School Health Dialogue: A Prompt-Expansion and Response-Visualization Framework

Hayato Tomisu†
R-GIRO / Graduate School of Data Science
Ritsumeikan Univ. / Shiga Univ.
Shiga, Japan
e-mail: tomisu@fc.ritsumei.ac.jp

Kazue Yamamura†
Graduate School of Human Science / School Nurse
Ritsumeikan Univ. / Ritsumeikan Moriyama J&SHS
Shiga, Japan
e-mail: kazue926@mrc.ritsumei.ac.jp

Junya Ueda
ImpactLab.
Shiga, Japan
e-mail: jueda@impactlab.jp

Tsukasa Yamanaka
Faculty of Life Sciences
Ritsumeikan Univ.
Shiga, Japan
e-mail: yaman@fc.ritsumei.ac.jp

***Abstract*—Adolescents often express mental or physical discomfort in vague terms, thereby placing a high cognitive burden on school nurses, who must interpret incomplete information. This study proposes a two-layer framework to improve school health communication by transforming ambiguous student utterances into structured, explainable dialogue flows. The first layer, auto-prompt expansion, enriches student input into slot-based representations. The second, the Prompt-Graph Domain-Specific Language, maps these representations onto a transparent decision graph for nurse supervision. The system integrates large language model orchestration, animated avatars, and real-time graph rendering. In an evaluation of 50 student complaints, prompt expansion achieved an F1 score of 0.82, whereas slot extraction scored 0.43 owing to lexical variability. AI-based rubric evaluations yielded high tone scores, indicating consistent empathy in responses; however, lower ratings for accuracy and completeness revealed deficiencies in medical specificity and follow-up guidance. Future studies will address clinical tuning, symptom normalization, and long-term field validation.**

Keywords-large language models; decision graph visualization; conversation management; adolescent wellness; school infirmary.

I. Introduction

Mental health complaints among Japanese junior and senior high school students have grown significantly complex since the onset of the COVID-19 pandemic. However, adolescents often describe their symptoms vaguely, leaving school nurses to make rapid severity decisions based on fragmented information. At Ritsumeikan Moriyama Junior and Senior High School, for example, the total number of infirmary visits increased by 52.6% between 2019 and 2021, with 38% of those visits related to non-injury and non-illness issues [1]. This growing cognitive load and diagnostic pressure highlight the urgent need for tools that transform weak, ambiguous student utterances into actionable clinical cues, while preserving the nurse's supervisory role.

Large Language Model (LLM)-based diagnostic support systems have shown promise in hospital settings but typically assume well-structured input for adult users. These systems are not designed to process the multi-symptom, low-verbal expressions typical of schoolchildren, nor do they provide the transparency required for nurse oversight. Consequently, a significant gap persists at the intersection of school health, adolescent mental health, and explainable Artificial Intelligence (AI). To address this gap, we propose a two-part framework: auto-prompt expansion, which enriches vague student input into structured representations; and Prompt-Graph Domain-Specific Language (DSL), which maps these representations onto explainable diagnostic flows. The system reduces nurses' cognitive burden and improves the consistency of initial assessments.

The remainder of this paper is organized as follows: Section II reviews related work, Section III describes the proposed method, and Section IV details its technical implementation. Section V presents a performance evaluation of the proposed method. Section VI discusses the evaluation results, limitations, and future directions. Finally, Section VII concludes the paper.

II. Related Work

A. Automatic Prompt Expansion for LLM

LLMs showed impressive few-shot abilities when supplied with carefully crafted prompts [2]. To minimize manual effort, a line of research has emerged on automatic prompt engineering. AutoPrompt searches the discrete token space via gradients to elicit factual or sentiment knowledge from masked language models [3]. In contrast, soft-prompt tuning learns continuous prefix embeddings that can be mixed and weighted for downstream tasks [4]. Reinforcement Learning Prompt (RLPrompt) frames prompt tokens as an action space and optimizes them with reinforcement learning rewards [5], while Gradient-free Instructional Prompt Search (GrIPS) performs gradient-free, edit-based instruction search to improve natural-language prompts iteratively [6]. More

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org

recent methods exploit the models themselves: Auto-Chain-of-Thought (CoT) samples its reasoning chains to create enriched demonstrations automatically [7], and Zhou et al. showed that LLM can rival humans at generating and ranking high-quality prompts [8]. Building on these insights, our Auto-Prompt Expansion optimizes prompts for adolescent health utterances.

B. Explainable and Visual Reasoning

Prompting strategies have also been used to expose model reasoning. Chain-of-Thought (CoT) prompting makes LLMs emit intermediate steps, creating human-readable rationales [9]. Accuracy improves when multiple reasoning chains are sampled and the most frequent answer is selected [10]. Least-to-Most prompting decomposes complex problems into ordered sub-tasks, yielding an explicit plan-and-solve trace [11]. The ReAct framework interleaves rationales with executable actions, such as web searches, so each step is transparent [12]. Tree-of-Thoughts extends this idea to a branching search that can be audited after inference [13]. Extending plain text, ReasonGraph visualizes reasoning paths as interactive flow diagrams [14], and GraphReason converts multiple CoT traces into a unified graph to detect contradictions [15]. Self-Refine lets the model critique and iteratively refine its answers, exposing an evolution of thought that humans can inspect [16].

Despite these proposals, existing work has been evaluated mainly on synthetic Question Answering or programming tasks. None targets the ambiguity, safeguarding needs, and workflow constraints of school health communication. The proposed Prompt-Graph DSL addresses this gap by converting enriched student statements into a deterministic dialogue graph rendered for nurse monitoring, unifying automatic prompt expansion with visual explainability.

III. Proposed Method

We aimed to develop an automated AI chatbot that reduces the burden on school nurses by capturing various student inputs and enabling nurses to efficiently evaluate and monitor the chatbots' behavior.

The design is guided by three core principles:

1) *Minimal cognitive load*: Students should be able to convey their condition using brief, natural-language utterances, without requiring specialized prompting.
2) *Human transparency*: Nurses must understand why the model responds in a particular way.
3) *Auditability*: Every reasoning step must be reproducible from stored artifacts; no specialized prompting is required.

To satisfy these principles, we propose a two-layer architecture (Figure 1). Layer 1 automatically expands vague prompts into structured representations. Layer 2 deterministically maps these representations onto a declarative decision graph and renders the selected path for nurse supervision. Collectively, these layers enable:

1) the generation of richer, more informative responses from limited input, and
2) the provision of structured, multi-slot symptom representations for supervising nurses.

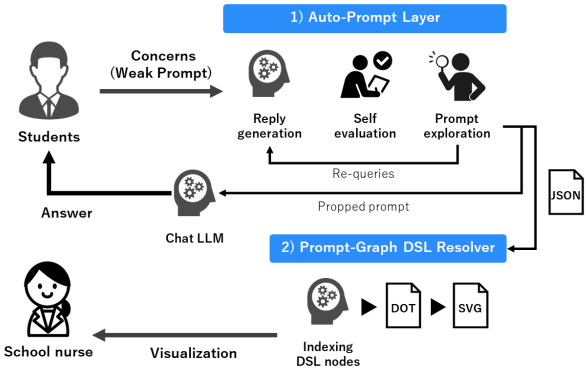

Figure 1. Overview of the proposed method.

A. Problem Statement

Given a short, often vague student utterance u, we aim to produce (i) a nurse-safe reply r, (ii) a slot vector $J = (sym, i, d, c)$, and (iii) a path P on a nurse-auditable Prompt-Graph. The following mathematical expressions do not represent optimization procedures in the implementation; they function as compact design abstractions for prompt engineering. The conceptual search dynamics operate on an explicit state graph, which we model as a beam search over states by drawing inspiration from Grice's cooperative principle in conversation [17]. When information is missing, the system applies the Quantity maxim, which prompts clarification. When the tone is overly rigid, the Manner maxim guides adjustments to improve clarity and communicative ease [18]. In practice, this search is implemented using iterated template prompts; no vector arithmetic or gradient-based optimization is involved.

Each prompt engineering intuition is mapped to three formal objects:

State
$s = \langle u, J \rangle$ — current system prompt u and the partially filled slot vector J.

Operator
Γ = {APPEND_ASK, APPEND_EMPATHY, APPEND_SAFE} — add a clarifying question / add an empathy phrase / prepend a safety reminder.

Score
$\text{Score} = w_c\, C + w_e\, E + w_s\, S$ (1) — weighted sum of Coverage, Empathy, and Safety.

The auxiliary quantities are defined as follows:

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org

$coverage(J)$
$C := coverage(J) = \frac{\#\text{ filled slots}}{4} \in [0, 1]$. Full coverage, therefore, yields $C = 1$.

$safe(r)$
$S := safe(r) \in \{0, 1\}$, returns 1 if the reply r satisfies all safety filters else 0.

E Cosine similarity between the reply embedding and a fixed "empathy prototype" vector.

w_c, w_e, w_s
Non-negative weights are chosen empirically on a validation set.

τ The score threshold that a candidate must reach to be accepted; tuned once per deployment.

For "My head feels heavy and a little queasy.", the first pass yields sym = headache, i = mild, c = nausea, $d = \varnothing$ ($coverage = 0.75$). Operator ASK requests duration, completes J, and the resolver maps J deterministically to a node P (e.g., mild headache+nausea, duration $<$ 1h).

Our two-layer pipeline realizes a function

$$F : \mathcal{U} \longrightarrow \mathcal{R} \times \mathcal{J} \times \mathcal{P} \tag{2}$$

where $\mathcal{U}$ is the space of student utterances and $\mathcal{R}$ the space of chatbot replies, $\mathcal{J}$ is the space of complete slot vectors, and $\mathcal{P}$ is the set of paths in the Prompt-Graph. This function assigns each utterance $u \in \mathcal{U}$ to a triple (r, J, P), where

- $r \in \mathcal{R}$ is the chatbot's reply,
- $J \in \mathcal{J}$ is the fully populated slot vector,
- $P \in \mathcal{P}$ is the path selected in the Prompt-Graph.

B. Layer 1: Auto-Prompt Expansion

In this layer, the system prompt u and the partial slot vector J are used to update the current state. Here, $sym \in \mathcal{S}$ denotes the symptom, $i \in (\text{mild}, \text{moderate}, \text{severe})$ represents intensity, $d \in \mathbb{R}_{>0}$ indicates duration in minutes, and c denotes a set of co-symptoms. Each state corresponds to an entry in the Knowledge Base, and operator effects are realized through system prompt substitutions.

After each LLM call, the system parses the reply to update the slot vector J and recompute the auxiliary variables C, E, and S. The same acceptance criterion, $\text{Score} \geq \tau$ is applied. Let ψ denote the final sequence of system prompts obtained at termination. Upon success, the layer outputs: (i) the consolidated JavaScript Object Notation (JSON) record $J^\star$ and (ii) the dialogue trace $\langle \psi, r, \text{Score} \rangle^\star$, both of which are passed to Layer 2 for path visualization.

C. Layer 2: Prompt-Graph DSL Resolver

Given $J^\star$, the resolver evaluates predicates in breadth-first order to identify the first matching node. Given that the predicates are mutually exclusive by design, the resolver operates deterministically. Each edge carries either a follow-up question or an action suggestion. The $J^\star \mapsto P$ is logged to support offline replay and error analysis.

The resolver exports the subgraph $G_P = (V_P, E_P)$, induced by all the nodes within two hops (i.e., children and grandchildren) of the current node. The view is updated at every turn, thereby providing nurses with situational awareness while respecting the simplicity constraints of the kiosk environment.

IV. Implementation

Figure 2 illustrates the architecture of the digital school nurse system as deployed in the pilot environment. The system employs a service-oriented design, wherein independent microservices provide conversational intelligence, prompt expansion, and graph visualization. Developed using Python 3.11 and orchestrated by Dify, the system integrates three key open-source components: ChatdollKit for the animated student interface, Dify for LLM orchestration, and a custom visualization module for nurse-side, node-level feedback. This section details the operational runtime cooperation among these services, in accordance with the requirements outlined in Section III.

A. Conversation Pipeline

The conversation pipeline executes as follows:

1) **Student Action**: A student verbally interacts with the digital nurse avatar. The system captures the speech, converts it to text, and transmits it to the Dify chat API.
2) **Prompt Enhancement:** An agent (o3-mini) rewrites the system prompt based on the rules defined in Prompt 1.
3) **Self Evaluation**: The generated response is scored by GPT-4o-mini using the evaluation prompt in Prompt 2. If the score falls below a predefined threshold, steps 2 and 3 are iterated up to three times until the threshold is met.
4) **Response Generation**: Dify sends the enhanced prompt and a system template (Prompt 3) to GPT-4o to produce the final reply.
5) **Slot Extraction**: Using the enhanced prompt, GPT-4o-mini extracts key information into a JSON schema as detailed in Prompt 4. These JSON objects are logged to a JSON Lines (JSONL) file for historical review.
6) **Prompt-Graph Rendering**: The resulting JSON record is forwarded to the Prompt-Graph Resolver, where a Python script converts it to a NetworkX graph via YAML, extracts the relevant subgraph, and renders it as an SVG file using Graphviz.
7) **Output Delivery**: The validated reply is sent back to ChatdollKit to be vocalized by the avatar, while the corresponding SVG visualization is simultaneously embedded into the nurse's monitoring dashboard.

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org

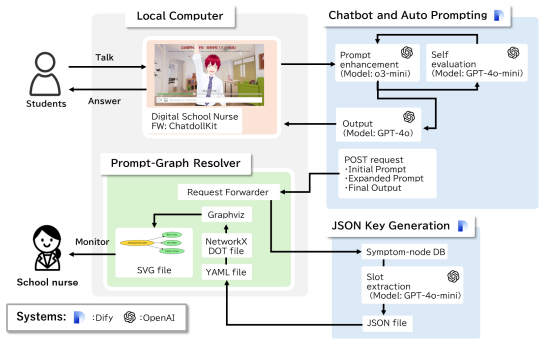

Figure 2. System overview integrating an animated avatar, LLM orchestration, and nurse-facing Prompt-Graph visualization.

```
You are a **Prompt-Expansion Agent**.
Your task is to transform each utterance of health care
     from a Japanese school student into a richer, natural-
     sounding Japanese sentence or short paragraph that
     fully surfaces the student's meaning so later modules
     can answer more accurately.
For every input line:
1. Rewrite it in Japanese, incorporating any information
     that is already present about:
- When it happened, started, or ended
- Where it took place
- What exactly happened or is needed
- Why it is a problem or need
- How it is unfolding or how the student feels
(Include **Who** only if the original utterance mentions
     people explicitly.)
2. Do **not** invent facts or add content that is absent
     from the original utterance.
3. Keep the output plain-no numbering, bullet lists,
     roleplay tags, or JSON.
Return **only** the enhanced Japanese sentence or short
     paragraph.
You must answer in Japanese.
```

Prompt 1: Expansion prompt for utterance enrichment by o3-mini

```
You are a **Prompt-Expansion Agent**.
For each single-line Japanese utterance from a middle-
     school student (``raw input''):

1. Produce an **enhanced Japanese sentence or short
     paragraph** (``enhanced'') that restates the raw input
      and, where the original wording already contains it,
     explicitly includes:
   - When (timing / duration)
   - Where (location)
   - What (event, symptom, request, etc.)
   - Why it matters to the student
   - How the situation is unfolding or how the student
        feels
   (Add **Who** only if it is explicitly mentioned.)
   *Never invent details that are not in the raw input.*
2. **Self-check** the ``enhanced'' text and assign
     quantitative scores (0.00-1.00, two decimals):
   - **coverage** = fraction of the five W-H elements
        present
   - **empathy** = degree of empathy conveyed toward a
        young student
   - **safety** = 1.00 if the text contains **no** medical
         diagnosis, drug names, or prescriptive medical
        advice; otherwise 0.00
   If **coverage < 0.75** **or** **safety < 1.00**,
        rewrite the ``enhanced'' text once, then
        recalculate the scores and use the improved version
        .
3. **Output format (exactly two lines, no extra text)**
   {{ENHANCED}}
   [coverage={{COV}}, empathy={{EMP}}, safety={{SAFE}}]
   Replace the placeholders with the final enhanced text
        and the three numeric scores.
   Do **not** add numbering, bullet points, role-play tags
        , or JSON.
```

Prompt 2: Self-evaluation prompt for prompt quality scoring by GPT-4o-mini

```
You are a friendly school nurse assistant.
Collect four items: symptom, intensity (mild/moderate/
     severe),
duration (minutes or hours), and co-symptoms (comma
     separated).
Ask only one clarifying question at a time.
Avoid medical diagnoses and medication names.
```

Prompt 3: Chatbot prompt for structured response generation by GPT-4o

```
{
"symptom": "headache",
"intensity": "mild",
"duration": "15min",
"coSymptom": "nausea"
}
```

Prompt 4: Slot extraction prompt for filling the JSON schema by GPT-4o-mini

B. *Prompt-Graph Resolver and Visualization*

At startup, it parses the YAML file converted from JSON into a Pydantic object tree and constructs a directed multi-graph using NetworkX. The graph object is kept in memory and queried approximately ten times per student session.

Each traversal event is serialized as $\langle t, J^{\star}, n_{\text{prev}}, n_{\text{next}} \rangle$ and appended to a compressed JSONL file for offline analytics. For visual feedback, the resolver extracts the sub-graph reachable within two hops of the current node, exports it to DOT format, and invokes Graphviz in headless mode to export SVG.

C. *Use Case of Implemented System*

Figure 3 shows the system in use during its in-situ pilot deployment. Positioned within the school infirmary, the kiosk invites students to interact with the animated digital nurse avatar.

V. Performance Evaluation

A. *Procedure*

A corpus of 50 student complaints was prepared: 20 were collected from handbooks and public websites intended for school nurses, and 30 ambiguous examples were mined from social media and public question-and-answer platforms. For each utterance, we created three gold-standard artifacts: (1) an enhanced prompt that rephrased the complaint in a more informative context, (2) an exemplary nurse response, and (3) a four-slot JSON record capturing symptom, intensity,

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org

Figure 3. Students interacting with the deployed system.

duration, and coSymptom. These artifacts were initially corrected using GPT-4o-mini with a simple prompt (shown in Prompt 5) to fix typos, missing characters, and expressions difficult for the model to interpret. The creator then manually verified that the original intent remained unchanged, thereby finalizing the gold-standard dataset.

All model runs followed the configurations shown in Figure 2. Prompt expansion was conducted at a temperature of 0.7, whereas slot extraction used a temperature of 0.1 for higher determinism. Prompt fidelity was evaluated using standard confusion matrix metrics—accuracy, precision, recall, F1 score—where a cosine similarity of ≥ 0.70 (measured using Sentence-Bidirectional Encoder Representations from Transformers (BERT), all-mpnet-base-v2) was considered a true positive. Answer quality was assessed by GPT-4o using a rubric-based prompt (Prompt 6), with scores from 1–5 assigned for accuracy, completeness, and tone. JSON slot extraction was evaluated using exact string matches per slot, with True Positives (TP) for correct values, False Negatives (FN) for missing or incorrect values, and False Positives (FP) for spurious outputs. All utterances used for evaluation were drawn from publicly available materials or anonymized examples; no identifiable student data were collected.

B. Results

1) Prompt Expansion: The enhanced prompts achieved 41 TP and 9 FN, yielding an F1 score of 0.82. No hallucinated slot values were observed below the similarity threshold. This indicated that most errors stemmed from partial omissions rather than from fabrication.

2) Answer Quality: Rubric-based evaluation produced mean scores of 3.63 for accuracy, 3.71 for completeness, and 4.73 for tone on a five-point scale. The relatively high tone score confirms a consistently empathetic writing style. In contrast, the lower accuracy and completeness scores, which were manually verified, revealed gaps in medical specificity and follow-up guidance that require further prompt engineering.

3) JSON Slot Extraction: Using the lower temperature setting, the system produced 39 TP slots, 56 FP slots, and 47 FN slots, which resulted in an F1 score of 0.43. Most of the remaining errors stemmed from lexical variation in location terms and inconsistent mapping of free-text severity phrases to the three-level scale.

```
Please correct typos, missing characters, and expressions
     that may be difficult for machines to interpret. Do
     not change the meaning or intent. Keep the original
     wording as much as possible. The input and output text
      is in Japanese.
```

Prompt 5: Proofreading prompt for output correction by GPT-4o-mini

```
SYSTEM_MSG = (
"You are a strict school nurse evaluator. Given a gold
     reference answer and "
"a candidate answer, rate the candidate on Accuracy,
     Completeness, and Tone "
"on a scale from 1 (poor) to 5 (excellent). "
)
USER_TEMPLATE = (
"GOLD:\n{gold}\n\nCANDIDATE:\n{pred}\n"
)
```

Prompt 6: Rubric-based evaluation prompt for answer scoring by GPT-4o

VI. Discussion

A. Quantitative Findings from Performance Evaluation

The results in Section V indicate that the proposed pipeline effectively captures student intent through prompt expansion by achieving an F1 score of 0.82. This suggests that the model can reliably augment vague student utterances into structured representations when supported by an auto-prompt expansion. Conversely, structured slot extraction remains a significant bottleneck, with an F1 score of 0.43. A detailed error analysis revealed that most FPs and FNs were attributable to lexical variations in symptom and severity expressions. The system attained a precision of 0.41 and a recall of 0.45, suggesting that the current model struggles to consistently identify and normalize the key elements of the complaint. To address this limitation, additional normalization techniques such as controlled vocabularies, synonym clustering, or regular expression filters may be necessary.

Moreover, rubric-based evaluations indicate that while the consistently high tone scores confirm the system's ability to generate empathetic responses, the lower accuracy and completeness scores expose deficiencies in medical specificity and the inclusion of appropriate follow-up guidance.

B. Qualitative Feedback from Pilot Deployment

Qualitative observations during the pilot deployment revealed promising user acceptance and workflow integration. Students reported that the animated avatar was "less intimidating than talking to an adult right away," and many appeared to be more willing to disclose emotional or ambiguous symptoms. This is particularly valuable in the context of adolescent mental health, where emotional or social barriers often hinder verbal expression.

From the school nurses' perspective, the visual decision graph rendered on the Scalable Vector Graphics (SVG)

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org

dashboard enabled passive oversight. This form of transparent feedback preserved the nurse's supervisory role while minimizing cognitive overhead.

Notably, this qualitative evaluation focused on user interaction and interface design rather than on the diagnostic capabilities of the proposed pipeline. The deployed system was a prototype built on GPT APIs, without integration of the auto-prompt expansion or Prompt-Graph resolver described in this study. Therefore, the feedback should be interpreted as a formative assessment of interaction design, not a summative evaluation of system performance.

C. Limitations

This study is a controlled pilot with simulated student inputs, so external validity remains limited. We do not include external baselines in the camera-ready; instead, we release the prompts, scoring rubric, and slot schema to support replication. Slot extraction currently relies on LLM generalization without domain-specific post-editing, hence structured outputs are not yet suitable for unsupervised clinical use. The qualitative evaluation is small and short-term and does not assess long-term behavior or health outcomes. Near-term work targets synonym normalization, ordinal severity mapping, lightweight post-editing rules, a small supervised adapter, and per-slot error analysis.

VII. Conclusion and Future Work

This study introduced a two-layer framework designed to mitigate the cognitive load on school nurses by transforming ambiguous student complaints into structured, explainable dialogue flows. Our evaluation demonstrated that the auto-prompt expansion layer effectively enriches vague inputs, achieving a high F1 score of 0.82. However, the subsequent slot extraction process remains a challenge, with an F1 score of 0.43, highlighting issues with lexical variability in student expressions. While AI-based rubric evaluations confirmed the system's ability to generate empathetic responses, they also revealed deficiencies in medical specificity and follow-up guidance, underscoring the need for further refinement.

Future work will concentrate on enhancing clinical reliability and robustness. Key priorities include the implementation of controlled vocabularies and synonym normalization to improve the precision of slot extraction. Furthermore, a long-term field study is necessary to validate the system's real-world effectiveness, assess its impact on nurse decision-making, and understand how student interactions evolve over time. Through these efforts, we aim to develop a clinically reliable tool that enhances adolescent health support in educational settings.

Acknowledgment

This work was supported by JST RISTEX Japan Grant Number JPMJRS24K3, the Japan Health Foundation, the I-O DATA Foundation, and the Sasakawa Scientific Research Grant from the Japan Science Society. In this work, authors marked with † (Hayato Tomisu and Kazue Yamamura) contributed equally as co-first authors. We would like to thank Editage (www.editage.jp) for English language editing.

References

[1] K. Yamamura, "Current Situation of Visits to the Health Office during the Corona Disaster （コロナ禍における保健室来室の現状）," *Journal of the Japanese Association for the Study of Guidance*, vol. 40, pp. 11–17, 2023.

[2] T. B. Brown *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[3] T. Shin *et al.*, "AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts," in *Proc. EMNLP*, 2020, pp. 4222–4235.

[4] G. Qin and J. Eisner, "Learning How to Ask: Querying Language Models with Mixtures of Soft Prompts," in *Proc. NAACL*, 2021, pp. 5203–5212.

[5] M. Deng *et al.*, "RLPrompt: Optimizing Discrete Text Prompts with Reinforcement Learning," in *Proc. EMNLP*, 2022, pp. 9593–9611.

[6] A. Prasad *et al.*, "GrIPS: Gradient-Free, Edit-Based Instruction Search for Prompting Large Language Models," in *Proc. EACL*, 2023, pp. 3083–3099.

[7] Z. Zhang *et al.*, *Automatic chain of thought prompting in large language models*, arXiv:2210.03493, 2023.

[8] Y. Zhou *et al.*, *Large language models are human-level prompt engineers*, arXiv:2211.01910, 2023.

[9] J. Wei *et al.*, "Chain-of-Thought Prompting Elicits Reasoning in Large Language Models," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 24 824–24 837.

[10] X. Wang *et al.*, "Self-Consistency Improves Chain-of-Thought Reasoning in Language Models," in *Proc. ICLR*, 2023. DOI: 10.48550/arXiv.2203.11171.

[11] D. Zhou *et al.*, "Least-to-Most Prompting Enables Complex Reasoning in Large Language Models," in *Proc. ICLR*, 2023.

[12] S. Yao *et al.*, "ReAct: Synergizing Reasoning and Acting in Language Models," in *Proc. ICLR*, 2023.

[13] S. Yao *et al.*, "Tree-of-Thoughts: Deliberate Problem Solving with Large Language Models," in *Advances in Neural Information Processing Systems*, vol. 36, 2023.

[14] Z. Li *et al.*, *ReasonGraph: Visualization of Reasoning Paths*, arXiv:2503.03979, 2025.

[15] L. Cao, "GraphReason: Enhancing Reasoning Capabilities of Large Language Models Through a Graph-Based Verification Approach," in *Proc. ACL Workshop on Natural Language Reasoning and Structured Explanations*, 2024, pp. 12–24.

[16] A. Madaan *et al.*, "Self-Refine: Iterative Refinement with Self-Feedback," in *Advances in Neural Information Processing Systems*, vol. 36, 2023.

[17] P. Grice, "Logic and conversation," in *Syntax and Semantics*, P. Cole and J. L. Morgan, Eds., vol. 3, New York, NY, USA: Academic Press, 1975, pp. 41–58.

[18] H. H. Clark and S. E. Brennan, "Grounding in communication," in *Perspectives on socially shared cognition*, L. B. Resnick *et al.*, Eds., Washington, DC, USA: American Psychological Association, 1991, pp. 127–149.

Courtesy of IARIA Board and IARIA Press. Original source: ThinkMind Digital Library https://www.thinkmind.org