-DIA. It introduces a model-agnostic confidence measurement to quantify the reliability of each modality's contribution and dynamically weight ensemble predictions.

A.4.2 Large Language Models

Llama 3.1 Meta's Llama 3.1 was used in two configurations: an 8B model (ollama's `llama3.1:8b-instruct-fp16`, FB16 quantization, 16GB memory) and a 70B model (ollama's `llama3.1:70b-instruct-q8_0`, Q8 quantization, 75GB memory).

Gemma 2 Gemma 2 is a 27B model from Google, deployed as ollama's `27b-instruct-fp16` with FB16 quantization (54GB memory).

Mixtral Mixtral comprises eight 22B models, implemented as ollama's `mixtral:8x22b` with Q4 quantization (80GB memory).

Mistral-large Mistral-large is a 123B model, provided as ollama's `mistral-large:123b` with Q4 quantization (69GB memory).

Models	MDD			MZ			DXY		
	R@1	R@3	ND@3	R@1	R@3	ND@3	R@1	R@3	ND@3
LDA	75.73	84.93	81.10	67.60	98.59	86.78	79.80	97.11	90.34
gemma2$_{27b}$ + PAG(LDA)	**82.00**	89.95	**86.75**	**71.83**	98.59	**88.43**	**81.73**	98.07	**91.41**
llama3.1$_{70b}$ + PAG(LDA)	70.71	**90.37**	82.46	68.30	**100.00**	87.56	74.03	98.07	88.57
mixtral$_{8\times22b}$ + PAG(LDA)	81.59	**90.37**	**86.75**	67.60	97.18	85.99	78.84	97.11	90.12
mistral-large$_{123b}$ + PAG(LDA)	66.94	89.95	80.75	64.08	97.88	84.76	75.00	**99.03**	89.53
XBNet	77.82	90.37	85.19	69.71	97.18	86.67	71.15	98.07	87.51
gemma2$_{27b}$ + PAG(XBNet)	**80.75**	92.05	**87.71**	**71.12**	**99.29**	**88.43**	76.92	**99.03**	**90.75**
llama3.1$_{70b}$ + PAG(XBNet)	76.56	**93.72**	86.68	68.30	95.07	84.73	66.34	**99.03**	86.09
mixtral$_{8\times22b}$ + PAG(XBNet)	76.56	92.88	86.26	69.01	97.18	86.23	76.92	**99.03**	90.12
mistral-large$_{123b}$ + PAG(XBNet)	66.10	92.46	82.02	66.90	97.18	85.73	**77.88**	98.07	90.12

Table 10: Experimental results of Predictive Models (PM) and PAG on MDD, Muzhi, and DXY datasets, using Recall(R), Precision(P), and NDCG(ND) metrics.

Models	R@1	R@3	R@5	P@1	P@3	P@5	ND@1	ND@3	ND@5
LDA	20.26	39.35	46.62	27.30	17.34	12.21	22.19	34.27	37.62
gemma2$_{27b}$ + PAG(LDA)	**22.36**	**42.78**	**51.87**	**29.63**	**18.59**	**13.42**	**24.33**	**37.22**	**41.33**
llama3.1$_{8b}$ + PAG(LDA)	18.88	37.88	47.95	25.17	16.62	12.52	20.59	32.58	37.16
llama3.1$_{70b}$ + PAG(LDA)	19.43	41.19	50.97	25.88	18.01	13.26	21.18	34.85	39.32
mixtral$_{8\times22b}$ + PAG(LDA)	22.04	41.61	50.15	29.11	18.16	13.06	23.95	36.26	40.13
mistral-large$_{123b}$ + PAG(LDA)	21.48	41.25	50.45	28.43	17.98	13.13	23.35	35.85	40.06
XBNet	18.96	38.47	49.59	27.36	17.33	13.07	21.31	33.71	38.75
gemma2$_{27b}$ + PAG(XBNet)	**20.57**	**46.67**	**58.40**	**27.77**	**20.29**	**15.09**	**22.53**	**38.79**	**44.11**
llama3.1$_{8b}$ + PAG(XBNet)	17.91	40.57	54.08	24.77	17.93	14.06	19.81	33.95	40.07
llama3.1$_{70b}$ + PAG(XBNet)	18.75	44.00	56.80	25.53	19.20	14.70	20.63	36.24	42.02
mixtral$_{8\times22b}$ + PAG(XBNet)	20.51	44.76	56.10	27.26	19.50	14.53	22.37	37.49	42.62
mistral-large$_{123b}$ + PAG(XBNet)	19.43	45.87	57.25	25.81	19.98	14.77	21.17	37.72	42.85
TabNet	17.05	31.77	42.04	25.30	14.92	11.35	19.35	28.68	33.32
gemma2$_{27b}$ + PAG(TabNet)	**20.03**	**42.74**	**53.78**	**27.84**	**18.82**	**13.98**	**22.18**	**36.35**	**41.36**
llama3.1$_{8b}$ + PAG(TabNet)	15.72	35.71	46.33	22.86	16.07	12.28	17.68	30.20	35.04
llama3.1$_{70b}$ + PAG(TabNet)	16.49	37.90	49.79	23.55	17.05	13.13	18.45	31.92	37.30
mixtral$_{8\times22b}$ + PAG(TabNet)	18.33	39.63	50.16	25.34	17.66	13.17	20.26	33.74	38.52
mistral-large$_{123b}$ + PAG(TabNet)	18.45	40.13	50.12	25.43	17.84	13.17	20.37	34.14	38.65
MAEA	**25.78**	47.56	58.82	**34.65**	20.75	15.22	**28.21**	41.90	46.96
gemma2$_{27b}$ + PAG(MAEA)	25.43	**52.29**	**63.79**	32.91	**22.42**	**16.36**	27.46	**44.21**	**49.38**
llama3.1$_{8b}$ + PAG(MAEA)	21.92	47.71	60.66	28.89	20.56	15.60	23.81	39.78	45.61
llama3.1$_{70b}$ + PAG(MAEA)	23.05	48.45	60.95	30.21	20.90	15.67	25.00	40.90	46.50
mixtral$_{8\times22b}$ + PAG(MAEA)	23.73	49.12	62.18	30.68	21.14	15.97	25.59	41.75	47.63
mistral-large$_{123b}$ + PAG(MAEA)	23.06	49.18	61.97	29.86	21.21	15.94	24.88	41.30	47.08

Table 11: WAM score of Predictive Models (PM) and PAG with three tests of PolyMed. Metrics "R", "P", and "ND" denote Recall, Precision, and NDCG respectively.

Utterance I've been feeling really tired and without much energy lately. I've also been throwing up and having really watery stools. I don't have much of an appetite, either.
Extraction `{"symptoms": {"Fatigue": True, "Nausea": False, "Vomiting": True, "Diarrhea": True, "Loss of appetite": True}}`
Validation `{"symptoms": {"Fatigue": True, "Nausea": True, "Vomiting": True, "Diarrhea": True, "Loss of appetite": True}}`

Utterance I've been having a lot of trouble lately with persistent coughing and I'm also coughing up quite a bit of phlegm. On top of that, I'm experiencing some shortness of breath, and there's this uncomfortable feeling in my throat that just won't go away. It's really starting to worry me.
Extraction `{"symptoms": {"Persistent coughing": True, "Coughing up phlegm": True, "Shortness of breath": True, "Uncomfortable feeling in the throat": True}}`
Validation <NPB>

Utterance I've been having a really tough time with this persistent stuffiness lately. My nose feels completely congested, and there's noticeable swelling inside. It's like my nasal passages are constantly blocked and swollen, making it uncomfortable to breathe. However, I haven't had any fever or experienced any double vision.
Extraction Here is the structured summary of the patientś reported symptoms:{"symptoms": {"Stuffy nose": True, "Nasal congestion": True, "Swelling in nasal passages": True, "Difficulty breathing": True, "Fever": False, "Double vision": False}}
Validation The following information from the text data is missing in the structured data: * "persistent stuffiness"* "noticeable swelling inside"* "nasal passages are constantly blocked and swollen"Corrected Structured Data:{"symptoms": {"Stuffy nose": True,"Nasal congestion": True,"Swelling in nasal passages": True,"Difficulty breathing": True,"Fever": False,"Double vision": False,"Persistent stuffiness": True,"Noticeable swelling inside": True,"Constant blockage and swelling of nasal passages": True}}

Utterance "I've been dealing with a fever and these red spots that have turned into blisters. My mouth has been sore, and I really haven't felt like eating anything. I'm quite worried because it seems to be progressively getting worse. Can you help me understand what's going on?"
Extraction It seems you haven't provided any text related to a patient's dialogue with their doctor. Could you please provide the necessary information so I can assist you in extracting a structured summary of the patient's reported symptoms, medical history, and specific concerns?
Validation <NPB>

Table 12: Feature extraction examples. **Utterance** is a patient symptom utterance generated by Table 15 prompt. **Extraction** and **Validation** is corresponding extracted feature and validation using Table 16 prompt. The red text shows examples of incorrect information extraction.

Original {"Restlessness": true, "Fever": true, "Rash": true, "Blisters": true, "Lethargy": true, "Coughing": false, "Vomiting": false, "Anorexia (Loss of appetite)": false} **Draft** I've been feeling really restless and I've had this fever that just won't go away. There's also this rash that's appeared and some blisters have started forming. I'm also feeling more tired and lethargic than usual. It's been pretty uncomfortable and I'm not sure what's going on. **Validation** Lately, I've been feeling really restless and can't shake off this persistent fever. I've noticed a rash and some blisters developing, and overall, I've been feeling much more tired and lethargic than usual. Despite these symptoms, I haven't been experiencing any coughing or vomiting, and my appetite remains normal without any signs of anorexia.
Original {"Chest tightness": true, "Pain behind the breastbone": true, "Diarrhea": true, "Acid reflux": true} **Draft** I've been dealing with some really uncomfortable chest tightness and pain right behind my breastbone. On top of that, I've been experiencing diarrhea and a lot of acid reflux. It's starting to get pretty concerning, and I'm not sure what's causing all of this. **Validation** <NPB>

Table 13: Utterance generation examples. The **Original** is the symptom data of the ADS dataset. **Draft** and **Validation** are generated by the prompt of Table 15

User Prompt Disease information:
Mastitis - (51.25%) Breast swelling, breast pain, skin redness, warmth to touch, lump or thickening of tissue, fever, chills. Coronary heart disease - (44.31%) Chest pain (angina), Shortness of breath, Fatigue, Irregular heartbeat. Enteritis - (2.81%) Abdominal pain, Diarrhea, Nausea, Vomiting, Fever, Bloody stools, Loss of appetite, Weight loss, Fatigue. Pneumonia - (1.37%) cough, fever, chills, difficulty breathing/shortness of breath, chest pain when breathing or coughing, fatigue, nausea or vomiting, diarrhea, confusion (especially in older adults), loss of appetite. Asthma - (0.2%) Coughing (especially at night), Wheezing, Shortness of breath, Chest tightness or pain, Difficulty breathing, Fatigue. Esophagitis - (0.03%) Difficulty swallowing, chest pain, heartburn, regurgitation of food or liquids, sensation of a lump in the throat, chronic cough, hoarse voice, frequent hiccups. Traumatic brain injury - (0.02%) Headache, dizziness, confusion, nausea or vomiting, fatigue, speech difficulties, sleeping problems, memory loss, mood changes, sensitivity to light and sound, blurred vision, ringing in the ears, seizures. Thyroiditis - (0.01%) Neck swelling, Pain in the neck or throat, Fatigue, Weight gain, Cold intolerance, Constipation, Dry skin, Muscle weakness, Joint pain, Irregular menstrual periods, Depression, Thinning hair. Conjunctivitis - (0.0%) [Redness of the eye, Itching or burning sensation, Excessive tearing, Thick yellow discharge that crusts over the eyelashes during sleep, Blurred vision, Increased sensitivity to light, Swollen lymph nodes, Gritty feeling in the eyes] Rhinitis - (0.0%) ["Sneezing", "Runny nose", "Stuffy nose", "Coughing", "Itchy nose", "Postnasal drip", "Watery eyes", "Fatigue", "Headache"] External otitis - (0.0%) Ear pain, itching, discharge from ear, difficulty hearing, redness or swelling of outer ear. Dermatitis - (0.0%) [itchy skin, redness, swelling, rash, burning sensation, dry and scaly skin, blisters, crusting, oozing and weeping sores, skin color changes]
Patient symptom:
Lately, I've been feeling really fatigued and have a fever that just won't go away. There are times when I feel dizzy and have even started vomiting. I've also noticed some discomfort in my throat and an unusual sensitivity to cold. However, I haven't been experiencing a cough, incontinence, body aches, expectoration, or nausea. Can you help me figure out what's going on? Diagnosis:
D^T **Generation** Based on the symptoms provided - fatigue, fever that won't go away, dizziness, vomiting, throat discomfort, and unusual sensitivity to cold - the most likely diagnosis is Mastitis with a probability of 51.25%. The key symptoms that support this conclusion are the persistent fever, fatigue, and throat discomfort. These align well with the typical presentation of mastitis, which often includes breast swelling, pain, skin redness, warmth to touch, and fever or chills.
It is important to note that while these symptoms do not perfectly match those of Mastitis (there's no mention of specific breast-related symptoms), they are the closest fit given the disease information provided. However, a comprehensive medical evaluation by a healthcare professional would be necessary for an accurate diagnosis.
D^L **Generation** Mastitis Traumatic brain injury Thyroiditis Rhinitis

Table 14: PAG Diagnostic Example. **User Prompt** supplies the LLM with diagnostic details, including the patient's symptom utterance and "Disease information"—which aggregates the predictive model's output with LLM-generated knowledge for each disease. The D^T **Generation** is the corresponding text-based rationale, while the D^L **Generation** represents the final disease label inferred from the user prompt and D^T. The LLM Diagnosis was produced using `system messages` provided in Table 18.

Task	Prompt
Utterance Generation	You are tasked with generating a patient's utterance that conveys their symptoms to a doctor in a natural and convincing way. You will be provided with a diagnosis, a list of symptoms the patient has, and a list of symptoms the patient does not have. Your job is to create an utterance that reflects the patient's condition by describing only the symptoms they experience, while avoiding the use of any disease names or direct references to medical diagnoses. Make the response sound like a genuine patient seeking medical attention. Be concise but descriptive enough to convey the severity or concerns. DO NOT INCLUDE INFORMATION THAT IS NOT IN THE PATIENT SYMPTOMS DO NOT INCLUDE DOCTOR'S TALK. ONLY NEED PATIENT'S UTTRANCE DO NOT INCLUDE DISEASE NAME IN THE UTTERANCE, ONLY INCLUDE SYMPTOMS YOU MUST INCLUDE ALL THE PAITENT'S SYMTPOMS Here is the patient's information:
Utterance Validation	You are tasked with verifying whether a patient's utterance fully captures both the symptoms they are experiencing and those they are not experiencing. You will be provided with two lists: one containing symptoms the patient has and one containing symptoms the patient does not have. - If there is no missing symptoms, generate <NPB> token. - If any symptoms (experienced or not experienced) are missing, rewrite the utterance to include all relevant information. The rewritten utterance should sound natural and reflect how a patient would describe both their symptoms and the absence of specific symptoms to the doctor.

Table 15: `system messages` for utterance generation.

Task	Prompt
Feature Extraction	Analyze this patient's dialogue with their doctor and extract a structured summary of the patient's reported symptoms, medical history, and any specific concerns they mention. Organize the information in a clear and concise format. `{#Feature schema example}` IF THERE IS NO CORRESPONDING INFORMATION, MARK IT AS NULL. IF THERE ARE MULTIPLE ELEMENTS SEPARATE THEM WITH A COMMA. ENTER SYMPTOMS AS A DICTIONARY TYPE AND VALUES AS True IF THE PATIENT HAS THE SYMPTOM OR False IF THE PATIENT DOES NOT. STRUCTURED DATA MUST BE DICTIONARY STRUCTURE. EXTRACT INFORMATION WITHOUT EXPLANATIONS AND GREETINGS. COLLECT INFORMATION ONLY ABOUT THE USER'S SYMTPOMS AND DO NOT COLLECT ANY OTHER INFORMATION SUCH AS MEDICAL HISTORY, SPECIFIC CONCERNS. Here is the patient's information:
Feature Validation	Given the following text data and corresponding structured data, check if all the information from the text data is accurately and completely included in the structured data. If any information is missing in structured data, specify which details are not present. `{#Feature schema example}` - If there is no missing information, generate <NPB> token. - If there is missing information, add missing data to the structured data. - Ensure there are no syntax errors. If any errors exist, correct them to make the output a valid Python dictionary. STRUCTURED DATA MUST BE DICTIONARY STRUCTURE EXTRACT STRUCTURED INFORMATION WITHOUT EXPLANATIONS AND GREETINGS AND TITLES

Table 16: `system messages` for feature extraction.

Task	Prompt
Term match	You are an expert in medical terminology. Given a query term and a list of retrieved terms, match the retrieved term that has the same meaning as the query term. Provide only the term that matches. Match the term with the same meaning as the query term. If there is no appropriate match, return <NW>. Example Query Term: Hypertension Retrieved Terms: High blood pressure Hypotension Tachycardia Bradycardia Arrhythmia Hyperglycemia Hypoglycemia Hypertrophy Hyperlipidemia Hypoxia Correct Match: High blood pressure
LLM Diagnosis	You are an expert in medical diagnostics. Given patient information provided by a patient, predict up to 5 suspected diseases that the patient may have. Provide the list of predicted diseases without any numbering or explanation or hyphen. Example Patient Information: Sex: Female Age: 45 Family History: Heart disease, high cholesterol Background: Non-smoker, regular exercise, no alcohol Underlying Disease: Hypertension Symptoms: Chest pain, shortness of breath, dizziness, fatigue, irregular heartbeat Predicted Diseases: Common Cold Influenza Pneumonia Allergic Rhinitis Bronchitis
RAG	You are an expert in medical diagnostics. Given patient information provided by a patient, predict up to 5 suspected diseases that the patient may have by referring to the Disease Knowledge. Provide the list of predicted diseases without any numbering or explanation. Example Patient Information: Sex: Female Age: 45 Family History: Heart disease, high cholesterol Background: Non-smoker, regular exercise, no alcohol Underlying Disease: Hypertension Symptoms: Chest pain, shortness of breath, dizziness, fatigue, irregular heartbeat Disease Knowledge: Upper respiratory infections: Knowledge of Upper respiratory infections Influenza: Knowledge of Influenza Pneumonia: Knowledge of Pneumonia Allergic rhinitis: Knowledge of Allergic rhinitis Bronchitis: Knowledge of Bronchitis Sinusitis: Knowledge of Sinusitis Tuberculosis: Knowledge of Tuberculosis Asthma: Knowledge of Asthma COVID-19: Knowledge of COVID-19 Lung Cancer: Knowledge of Lung Cancer Predicted Diseases: Upper respiratory infections Influenza Pneumonia Allergic Rhinitis Bronchitis

Table 17: `system messages` for LLM prediction and RAG with Term match.

Task	Prompt
K_d Generation	What symptoms will appear in the {} disease? List only symptom, without any description, following form [symptom1, symptom2, ...].
D^T Generation	Given the patient's symptoms and relevant disease information, provide a concise diagnosis. Briefly explain the reasoning behind this diagnosis, identifying key symptoms, relevant medical history, or any notable patterns that support the conclusion. Do not include any disclaimers, warnings, or mentions of AI generation
D^L Generation	You are an expert in medical diagnostics. Given patient information (pi) provided by a patient, predict suspected diseases that the patient may have by referring to the retrieved Disease information (di) and diagnosis (ds). Provide the list of predicted diseases without any numbering or explanation. Example: Patient information: pi Disease information: di Diagnosis: ds Predicted Diseases: Upper respiratory infections Influenza Pneumonia Allergic Rhinitis Bronchitis
PAG w/o K_d	You are an expert in medical diagnostics. Given a patient's information, a query disease diagnosed by a predictive model, and a list of predicted diseases, remove any inaccurate or irrelevant diseases. If no modifications are necessary, return <NW>. Provide only the filtered list of diseases without any numbering or explanation. Patient Information: Sex: Female Age: 35 Family History: History of respiratory diseases Background: Non-smoker, no known allergies Underlying Disease: None Symptoms: Fever, cough, sore throat, fatigue, body aches Predicted Diseases: Common Cold Influenza Pneumonia Allergic Rhinitis Bronchitis Sinusitis Tuberculosis Asthma COVID-19 Lung Cancer Filtered List: Common Cold Influenza Pneumonia Allergic Rhinitis Bronchitis Sinusitis COVID-19

Table 18: `system messages` for PAG.