Neural at ArchEHR-QA 2026: One Method Fits All: Unified Prompt Optimization for Clinical QA over EHRs

Abrar Majeedi[1,*], **Viswanatha Reddy Gajjala**[1,*],
Sai Prasanna Teja Reddy Bogireddy[2,*], **Siddhant Rai**[3,*]
[1]University of Wisconsin–Madison [2]University of Chicago [3]Independent Researcher
{amajeedi, vgajjala}@wisc.edu, bogireddytejareddy@uchicago.edu
*Equal contribution

Abstract

Automated question answering (QA) over electronic health records (EHRs) demands precise evidence retrieval, faithful answer generation, and explicit grounding of answers in clinical notes. In this work, we present *Neural1.5*, our method for the ArchEHR-QA 2026 shared task at CL4Health@LREC 2026, which comprises of four subtasks: question interpretation, evidence identification, answer generation, and evidence alignment. Our approach decouples the task into independent, modular stages and employs DSPy's MIPROv2 optimizer to automatically discover high-performing prompts, jointly tuning instructions and few-shot demonstrations for each stage. Within every stage, self-consistency voting over multiple stochastic inference runs suppresses spurious errors and improves reliability, while stage-specific verification mechanisms (e.g., self-reflection and chain-of-verification for alignment) further refine output quality. Among all teams that participated in all four subtasks, our method ranks second overall (mean rank 4.00), placing 4th, 1st, 4th, and 7th on Subtasks 1–4, respectively. These results demonstrate that systematic, per-stage prompt optimization combined with self-consistency mechanisms is a cost-effective alternative to model fine-tuning for multi-faceted clinical QA.

Keywords: clinical question answering, prompt optimization, electronic health records, large language models, evidence grounding, evidence alignment

1. Introduction

Patient medical advice requests have surged 55% since 2019, with physicians now spending 24% more time on inbox management (Arndt et al., 2024). Automatically answering these questions using electronic health records (EHRs) could substantially reduce clinician burden, yet requires systems that not only generate accurate responses but also explicitly ground every claim in verifiable clinical evidence. The ArchEHR-QA 2026 shared task at CL4Health@LREC 2026 (**?**) takes a step towards addressing this challenge through four complementary subtasks: (1) transforming verbose patient questions into concise clinician-interpreted queries, (2) identifying minimal evidence sentences from clinical notes, (3) generating grounded answers, and (4) aligning each answer sentence to its supporting evidence. Together, these subtasks capture the complete pipeline from question understanding to explainable, evidence-backed clinical QA.

The natural language processing capabilities of Large Language Models (LLMs) present a promising approach. Although LLMs have demonstrated strong performance in clinical QA (Singhal et al., 2025), their deployment faces two critical barriers. First, fine-tuning on clinical data is constrained by limited supervised datasets and overfitting risks. Second, prompt engineering for multi-stage clinical workflows remains labor-intensive and domain-specific, requiring expert iteration to identify optimal instructions and demonstrations (Karayanni et al., 2024). Existing automated prompt optimization techniques (Wang et al., 2023) typically treat tasks holistically, failing to leverage the modular structure inherent in clinical QA pipelines where evidence identification, answer generation, and citation alignment each demand distinct reasoning patterns.

In this work, we present **Neural1.5**, a modular LLM method that addresses all four ArchEHR-QA 2026 subtasks through *automated, stage-specific prompt optimization*. Our key design principle is systematic task decomposition: by defining clear evaluation objectives for each subtask, we enable DSPy's MIPROv2 optimizer (Khattab et al., 2024) to automatically discover high-performing prompts that jointly tune instructions and few-shot demonstrations. We further enhance reliability through *self-consistency voting* (Wang et al., 2022) for evidence identification and a *three-stage alignment pipeline* combining initial alignment, self-reflection to prune false positives, and chain-of-verification with confidence-weighted majority voting.

Our proposed method achieves competitive results across all four subtasks, ranking **4th** (Subtask 1), **1st** (Subtask 2 with Strict Micro F1 of 63.7), **4th** (Subtask 3), and **7th** (Subtask 4), with a **mean rank of 4.00** across all subtasks—**second overall** among teams participating in all four subtasks. These results demonstrate that systematic, per-stage prompt optimization combined with self-

Proceedings of the Third Workshop on Patient-Oriented Language Processing (CL4Health), pages 530–538
12 May 2026.

consistency mechanisms offers a cost-effective alternative to model fine-tuning for multi-faceted clinical QA.

The key contributions of our work are:

- **Full-Pipeline Automated Optimization:** We propose the method to apply automated prompt optimization independently to all four ArchEHR-QA subtasks, with stage-specific objectives that capture distinct reasoning requirements for question interpretation, evidence retrieval, answer generation, and citation alignment.
- **Three-Stage Alignment with Self-Reflection:** We introduce an alignment pipeline that combines initial sentence-level alignment, self-reflection to identify and remove spurious citations, and chain-of-verification, with confidence-weighted majority voting across multiple stochastic runs to suppress hallucinations/errors.
- **Consistent Multi-Task Performance:** With a mean rank of 4.00 across all four subtasks, we demonstrate that modular, optimization-driven approaches can maintain strong performance across diverse clinical QA challenges, offering a practical blueprint for deploying LLMs in safety-critical healthcare applications.

2. Related Work

Clinical QA: Developing QA systems for clinical data has long been an interest in biomedical NLP. Earlier datasets like emrQA (Pampari et al., 2018) generated large-scale QA pairs from electronic medical records. Recent research has shown that LLMs can achieve near-expert performance on medical QA benchmarks (Singhal et al., 2025). The ArchEHR-QA shared task (Soni and Demner-Fushman, 2026b; Soni et al., 2025) advances this line of work by requiring systems to ground answers explicitly in clinical notes. Our prior work (Bogireddy et al., 2025) demonstrated the effectiveness of automated prompt optimization for a single task, achieving second place overall. Building on these insights, the 2026 edition expands to four complementary subtasks spanning question interpretation, evidence identification, answer generation, and evidence alignment.

Prompt Optimization: There is growing interest in automated prompt search. Methods such as APE (Zhou et al., 2022) and OPRO (Yang et al., 2023) treat prompt design as a black-box optimization problem. MIPRO (Opsahl-Ong et al., 2024) extends this to multi-stage LLM programs, jointly optimizing instructions and demonstration examples. Our work leverages MIPROv2 (Opsahl-Ong et al., 2024), which uses a combination of prompt proposal and Bayesian search.

Self-Consistency: LLMs can produce variable outputs given the same prompt. The self-consistency decoding strategy (Wang et al., 2022) addresses this by sampling multiple outputs and selecting the most consistent result.

3. Task Description

The ArchEHR-QA 2026 shared task (Soni and Demner-Fushman, 2026b) comprises four subtasks. The dataset (Soni and Demner-Fushman, 2026a) consists of patient-authored questions, clinician-interpreted counterparts, clinical note excerpts with sentence-level relevance annotations, and reference clinician-authored answers with answer–evidence alignments.

Subtask 1: Question Interpretation. Given a free-text patient question, generate a concise clinician-interpreted question ($\leq$15 words) that captures the core clinical information need.

Subtask 2: Evidence Identification. Given the patient question, clinician question, and a clinical note excerpt with numbered sentences, identify the minimal set of note sentences that provide evidence needed to answer the question.

Subtask 3: Answer Generation. Given the questions and clinical note excerpt, generate a grounded natural-language answer ($\leq$75 words) using only information from the notes.

Subtask 4: Evidence Alignment. Given the questions, clinical note excerpt, and a reference answer with numbered sentences, align each answer sentence to its supporting note sentence(s).

4. Methodology

Our method draws on a human-inspired decoupling strategy, separating question understanding, evidence gathering, answer formulation, and evidence attribution into distinct stages. We operationalize this intuition as a modular pipeline, with each subtask addressed by a DSPy program whose prompts are optimized independently. The initial prompt templates (DSPy signatures) for all subtasks are provided in Appendix A.

4.1. Subtask 1: Question Interpretation

We define a DSPy `Signature` for question interpretation that instructs the LLM to transform a patient narrative into a concise clinician-interpreted question. The signature encodes domain-specific

heuristics: preserving key medical terms exactly (procedure names, medication names, condition names), using patient-specific phrasing ("him/her/the patient"), and following high-scoring question patterns (e.g., "Why was [X] recommended to him/her?").

The module uses `ChainOfThought` prompting and enforces a strict 15-word limit through post-processing. We optimize the prompt using MIPROv2 with a composite metric combining semantic entailment (via an LLM-as-judge evaluator assessing AlignScore-like semantic equivalence), key term preservation, and structural conformity.

Prompt-Optimization Objective: MIPROv2 searches the space of instructions and few-shot exemplars to maximize a weighted composite of semantic alignment (60%), key term preservation (25%), and question structure (15%) on the development set.

4.2. Subtask 2: Evidence Identification

For each question–note pair, we classify each note sentence as essential or irrelevant. Given the clinical note with sentences $s_1, s_2, \ldots, s_n$ and gold labels $y_i \in \{\text{essential}, \text{supplementary}, \text{not-relevant}\}$, the model predicts binary labels $\hat{y}_i \in \{0, 1\}$.

The classification uses a two-step process. First, for each sentence, we generate reasoning about why it is or is not essential using dedicated reasoning signatures (`EssentialReasoning` and `NonEssentialReasoning`). These reasoning traces serve as few-shot demonstrations for the main classifier. Second, a `MedicalAnswerWithCitations` signature classifies all sentences jointly, providing a relevancy score (0–10) and reasoning for each.

Prompt-Optimization Objective: We invoke MIPROv2 to optimize the classification prompt, maximizing the sentence-level F_1 between predicted and gold essential sentences:

$$F_1(Y^+, \hat{Y}^+), \qquad Y^+ = \{i \mid y_i = \text{essential}\}, \quad \hat{Y}^+ = \{i \mid \hat{y}_i = 1\}.$$

Self-Consistency Voting: The classifier is executed $R = 5$ times on the same input with stochastic sampling (temperature 0.8). The final label is obtained by majority vote:

$$v_i = \sum_{r=1}^{R} \hat{y}_i^{(r)}, \quad \hat{y}_i = \begin{cases} 1 & \text{if } v_i \geq \lceil R/2 \rceil, \\ 0 & \text{otherwise.} \end{cases}$$

This suppresses spurious single-run errors and retains sentences identified as essential by at least three of five passes.

4.3. Subtask 3: Answer Generation

Given a question q and the set of essential sentences $E = \{s_i \mid \hat{y}_i = 1\}$ from Subtask 2, Stage 3 produces a concise answer a_{gen} ($\leq$75 words). The `GroundedMedicalAnswer` signature encodes key constraints: the answer must use only facts from the clinical notes, preserve exact clinical terminology, write in professional register, and avoid citation markers.

To further improve answer quality, we employ a consolidation step: the model generates $R = 5$ candidate answers using stochastic sampling (temperature 0.9), and a separate consolidation prompt selects claims consistently supported across candidates to produce a single final answer.

Prompt-Optimization Objective: MIPROv2 optimizes the answer generation prompt using an LLM-as-judge metric that evaluates: faithfulness to the clinical notes, medical completeness (concept coverage), lexical similarity to the reference, and coherence (structure and register).

4.4. Subtask 4: Evidence Alignment

Given a question q, clinical note sentences, and reference answer sentences $a_1, \ldots, a_m$, the method must produce alignment links (a_j, s_i) mapping each answer sentence to its supporting note sentence(s).

We introduce a **three-stage pipeline**:

Stage A – Initial Alignment: The `EvidenceAlignment` signature instructs the model to align each answer sentence to note sentences that directly support it, outputting confidence scores for each link. The prompt explicitly warns against both over-citing and under-citing.

Stage B – Self-Reflection: The `SelfReflection` signature critically reviews the initial alignment to identify and remove false positive links (indirect or inferential connections) and add any clearly missing links. This step primarily targets precision improvement.

Stage C – Chain-of-Verification: The `ChainOfVerification` signature performs a final verification pass, checking each link against three criteria: (i) does the answer sentence directly reference information from the note sentence? (ii) would removing the note sentence cause the answer sentence to lose specific evidence? (iii) is the connection direct rather than inferential?

Confidence-Weighted Majority Voting: The full three-stage pipeline is executed R times with stochastic sampling. For each answer–evidence

link, we count votes across runs and compute average confidence. A link is retained only if it receives at least $\lceil R/2 \rceil$ votes *and* the average confidence exceeds a threshold $\tau_c = 0.9$. This dual filtering mechanism balances recall and precision.

5. Experimental Setup

Dataset: We evaluated our method on the ArchEHR-QA 2026 dataset (Soni and Demner-Fushman, 2026a), which contains patient questions alongside clinical note excerpts derived from the MIMIC database, with sentence-level relevance annotations and reference answers with answer–evidence alignments. The development set comprises 20 cases (IDs 1–20) used for prompt optimization. Test set sizes vary by subtask: Subtasks 1–3 evaluate on 47 cases (IDs 121–167), while Subtask 4 evaluates on 147 cases (IDs 21–167), reflecting the staged data release schedule.

Evaluation Metrics: Each subtask is evaluated independently. **Subtask 1** (Question Interpretation) is evaluated using ROUGE, BERTScore, AlignScore, and MEDCON. **Subtask 2** (Evidence Identification) uses Precision, Recall, and F1 over predicted vs. gold evidence sentences, with strict (essential-only) and lenient (including supplementary) variants. **Subtask 3** (Answer Generation) uses BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), SARI (Xu et al., 2016), BERTScore (Zhang et al., 2020), AlignScore (Zha et al., 2023), and MEDCON (Yim et al., 2023). **Subtask 4** (Evidence Alignment) uses Precision, Recall, and F1 over predicted alignment links.

LLM Configuration: Across all subtasks, we employ the GPT-4.1 model accessed via the Azure OpenAI API. We chose GPT-4.1 for its strong instruction-following capabilities and large context window, which are critical for processing lengthy clinical notes. For prompt optimization with MIPROv2, we use temperature 0.3 to ensure deterministic optimizer feedback. For self-consistency runs, we use temperature 0.7 (Subtask 2), 0.8 (Subtask 4), and 0.9 (Subtask 3) to capture model uncertainty. The maximum context window is set to 10,000 tokens for Subtasks 2–4 and 2,000 tokens for Subtask 1. All other decoding parameters (e.g., top-p, frequency and presence penalties) are held at API defaults. We estimate the computational cost per case at approximately 5 LLM calls for Subtask 1, 25 calls (5 self-consistency runs $\times$ 5 reasoning steps) for Subtask 2, 6 calls for Subtask 3 (5 candidates + 1 consolidation), and 15–25 calls for Subtask 4 (3 stages $\times$ R runs).

Team	Overall	R.L.	B.S.	A.S.	M.C.
HealthNLLP_Ret.	**31.2**	**35.3**	**46.8**	**24.0**	18.7
KPSCMI	30.8	27.8	41.0	26.4	**27.9**
OptiMed	29.9	28.8	43.1	27.7	19.9
Neural1.5 (Ours)	28.9	31.3	43.6	15.2	25.6
Yale-DM-Lab	27.1	28.2	40.6	19.7	19.8

Table 1: Subtask 1: Question Interpretation results on the test set. **Bold** = best, underlined = second best.

6. Results

Tables 1–4 present per-subtask results, and Table 5 summarizes rankings across all four subtasks.

Subtask 1: Question Interpretation. Our method achieves an overall score of 28.9, ranking 4th among 13 teams (Table 1). Our MEDCON score of 25.6 is the second highest, indicating strong preservation of medical concepts. The relatively lower AlignScore (15.2) suggests room for improvement in semantic alignment with reference questions, possibly due to differences in question framing style.

Subtask 2: Evidence Identification. Our method ranks **1st** with an overall Strict Micro F1 of 63.7 (Table 2), demonstrating the effectiveness of the prompt-optimized classification with self-consistency voting. Notably, our method achieves strong performance across all evaluation granularities: Strict Micro (P=60.2, R=67.6, F1=63.7), Lenient Micro (P=77.8, R=67.6, F1=72.4), and the highest Lenient Macro F1 of 73.1. This balanced profile avoids the extreme precision-recall tradeoffs seen in some competing methods—for example, Yale-DM-Lab achieves the highest recall (75.7) but at the cost of the lowest precision (52.3).

Subtask 3: Answer Generation. We rank 4th with an overall score of 35.2 (Table 3). Our method achieves competitive BLEU (9.4) and AlignScore (34.3) results, with the latter being the second highest among all teams. Our MEDCON score of 40.9 ranks 4th, indicating reasonable preservation of medical concepts in the generated answers. The answer consolidation step, which aggregates five candidate answers via majority-supported claims, helps ensure factual consistency.

Subtask 4: Evidence Alignment. Our method achieves a Micro F1 of 78.6, ranking 7th among 15 teams (Table 4). The three-stage pipeline (alignment, self-reflection, chain-of-verification) combined with confidence-weighted majority voting yields high precision (84.3), though recall (73.7)

Team	Overall	Strict Micro			Lenient Micro			Strict Macro			Lenient Macro		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Neural1.5 (Ours)	**63.7**	60.2	67.6	**63.7**	77.8	67.6	72.4	64.3	69.7	64.8	81.9	69.7	73.1
OptiMed	63.2	56.7	71.3	63.2	73.1	71.3	72.2	61.5	73.0	64.1	78.1	73.0	72.9
UIC-AIHealth4All	62.9	59.3	67.0	62.9	74.3	67.0	70.4	61.8	69.2	63.0	77.8	69.2	70.5
Yale-DM-Lab	61.9	52.3	**75.7**	61.9	68.0	**75.7**	71.6	56.7	**75.5**	61.1	73.8	**75.5**	70.4
TJU222	61.4	54.6	70.0	61.4	74.4	70.0	**72.2**	61.4	70.3	60.6	78.9	70.3	70.7
TAMU-NLP-Lab	60.9	56.7	65.9	60.9	76.0	65.9	70.6	63.0	66.7	59.9	80.0	66.7	69.1

Table 2: Subtask 2: Evidence Identification results on the test set.

Team	Overall	BLEU	R.L.	SARI	B.S.	A.S.	M.C.
WisPerMed	**36.3**	**9.9**	**27.8**	58.6	**46.8**	**31.7**	**43.1**
TAMU-NLP-Lab	36.2	9.7	27.4	59.6	46.2	34.5	39.9
BIT.UA-AAUBS	35.6	8.6	26.4	**60.0**	45.0	30.2	43.2
Neural1.5 (Ours)	35.2	9.4	25.6	57.7	43.5	34.3	40.9
HealthNLP_Ret.	34.6	7.0	25.4	59.2	43.8	33.6	38.7
OptiMed	34.5	5.7	25.2	56.5	43.1	37.4	39.1

Table 3: Subtask 3: Answer Generation results on the test set.

Team	Overall	M.P.	M.R.	M.F1
BIT.UA-AAUBS	**81.5**	**88.0**	75.9	**81.5**
WisPerMed	81.3	86.9	76.3	81.3
Yale-DM-Lab	80.4	83.3	77.7	80.4
OptiMed	80.3	80.7	**79.8**	80.3
UIC-AIHealth4All	79.8	83.6	76.3	79.8
tt501	79.1	78.2	80.1	79.1
Neural1.5 (Ours)	78.6	84.3	73.7	78.6
KPSCMI	78.1	86.2	71.5	78.1

Table 4: Subtask 4: Evidence Alignment results on the test set. Overall = Micro F1.

Team	ST1	ST2	ST3	ST4	Mean↓
OptiMed	3	2	6	4	3.75
Neural1.5 (Ours)	4	**1**	4	7	**4.00**
WisPerMed	6	12	**1**	2	5.25
HealthNLLP_Ret.	**1**	7	5	9	5.50
KPSCMI	2	8	–	8	6.00
Yale-DM-Lab	5	4	13	3	6.25
TAMU-NLP-Lab	11	6	2	–	6.33
BIT.UA-AAUBS	13	10	3	**1**	6.75

Table 5: Summary of rankings across all four subtasks for teams that participated in at least three subtasks. Mean Rank is computed over participated subtasks (lower is better). "–" = non-participation.

is somewhat lower. The conservative confidence threshold ($\tau_c = 0.9$) favors precision over recall, reflecting our design choice to minimize false alignment links.

Cross-Subtask Consistency. Table 5 highlights a key strength of our approach: *consistent performance across all four subtasks*. With a mean rank of 4.00 across subtasks, our method is second only to OptiMed (3.75) among teams participating in all four subtasks. While several teams achieve top ranks on individual subtasks (e.g., WisPerMed on ST3, BIT.UA-AAUBS on ST4), they exhibit more variance across the full task suite. Our modular prompt optimization framework delivers robust performance without specializing in any single subtask at the expense of others.

7. Conclusion

We present a modular approach for all four subtasks of the ArchEHR-QA 2026 shared task, leveraging DSPy's MIPROv2 optimizer to autonomously discover high-performing prompts for each stage. For Subtask 1, the method transforms patient narratives into concise clinician queries optimized for semantic alignment. For Subtask 2, sentence-level evidence classification with self-consistency voting achieves the best F1 score among all participants. For Subtask 3, answer generation with multi-candidate consolidation produces grounded, clinically faithful responses. For Subtask 4, a novel three-stage alignment pipeline with self-reflection and chain-of-verification enables precise answer–evidence grounding. Across all four subtasks, our method achieves a mean rank of 4.00 (Table 5), the second best among teams participating in all subtasks, underscoring the consistency of our modular design.

Our results demonstrate that systematic prompt optimization, combined with self-consistency mechanisms, is a cost-effective and competitive alternative to model fine-tuning across diverse clinical QA

tasks. Future work may explore integrating external medical knowledge, cross-subtask feedback (e.g., using Subtask 1 outputs to improve Subtask 2), and extending the approach to longer clinical documents.

8. Limitations

Despite competitive performance, our method has several limitations. The pipeline treats each subtask largely independently, missing potential synergies (e.g., using evidence identification results to constrain answer generation). The self-consistency mechanism increases computational cost by a factor of R (typically 3–5 runs per input). The confidence threshold for Subtask 4 was tuned on the small development set (20 cases) and may not generalize optimally. Additionally, the method relies on GPT-4.1, making it dependent on a proprietary API, and has not been evaluated on notes from institutions beyond MIMIC. The conservative design choices (high confidence thresholds, strict majority voting) favor precision over recall, which may not be ideal for all clinical use cases.

9. Prompts and Code Availability

To promote transparency and reproducibility, we release all manual and optimized prompt templates, together with our full pipeline implementation at our GitHub repository.[1] The initial prompt templates for all subtasks are included in Appendix A.

10. Bibliographical References

Brian G Arndt, Mark A Micek, Adam Rule, Christina M Shafer, Jeffrey J Baltus, and Christine A Sinsky. 2024. More tethered to the EHR: EHR workload trends among academic primary care physicians, 2019–2023. *Annals of Family Medicine*, 22(1):12–18.

Sai Prasanna Teja Reddy Bogireddy, Abrar Majeedi, Viswanath Gajjala, Zhuoyan Xu, Siddhant Rai, and Vaishnav Potlapalli. 2025. Neural at archehr-qa 2025: Agentic prompt optimization for evidence-grounded clinical question answering. In *Proceedings of the 24th Workshop on Biomedical Language Processing (Shared Tasks)*, pages 104–109.

Nader Karayanni, Aya Awwad, Chein-Lien Hsiao, and Surish P Shanmugam. 2024. Keeping experts in the loop: Expert-guided optimization for clinical data classification using large language models. *arXiv preprint arXiv:2412.02173*.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2024. Dspy: Compiling declarative language model calls into self-improving pipelines. In *Proceedings of the Twelfth International Conference on Learning Representations*.

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

Krista Opsahl-Ong, Michael J Ryan, Josh Purtell, David Broman, Christopher Potts, Matei Zaharia, and Omar Khattab. 2024. Optimizing instructions and demonstrations for multi-stage language model programs. *arXiv preprint arXiv:2406.11695*.

Anusri Pampari, Preethi Raghavan, Jennifer Liang, and Jian Peng. 2018. emrqa: A large corpus for question answering on electronic medical records. *arXiv preprint arXiv:1809.00732*.

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 on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Karan Singhal, Tao Tu, Juraj Gottweis, Rory Sayres, Ellery Wulczyn, Mohamed Amin, Le Hou, Kevin Clark, Stephen R Pfohl, Heather Cole-Lewis, et al. 2025. Toward expert-level medical question answering with large language models. *Nature Medicine*, pages 1–8.

Sarvesh Soni and Dina Demner-Fushman. 2026a. A dataset for addressing patient's information needs related to clinical course of hospitalization. *Scientific Data*.

Sarvesh Soni and Dina Demner-Fushman. 2026b. Overview of the archehr-qa 2026 shared task on grounded question answering from electronic health records. In *Proceedings of the Third Workshop on Patient-Oriented Language Processing (CL4Health)*, Palma, Mallorca (Spain). ELRA.

Sarvesh Soni, Soumya Gayen, and Dina Demner-Fushman. 2025. Overview of the archehr-qa 2025 shared task on grounded question answering from electronic health records. In *Proceedings of the 24th Workshop on Biomedical Language Processing*, pages 396–405, Vienna, Austria. Association for Computational Linguistics.

[1] https://github.com/bogireddytejareddy/ArchEHR-QA-Neural

Xinyuan Wang, Chenxi Li, Zhen Wang, Fan Bai, Haotian Luo, Jiayou Zhang, Nebojsa Jojic, Eric P Xing, and Zhiting Hu. 2023. Promptagent: Strategic planning with language models enables expert-level prompt optimization. *arXiv preprint arXiv:2310.16427*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. 2023. Large language models as optimizers. *arXiv preprint arXiv:2309.03409*.

Wen-wai Yim, Yujuan Fu, Asma Ben Abacha, Neal Snider, Thomas Lin, and Meliha Yetisgen. 2023. Aci-bench: a novel ambient clinical intelligence dataset for benchmarking automatic visit note generation. *Scientific data*, 10(1):586.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *arXiv preprint arXiv:2305.16739*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. In *The Eleventh International Conference on Learning Representations*.

A. Prompt Templates

This appendix presents the core prompt templates (DSPy signatures) used in our method. These are the *initial* templates provided to MIPROv2; the optimizer refines the instructions and selects few-shot demonstrations automatically. Full optimized prompts are available in our repository.

A.1. Subtask 1: Question Interpretation

Prompt Template: Question Interpretation

Transform a patient's narrative into a concise clinical question ($\leq$15 words) that a clinician would need to answer by reviewing the patient's medical record.

Core Constraints:

1. $\leq$15 words, strictly enforced.
2. Patient-specific: use "him/her/the patient" — never generic.
3. Preserve medical terms: use exact procedure/medication names from the narrative.
4. Must end with a question mark.

High-Scoring Patterns:

Patient's concern	Target pattern
"Why did they do X?"	"Why was [X] recommended to him/her?"
"Will I recover?"	"What is the expected course of recovery for him/her?"
"Why X instead of Y?"	"Why was [X] recommended over [Y]?"
"Why was I given medication?"	"Why was he/she given [medication]?"
"Is this related to…?"	"Are his/her [symptoms] related to [condition]?"

Input: Patient narrative.
Output: Concise clinician question ($\leq$15 words).

Inference mode: `ChainOfThought` prompting with post-processing to enforce the word limit.

A.2. Subtask 2: Evidence Identification

Prompt Template: Reasoning Demonstrations

Essential Reasoning: Given a patient narrative, patient question, clinician question, and a clinical note sentence, provide reasoning for why this note sentence *is essential* to address the question.

Non-Essential Reasoning: Given the same inputs, provide reasoning for why this note sentence *is not essential* to address the question.

Purpose: These two prompts are used at training time to generate per-sentence reasoning traces, which serve as few-shot demonstrations for the classifier below.

Prompt Template: Evidence Classifier

You are a medical assistant.
1. You are provided with a patient narrative, patient question, and clinician question.
2. You are provided with the clinical notes related to the case.

Classify each clinical note sentence as either **essential** or **irrelevant** in addressing the patient question and clinician question. Provide a relevancy score (0–10) and reasoning for each.
Be **very critical** when assigning the essential tag. Only assign it if the note sentence is directly relevant to the specific question asked.
First-Order Relevance Only: A note is essential *only* if it directly answers or provides evidence for the question. Background context or treatment summaries are *not* essential.

Input: Patient narrative, patient question, clinician question, clinical notes.
Output per note: `<id>: <sentence> -> essential|irrelevant -> <score> -> <reasoning>`

Inference: Run $R{=}5$ times at temperature 0.8; majority vote determines the final label.

A.3. Subtask 3: Answer Generation

Prompt Template: Grounded Answer Generation

You are a medical assistant answering a patient's question using **only** information from the clinical note excerpt.
Constraints:
1. Answer must be at most 75 words ($\sim$5 sentences).
2. Use only facts stated in the clinical note. Do not add outside medical knowledge, generic advice, or speculation.
3. Write in professional clinical register (not simplified lay language).
4. Do not include citation markers such as [1], [2].
5. Reuse exact clinical wording and terminology from note sentences as much as possible.
6. The last sentence must directly answer the patient's question.

Input: Patient narrative, patient question, clinician question, clinical note excerpt.
Output: Concise grounded answer ($\leq$75 words).

Inference: Generate $R{=}5$ candidates at temperature 0.9; consolidate via a separate prompt that retains only claims consistently supported across candidates.

Prompt Template: Answer Consolidation

You are a clinical answer consolidation system. Given a patient question and 5 candidate answers generated from a clinical note:
1. Retain only claims consistently supported across the candidate answers.
2. Ground strictly in clinical note content—do not add external knowledge or speculate.
3. Use professional medical register.
4. Limit to 75 words ($\sim$5 sentences).
5. Do not include patient names or identifying information.

Output only the final consolidated answer.

A.4. Subtask 4: Evidence Alignment

Prompt Template: Stage A — Initial Alignment

You are a medical evidence alignment specialist. Align each answer sentence to the specific clinical note sentence(s) that **directly** support it.
Alignment Rules:
1. Align only when the answer sentence directly paraphrases, summarizes, or references information explicitly stated in the note sentence.
2. Do not align based on indirect associations, background context, or inferential connections.
3. Over-citing (unnecessary links) and under-citing (missing links) are both penalized.
4. Each answer sentence must be attributed to at least one note sentence. If no direct support exists, choose the closest note sentence and assign a low confidence (0.10–0.30).

Input: Patient narrative, patient question, clinician question, clinical note sentences, answer sentences.
Output per answer sentence:
`answer_sentence_k: [note_ids] (confidence=[scores])`

Prompt Template: Stage B — Self-Reflection

You are a strict reviewer performing self-reflection on an evidence alignment task. Critically review the initial alignment and identify:
1. **False positives** (primary focus): links where the answer does not directly use information from the linked note sentence. **Remove** these.
2. **False negatives** (secondary focus): missing links where an answer sentence clearly paraphrases or references a note sentence. **Add** only when direct and explicit.

Produce a corrected alignment with updated confidence scores.

Additional input: Initial alignment from Stage A.

Prompt Template: Stage C — Chain-of-Verification

You are a verification specialist. For **each** alignment link (answer sentence $k \rightarrow$ note sentence i), verify:

1. Does answer sentence k directly paraphrase or reference specific information from note sentence i?
2. If note sentence i were removed, would answer sentence k lose a specific piece of evidence it relies on?
3. Is the connection direct (not through inference or intermediate reasoning)?

If **any** check fails, remove the link. Return the final verified alignment.

Additional input: Reflected alignment from Stage B.
Post-hoc: Run the full three-stage pipeline R times; retain a link only if votes $\geq \lceil R/2 \rceil$ and average confidence ≥ 0.9.