

Prompting as Measurement: Enhancing Coding for Systematic Reviews with Large Language Models

Dandan (Danielle) Chen Kaptur¹, Yue (Katherine) Huang², Yanhui Guo³, Xuejun Ryan Ji⁴

¹Pearson, ²Measurement Incorporated, ³University of Illinois Springfield, ⁴Medical Council of Canada

Correspondence: danielle.chen@pearson.com

Abstract

Prompting functions as a measurement condition that alters LLM coding behavior. We examined whether few-shot prompting improves GPT-based qualitative coding in systematic reviews. Contrary to expectations, we found that the zero-shot approach produced the highest agreement with human coders, suggesting increasing conservatism as more prompt structure was added.

1 Introduction

Systematic reviews play a central role in synthesizing evidence, yet their implementation remains labor-intensive and cognitively demanding (Trad et al., 2025; Galli et al., 2025). A particularly critical step involves coding study characteristics from full-text articles according to predefined protocols. This process requires not only extracting relevant information but also making nuanced judgments that depend on contextual interpretation and domain expertise. Recently, large language models (LLMs), such as GPT, have shown promise in supporting or automating aspects of systematic review workflows, particularly tasks involving text classification and evidence extraction (Hasan et al., 2024; Han et al., 2025).

However, the effectiveness of LLMs depends strongly on prompt design. In prompt engineering research, zero-shot and few-shot approaches are among the most commonly used methods for adapting models to new tasks without additional training (Sahoo et al., 2024). In zero-shot prompting, the model performs a task based solely on task instructions (Radford et al., 2019). In contrast, few-shot prompting provides examples of desired input–output mappings, allowing the model to infer task structure and decision criteria from demonstrations. Prior research suggests that example-based prompting can improve performance on tasks requiring structured reasoning and consistent inter-

pretation, particularly when task definitions are underspecified or involve latent constructs (Han et al., 2025).

These issues are especially relevant to qualitative coding in systematic reviews, where protocol terms such as “substantial” or “evidence-based” function as constructs that require interpretation rather than simple information retrieval. Without explicit guidance, zero-shot prompting may lead to variability in how such constructs are understood and applied (Hasan et al., 2024). By providing definitions and examples, few-shot prompting may anchor the model’s interpretation and produce more consistent coding decisions (Han et al., 2025). At the same time, examples may alter the model’s decision-making process by shifting it from instruction-based inference to pattern-guided reasoning, raising important questions about how prompt design shapes coding behavior (Adam et al., 2026).

In this study, we examine how prompt design influences qualitative coding using GPT-4o, building on a prior pilot study (Chen Kaptur et al., 2025) that identified substantial performance differences across coding-question types. We compare GPT-generated and human-generated binary codes under three conditions corresponding to zero-shot and few-shot approaches. Framed as a measurement problem, the study treats prompt design as a measurement condition that may influence how coding constructs are interpreted and applied. Through this comparison, we investigate whether changes in prompt structure alter coding decisions for questions that require interpretive judgment within a systematic review context.

2 Method

2.1 Design

We evaluated the consistency between GPT-generated and human-generated codes under three

prompting conditions. Condition 1 represents a zero-shot prompting approach in which GPT relies solely on the coding questions as instructions. Condition 2 represents a few-shot prompting approach that augments the prompt with demonstration examples derived from expert-coded articles. Condition 3 represents an enhanced few-shot prompting approach that further incorporates explicit construct definitions into the prompt in Condition 2.

The qualitative coding task involves generating binary responses (1 = yes, 0 = no) to coding questions from an established systematic review protocol (Chen et al., 2023). Based on pilot findings indicating that GPT produced the strongest overall performance among the models tested and that batch size had limited effects on outputs (Chen Kaptur et al., 2025), we selected GPT-4o as the LLM for generating codes and fixed the batch size at 1 such that each article–question pair was processed independently.

2.2 Data

We used the 32 articles retained in the final stage of the systematic review conducted by Chen et al. (2023) to generate qualitative codes in both Conditions 1 and 2, in addition to five articles as examples for Condition 2. The articles examine the use of technology-based assessments across educational systems in more than 30 countries and were published between 2018 and 2022. These articles were parsed from the PDF format into the plain text format to ensure compatibility with the language model input requirements.

Binary responses were generated for each of the 32 articles to Questions 6–11 in an established protocol of the same study (Chen et al., 2023), resulting in a total of 192 coded outputs (32 articles $\times$ 6 questions). We focus on Questions 6–11 because they ask whether an article provides a “substantial” amount of relevant content. Unlike Questions 1–5, which function primarily as filtering items, Questions 6–11 require more interpretive, content-based judgments and are therefore likely to be more sensitive to prompting differences. This distinction is supported by the pilot study (Chen Kaptur et al., 2025), in which Questions 1–5 showed relatively high human–machine agreement, whereas Questions 6–11 exhibited lower accuracy and greater variability under zero-shot prompting. Appendix A lists out all the coding questions from the original study.

2.3 Prompting

GPT Settings. All the three conditions were implemented under controlled settings using the API connected to GPT-4o, with outputs generated by a single coder, having the temperature parameter was fixed at 0 to promote deterministic responses.

Prompting Structure. For each condition, GPT was given the parsed article text and coding question and was asked to return formatted coding results, which are binary codes (1 = yes, 0 = no) to coding questions. The prompt included the following components intended to mirror human reasoning in systematic review coding:

1. *A role description* (specifically, “You are an education research assistant supporting a systematic review. Your task is to answer predefined coding questions using structured reasoning and direct evidence from parsed article text.”)
2. *A task description* (specifically, “You will be given: a parsed full-text from a peer-reviewed article, and a few questions. Respond with 1 for yes and 0 for no.”)

Condition 1. Using the zero-shot prompting approach, the model received only the article text, coding question, and formatting requirements. It was expected to infer the operational meaning of each coding construct based on the question wording and the article content. This condition served as the baseline for comparison with Conditions 2–3.

Condition 2. This condition employed a few-shot prompting approach that extended the zero-shot prompt by including demonstration examples of correct coding practice. In addition to the article text, coding question, and formatting requirements, the prompt contained five demonstration articles drawn from prior systematic review training materials associated with the same study (Chen et al., 2023). These articles were not part of the 32-article evaluation set. For each coding question, the model was provided with a human-generated binary answer, a three-step reasoning explanation, supporting textual evidence, and a confidence rating ranging from 1 to 5. These examples were intended to provide concrete guidance for mapping article content to coding decisions.

Condition 3. This condition employed an enhanced few-shot prompting approach that extended the prompting approach in Condition 2 by including an explicit definition of the key construct “sub-

stantial”, which appeared in each coding question. Providing this definition was intended to reduce ambiguity in interpretation and improve consistency in GPT-generated responses. Specifically, an article was classified as containing a “substantial” amount of information relevant to a coding question (coded as 1) if it met at least one of the following criteria: (a) contained two or more relevant paragraphs, (b) contained five or more relevant sentences, (c) included at least two predefined keywords, or (d) contained at least one predefined depth item. Keywords and depth items are provided in Appendix B.

2.4 Analysis

Model performance was evaluated for each condition using accuracy, precision, recall, F1, and quadratic weighted kappa (QWK), computed from comparisons between GPT-generated and human-generated codes. Analyses were conducted at three levels. First, we calculated aggregated metrics across all articles and all six coding decisions per condition. Second, we computed by-question metrics per condition, averaging results across articles for each coding question. Third, we computed by-article metrics per condition, averaging results across coding questions for each article. Using these metrics, we evaluated and interpreted differences among Conditions 1-3 as effects of the measurement condition (i.e., prompt design) on GPT-generated coding decisions.

3 Results

3.1 Overall Results Across Questions and Articles

Table 1 presents the aggregated comparison across Questions 6–11 and all articles. Overall, Condition 1 showed the strongest performance across the evaluated metrics. Conditions 2 and 3 produced nearly identical performance, not showing additional gains beyond Condition 1. The results suggest that the zero-shot prompting approach already captured as much agreement with the human-coded benchmark and that Condition 1 is the best, contrary to what is expected. The largest change across conditions was a substantial decrease in recall (.493 to .418 to .328), indicating increasing conservatism as more prompt structure was added.

3.2 By-Question Results

Table 2 and Figure 1 present the question-level comparison across Conditions 1–3. At the question

Condition	<i>N</i>	Accuracy	Precision	Recall	F1	QWK
C1	192	0.724	0.635	0.493	0.555	0.359
C2	192	0.698	0.596	0.418	0.491	0.286
C3	192	0.688	0.595	0.328	0.423	0.233
Difference C2–C1	–	-0.026	-0.039	-0.075	-0.064	-0.073
Difference C3–C1	–	-0.036	-0.040	-0.165	-0.132	-0.126
Difference C3–C2	–	-0.010	-0.001	-0.090	-0.068	-0.053

Table 1: Aggregated: Conditions 1–3.

level, Condition 1 showed the clearest consistency for accuracy: it was either the highest-performing condition or tied for the highest-performing condition across all six questions. For the other metrics, the pattern varied by question. Condition 2 showed localized advantages, including higher recall and F1 for Q6 and higher QWK for Q8. Condition 3 showed localized advantages for Q7, where it had the highest recall, F1, and QWK, and it also had the highest precision for Q6 and Q8.

Overall, the question-level results suggest that Conditions 2 and 3 did not provide consistent performance gains beyond Condition 1. Their advantages were limited to specific questions and specific metrics rather than appearing across the full set of coding questions. Thus, while additional prompt modifications changed the model’s coding behavior for some questions, these changes did not consistently improve agreement with the human-coded benchmark. Condition 1 therefore appeared to provide the most stable question-level performance pattern, especially for accuracy, while the remaining metrics showed more question-specific variation across conditions.

3.3 By-Article Results

Table 3 and Figure 2 present article-level performance across Conditions 1–3. The article-level results showed substantial variation, indicating that performance depended not only on prompt condition but also on article-specific features. Compared with Condition 1, Condition 2 showed higher accuracy for 7 articles, lower accuracy for 11 articles, and equal accuracy for 14 articles. Condition 3 showed higher accuracy for 6 articles, lower accuracy for 10 articles, and equal accuracy for 16 articles. Thus, Condition 1 showed the clearest overall pattern for accuracy, although many articles had equal accuracy across conditions.

For the other metrics, the article-level pattern was more mixed. Compared with Condition 1, Condition 2 showed higher F1 for 6 articles, lower F1 for 12 articles, and equal F1 for 14 articles. For

Question	Condition	Acc.	Prec.	Rec.	F1	QWK
Q6	C1	0.656	0.714	0.357	0.476	0.261
	C2	0.562	0.500	0.714	0.588	0.152
	C3	0.625	1.000	0.143	0.250	0.158
Q7	C1	0.688	0.611	0.786	0.688	0.385
	C2	0.688	0.667	0.571	0.615	0.355
	C3	0.688	0.600	0.857	0.706	0.394
Q8	C1	0.625	0.714	0.556	0.625	0.262
	C2	0.625	0.800	0.444	0.571	0.284
	C3	0.562	0.833	0.278	0.417	0.188
Q9	C1	0.688	0.000	0.000	0.000	0.000
	C2	0.688	0.000	0.000	0.000	0.000
	C3	0.656	0.000	0.000	0.000	-0.060
Q10	C1	0.906	1.000	0.400	0.571	0.529
	C2	0.875	1.000	0.200	0.333	0.297
	C3	0.875	1.000	0.200	0.333	0.297
Q11	C1	0.781	0.455	0.833	0.588	0.456
	C2	0.750	0.250	0.167	0.200	0.059
	C3	0.719	0.286	0.333	0.308	0.133

Table 2: By-question: Conditions 1–3.

QWK, Condition 2 was higher for 7 articles, lower for 11, and equal for 14. Condition 3 showed a similar pattern: it had higher F1 for 7 articles, lower F1 for 10, and equal F1 for 15; for QWK, it was higher for 8 articles, lower for 11, and equal for 13. These results suggest that Conditions 2 and 3 produced localized improvements for some articles, but these improvements were not consistent across the full article set or across metrics.

Figure 2 further shows that the magnitude of differences across conditions varied by article and metric. Some articles showed little or no difference across conditions, whereas others showed larger gaps, especially for recall, F1, and QWK. Taken together, the article-level results suggest that the additional prompt modifications in Conditions 2 and 3 changed model behavior for some articles, but did not produce a consistent improvement over Condition 1. Condition 1 therefore appeared to provide the most stable article-level pattern overall, while Conditions 2 and 3 showed more article-specific variation.

3.4 Summary of findings

Across the aggregated, question-level, and article-level analyses, Condition 1 showed the strongest overall performance pattern among the three tested conditions, especially in the aggregated results and in question-level accuracy. Conditions 2 and 3 showed some localized advantages for specific questions or articles, but these did not translate into consistent improvements across metrics or levels of analysis. Overall, the additional prompt modifications changed model behavior in some cases

but did not provide further consistent performance gains beyond Condition 1.

Taken together, the specific few-shot and the key-construct enhanced prompt configurations tested in this study did not produce consistent performance gains across metrics or levels of analysis. However, their question- and article-specific advantages suggest that these prompt features may work better under certain coding conditions. Future work should further examine which question characteristics and article characteristics are associated with improved LLM coding performance under different prompt configurations.

4 Discussion

The present study examined whether GPT-based qualitative coding benefits from moving beyond zero-shot prompting to a few-shot prompting framework that includes both structured definitions and demonstration examples. In systematic review tasks, where the target is not simply topic classification but expert-aligned judgment under uncertainty, the difference between zero-shot and few-shot prompting can be understood as the difference between unconstrained inference and guided evidentiary reasoning. Our results suggest that, contrary to expectations (e.g., [Brown et al., 2020](#)), the few-shot prompt structure did not improve coding performance. Across aggregate metrics, the highest levels of accuracy, F1, and agreement with human coders were observed under the zero-shot condition, whereas both few-shot conditions produced lower performance. These findings suggest that prompt engineering is not uniformly beneficial and

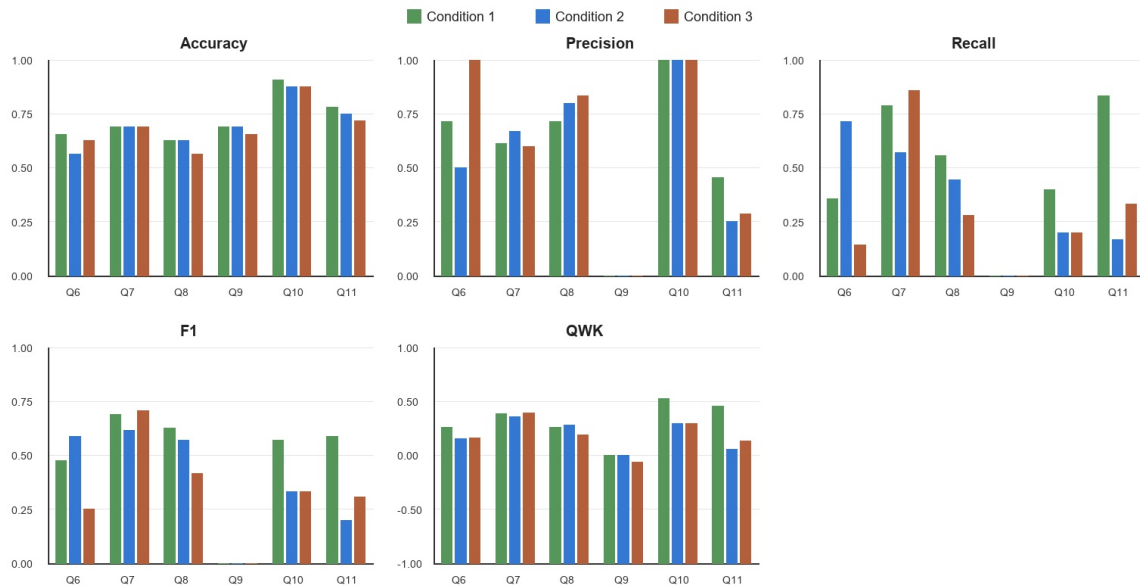


Figure 1: Plots By-question: Condition 1 vs. Condition 2 vs. Condition 3.

that providing additional guidance may sometimes constrain model behavior in ways that reduce rather than improve coding quality.

From a psychometric perspective, these findings support the view that qualitative coding can be conceptualized as a measurement task. Each coding question functions as an observed indicator of an underlying construct (e.g., Embretson and Reise, 2000), and the language model serves as a rater that translates textual evidence into categorical judgments. Metrics such as Accuracy, Precision, Recall, and F1 capture different aspects of decision quality, while QWK reflects agreement with human coding in a manner analogous to inter-rater reliability. Under this framework, prompt design can be viewed as a measurement condition that influences how the model interprets constructs and applies decision rules (e.g., Messick, 1989). The decline in QWK from Condition 1 to Conditions 2 and 3 suggests that adding examples and construct definitions altered the coding behavior of the model in ways that reduced agreement with human raters, even when the intended goal was greater standardization.

The results also provide insight into how demonstration examples shape task interpretation. A common assumption in prompting research is that few-shot examples help align model outputs with desired decision rules by reducing ambiguity. In the

present study, however, the addition of examples and explicit construct definitions did not improve performance. Instead, progressively richer prompts were associated with lower recall and lower overall agreement. This pattern suggests that the examples may have encouraged a narrower or more restrictive interpretation of the coding criteria. Rather than increasing the model’s ability to identify relevant content, the examples appear to have shifted the model toward a more conservative decision strategy, resulting in fewer positive classifications and a corresponding decline in sensitivity.

The results also speak to a broader issue in prompting science—how examples shape task interpretation. In zero-shot settings, the model must infer the meaning of coding constructs from instructions alone. This may be sufficient for straightforward factual questions but less reliable for constructs that are underspecified, context-dependent, or threshold-based (e.g., whether evidence is “substantial”). Few-shot prompting can reduce this ambiguity by providing concrete instantiations of the intended decision rule. In effect, examples serve as a local calibration mechanism, anchoring the model’s interpretation of the protocol (e.g., Min et al., 2022). However, calibration is only beneficial if the examples appropriately represent the broader decision space. When examples emphasize particular interpretations or thresholds, they may

Article	Accuracy			Precision			Recall			F1			QWK		
	C1	C2	C3	C1	C2	C3	C1	C2	C3	C1	C2	C3	C1	C2	C3
A1	0.667	0.667	0.500	0.667	1.000	0.500	0.667	0.333	0.333	0.667	0.500	0.400	0.333	0.333	0.000
A2	0.333	0.167	0.333	0.500	0.333	0.500	0.500	0.250	0.250	0.500	0.286	0.333	-0.500	-0.667	-0.200
A3	1.000	0.833	1.000	1.000	0.500	1.000	1.000	1.000	1.000	1.000	0.667	1.000	1.000	0.571	1.000
A4	0.833	0.667	0.833	0.000	0.333	0.000	0.000	1.000	0.000	0.000	0.500	0.000	0.000	0.333	0.000
A5	1.000	0.667	0.500	1.000	1.000	0.000	1.000	0.333	0.000	1.000	0.500	0.000	1.000	0.333	0.000
A6	1.000	0.833	0.833	1.000	0.667	1.000	1.000	1.000	0.500	1.000	0.800	0.667	1.000	0.667	0.571
A7	0.667	1.000	0.500	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000	0.000	-0.200	1.000	-0.286
A8	0.833	0.500	0.500	0.800	1.000	0.667	1.000	0.250	0.500	0.889	0.400	0.571	0.571	0.182	0.000
A9	1.000	1.000	0.833	1.000	1.000	1.000	1.000	1.000	0.500	1.000	1.000	0.667	1.000	1.000	0.571
A10	0.833	0.667	0.833	0.500	0.000	0.500	1.000	0.000	1.000	0.667	0.000	0.667	0.571	-0.200	0.571
A11	0.667	0.833	0.500	1.000	1.000	1.000	0.500	0.750	0.250	0.667	0.857	0.400	0.400	0.667	0.182
A12	0.833	0.833	0.833	1.000	1.000	1.000	0.500	0.500	0.500	0.667	0.667	0.667	0.571	0.571	0.571
A13	0.667	0.667	0.500	0.500	0.000	0.000	0.500	0.000	0.000	0.500	0.000	0.000	0.250	0.000	-0.286
A14	0.833	0.833	0.667	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	-0.200
A15	0.833	0.833	0.833	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
A16	0.500	0.667	0.667	0.667	0.750	1.000	0.500	0.750	0.500	0.571	0.750	0.667	0.000	0.250	0.400
A17	0.833	0.833	0.667	1.000	1.000	1.000	0.667	0.667	0.333	0.800	0.800	0.500	0.667	0.667	0.333
A18	0.833	0.667	1.000	0.000	0.000	1.000	0.000	0.000	1.000	0.000	0.000	1.000	0.000	-0.200	1.000
A19	0.500	0.667	0.500	0.000	0.500	0.000	0.000	0.500	0.000	0.000	0.500	0.000	-0.286	0.250	-0.286
A20	0.833	0.667	1.000	0.500	0.333	1.000	1.000	1.000	1.000	0.667	0.500	1.000	0.571	0.333	1.000
A21	0.667	0.833	0.667	0.000	1.000	0.500	0.000	0.500	0.500	0.000	0.667	0.500	0.000	0.571	0.250
A22	0.500	0.500	0.500	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
A23	0.500	0.500	0.500	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
A24	0.833	0.833	0.833	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
A25	0.333	0.333	0.500	0.250	0.250	0.333	0.500	0.500	0.500	0.333	0.333	0.400	-0.200	-0.200	0.000
A26	0.667	0.833	0.833	0.500	1.000	0.667	1.000	0.500	1.000	0.667	0.667	0.800	0.400	0.571	0.667
A27	0.833	0.667	1.000	0.500	0.333	1.000	1.000	1.000	1.000	0.667	0.500	1.000	0.571	0.333	1.000
A28	0.500	0.667	0.500	0.333	0.000	0.000	0.500	0.000	0.000	0.400	0.000	0.000	0.000	0.000	-0.286
A29	0.833	0.667	0.833	0.500	0.333	0.500	1.000	1.000	1.000	0.667	0.500	0.667	0.571	0.333	0.571
A30	0.833	0.833	0.833	1.000	1.000	1.000	0.500	0.500	0.500	0.667	0.667	0.667	0.571	0.571	0.571
A31	0.500	0.500	0.500	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
A32	0.667	0.667	0.667	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table 3: By-article: Conditions 1–3.

unintentionally constrain the model’s reasoning and reduce its ability to generalize to new cases. From a measurement perspective, this can be viewed as a form of prompt-specific calibration effects: the model may appear more reliable under one prompting condition because the examples constrain its response space in ways that align better with the expert standard. This is a strength for applied coding, but also a reminder that performance is conditional on prompt design.

The by-question results further illustrate this phenomenon. Although some coding questions showed modest improvements under specific conditions, the overall pattern did not support a consistent advantage for either few-shot approach. The effects of prompt modifications varied substantially across questions, suggesting that the benefit of additional prompt structure depends on the characteristics of the construct being measured. Coding questions that require nuanced judgments about evidence sufficiency, methodological rigor, or threshold-based decisions may respond differently to prompt modifications than questions involving more direct factual classification.

Several limitations should be acknowledged. First, the study examined a single LLM and a specific set of coding tasks derived from a systematic review project. Second, the demonstration exam-

ples and construct definitions were drawn from the same coding protocol that served as the evaluation framework, which may limit generalizability to other domains. Third, only one set of examples and one operationalization of the construct definitions were evaluated. Different examples, alternative definitions, or a larger set of demonstrations may have produced different outcomes. Finally, agreement was evaluated relative to human-generated codes, which themselves are subject to measurement error and judgment variability. Future work should therefore examine how sensitive such gains are to the quantity, quality, and diversity of examples, and whether the apparent reliability of the model generalizes across review topics.

Despite these limitations, the study contributes to both prompting research and systematic review methodology by demonstrating that additional prompt structure does not necessarily improve coding performance. In this study, the simplest prompting condition produced the strongest overall results, while increasingly elaborate prompts resulted in lower recall, lower F1 values, and weaker agreement with human coders. These findings suggest that prompt design should be evaluated empirically rather than assumed to improve performance through the addition of examples or definitions. More broadly, the results suggest that

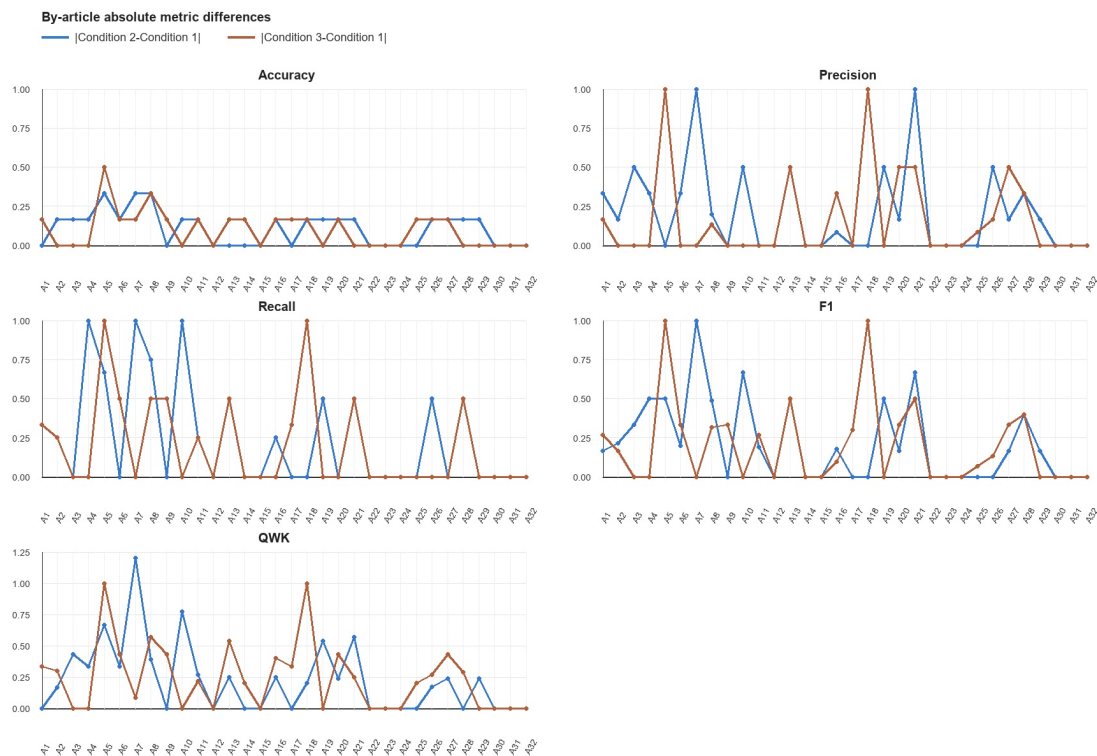


Figure 2: By article: |Condition 2 - Condition 1| vs. |Condition 3 - Condition 1|.

psychometric concepts such as construct representation, reliability, agreement, and decision thresholds provide a useful framework for evaluating how prompt design, as a measurement condition, shapes LLM-assisted evidence synthesis. As LLMs become more common in evidence synthesis workflows (O'Connor et al., 2024), understanding how prompts shape decision-making may be as important as understanding the capabilities of the underlying model itself.

Author Contribution

Dandan (Danielle) Chen Kaptur conceptualized the study, selected the systematic review instrument, conducted the literature review, and drafted the Introduction, Methods, and Discussion sections of the manuscript. Yue (Katherine) Huang and Xuejun Ryan Ji contributed to refining the research questions and study design, developed the analytic framework, and selected the statistical methods. Yanhui Guo developed the programming pipeline for generating LLM outputs. Yue conducted the statistical analyses, interpreted the results, and drafted the Results section. The authors used Microsoft Copilot to assist with literature discovery, drafting,

editing, and formatting. All the authors reviewed, revised, and approved the final manuscript.

References

- Till J. Adam, Salma A. S. Abosabie, Max Dittmer, Elise Wolf, Sara A. Abosabie, Clara Behnke, Felix Baier, Annabelle Weickmann, Ludwig Köser, Christoph U. Correll, and Niklas Rutsch. 2026. [Prompt engineering of large language models for paper screening in medical meta-analyses and systematic reviews: A prospective comparative study](#). *Research Synthesis Methods*, pages 1–18.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. [Language models are few-shot learners](#). *Preprint*, arXiv:2005.14165.
- Dandan Chen, Amos Jeng, Shiyu Sun, and Bradley Kaptur. 2023. [Use of technology-based assessments: a systematic review covering over 30 countries](#). *Assessment in Education: Principles, Policy & Practice*, 30(5-6):396–428.

Dandan Chen Kaptur, Yue Huang, Xuejun Ryan Ji, Yan-

- hui Guo, and Bradley Kaptur. 2025. [Enhancing systematic reviews with large language models](#). *arXiv preprint*, arXiv:2504.20276.
- Susan E. Embretson and Steven P. Reise. 2000. *Item Response Theory for Psychologists*. Lawrence Erlbaum Associates.
- Carlo Galli, Anna Gavrilova, and Elena Calciolari. 2025. [Large language models in systematic review screening: Opportunities, challenges, and methodological considerations](#). *Information*, 16(5):1–20.
- Binglan Han, Anuradha Mathrani, and Teo Susnjak. 2025. [Evaluating prompting strategies and large language models in systematic literature review screening: Relevance and task-stage classification](#). *arXiv preprint*, arXiv:2510.16091.
- Bashar Hasan, Samer Saadi, Noora Rajjoub, Moustafa Hegazi, Mohammad Al-Kordi, Farah Fleti, Magdoleen Farah, Irbaz B Riaz, Imon Banerjee, Zheng Wang, and Mohammad Hassan Murad. 2024. [Integrating large language models in systematic reviews: A framework and case study using robins-i for risk of bias assessment](#). *BMJ Evidence-Based Medicine*, 29(6):394–400.
- Samuel Messick. 1989. Validity. In Robert L. Linn, editor, *Educational Measurement*, 3 edition, pages 13–103. Macmillan.
- Sewon Min, Xinxu Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. [Rethinking the role of demonstrations: What makes in-context learning work?](#) In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 11048–11064. Association for Computational Linguistics.
- Annette M. O’Connor, Justin Clark, James Thomas, René Spijker, Wojciech Kusa, Vickie R. Walker, and Melissa Bond. 2024. [Large language models, updates, and evaluation of automation tools for systematic reviews: A summary of significant discussions at the eighth meeting of the international collaboration for the automation of systematic reviews \(icasr\)](#). *Systematic Reviews*, 13(1):290.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#). *OpenAI blog*, 1(8):1–24.
- Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. 2024. [A systematic survey of prompt engineering in large language models: Techniques and applications](#). *arXiv preprint*, arXiv:2402.07927.
- Fouad Trad, Ryan Yammine, Jana Charafeddine, Marlene Chakhtoura, Maya Rahme, Ghada El-Hajj Fuleihan, and Ali Chehab. 2025. [Streamlining systematic reviews with large language models using prompt engineering and retrieval augmented generation](#). *BMC Medical Research Methodology*, 25:130.

A Coding Questions from a Systematic Review Protocol

No.	Question
Q1	Is technology-based assessment an important topic (i.e., there is enough information about it) of this article?
Q2	Is it about learning outcomes?
Q3	Is it about learner-centered assessment?
Q4	Does it address pre-college education?
Q5	Does it specify a geographic context?
Q6	Does it offer substantial content about tests measuring learning outcomes?
Q7	Does it offer substantial content about tests supporting learning?
Q8	Does it offer substantial content about tests supporting teaching?
Q9	Does it offer substantial content about tests supporting non-teaching?
Q10	Does it offer substantial content about test-associated cost?
Q11	Does it offer substantial content about test-associated time?

B Keywords and Depth Criteria for Defining “Substantial” for Each Question

No.	Keyword	Depth Criteria
Q6	learning outcomes, achievement, mastery, proficiency, competency, assessment validity, reliability, standards, objectives, alignment, cut score	measurement methods (CTT, IRT, G-theory), validity/reliability evidence, alignment procedures, scoring models, empirical results, concrete examples
Q7	formative assessment, feedback, test-enhanced learning, retrieval practice, spacing, mastery learning, adaptive testing, personalized learning, metacognition, analytics	mechanisms/interventions, instructional strategies tied to tests, empirical findings on learning gains, implementation steps, concrete examples
Q8	data-driven instruction, curriculum alignment, item analysis, standards-based grading, differentiation, RTI, lesson planning, dashboards, progress monitoring, PLCs	teacher use-cases, analysis procedures, instructional adjustments from test data, tools/platforms, concrete examples
Q9	accountability, accreditation, admissions, placement, certification, licensure, selection, screening, program evaluation, policy, resource allocation, quality assurance	decision processes, stakeholders, criteria/metrics, policies/regulations, empirical outcomes, concrete examples
Q10	cost, budget, expense, pricing, fee, per-student cost, licensing, subscription, proctoring, scoring, item development, psychometrics, hardware, TCO, ROI, cost-effectiveness	numeric estimates/ranges, cost breakdowns, comparisons, models (TCO/ROI), concrete examples
Q11	time, duration, minutes, hours, administration time, grading time, turnaround, preparation time, scheduling, time-on-task, workload, training time, proctoring time	numeric estimates/ranges, workflow steps, time comparisons, bottlenecks and mitigations, concrete examples