Research Synthesis Methods (2026), **17**: 999–1017
doi:10.1017/rsm.2026.10094

Research Synthesis Methods

RESEARCH ARTICLE

Large language model-based paper classification framework with key-insight extraction and confidence-weighted voting

Zihan Song[1], Shan Huang[1], Ngeemasara Thapa[2], Xin Zhang[3], Byung-Kwon Park[3], Jie Lu[1], Wenyang Li[1], Wenbin Liu[1], Bei Zhan[1] and Jianfei Li[4]

[1]College of Industrial Internet, Hubei Engineering Institute, China
[2]Research and Development, Center for Emergency Medicine Advancement, Nepal
[3]Department of Management Information Systems, Dong-A University, Republic of Korea
[4]Department of External Cooperation, Zhejiang Academy of Industrial Internet Development, China

Corresponding author: Byung-Kwon Park; Email: bpark@dau.ac.kr

Received: 18 August 2025; **Revised:** 6 January 2026; **Accepted:** 25 February 2026; published online 22 April 2026

Keywords: artificial intelligence; evidence synthesis; large language model; literature screening; paper classification; systematic review

Abstract

Systematic reviews (SRs) are critical for evidence-based research but are time-consuming and labor-intensive. The rapid expansion of academic publications further challenges the performance and applicability of existing screening and classification methods. While large language models (LLMs) present new opportunities for automation, limited research has examined whether they can achieve classification performance comparable to human reviewers in large-scale, multi-class settings. With the goal of improving classification performance, we proposed an LLM-based framework that leverages full-text key-insight extraction to enhance literature classification. We constructed a manually curated dataset of 900 articles from 17 published SRs to quantitatively evaluate the classification capabilities of LLMs. The results provided empirical evidence of LLMs' potential in supporting large-scale SRs and introduced a practical pathway for improving efficiency and reliability in evidence synthesis. Empirical results showed that key-insight-based classification (KBC) significantly outperforms abstract-based classification (ABC). We implemented a confidence-weighted voting (CWV) mechanism using multiple LLMs to improve robustness. The CWV method achieved the highest macro *F*1-score of 0.796, substantially exceeding KBC (0.732), ABC (0.676), and unsupervised K-means clustering (0.446). By employing zero-shot LLMs, our approach demonstrated the potential for enhanced adaptability across diverse domains and classification tasks without requiring fine-tuning, demonstrating that a carefully designed pipeline can enable LLMs to achieve classification performance comparable to human reviewers.

Highlights

What is already known

- Traditional systematic review screening is labor-intensive, and existing machine learning methods are largely restricted to abstract-level data and depend on domain-specific corpora.

What is new

- A novel LLM framework utilizing full-text "key-insight" extraction instead of simple abstract analysis.
- Performance significantly improved from a 0.676 macro *F*1-score (abstract-based method) to 0.796 (confidence-weighted voting method).

This article was awarded Open Materials badge for transparent practices. See the Data availability statement for details.

© The Author(s), 2026. Published by Cambridge University Press on behalf of The Society for Research Synthesis Methodology. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

Potential impact for *RSM* readers

- Reduces dependency on manual labeling and specialized models in evidence synthesis.
- Enables high-precision, cross-disciplinary paper classification without task-specific fine-tuning.

1. Introduction

A systematic review (SR) is an academic research methodology that systematically and objectively identifies, evaluates, and synthesizes all available research evidence pertinent to a particular topic or question.[1–3] SRs play an essential role in supporting evidence-based decision-making. However, it is time- and resource-intensive, typically requiring 6–18 months to complete from protocol initiation to submission for publication.[4] One of the most crucial steps is title and abstract screening, where researchers manually assess potentially eligible studies retrieved through structured keyword searches.[5] On average, screening 60–120 records require approximately 1 hour.[5–7] However, as the volume of papers increases, the risk of human error increases, driven by reviewer fatigue and cognitive overload.[8]

To reduce workload while maintaining decision performance, researchers have explored the use of text mining and machine learning (ML) techniques.[9,10] Notable examples include Abstrackr, which can reduce screening effort by up to 67%,[11] and Ryaan, which achieves a 20–54% reduction.[12] Despite these advances, traditional supervised ML approaches are often domain-specific and dependent on labeled datasets. While unsupervised methods like k-means reveal latent structures without labeled data, their interpretability and parameter sensitivity constrain their utility as substitutes for manual screening.[13]

Recent advances in large language models (LLMs) enable efficient, accurate evidence synthesis by performing zero-shot reasoning, semantic understanding, and extracting structured data from unstructured text—reducing the need for manual annotation and task-specific labels.[14] These capabilities provide a basis for developing end-to-end automated SR pipelines that integrate literature screening, information extraction, and classification—potentially minimizing the requirement for manual annotation.

At present, most of the research have employed LLMs for screening processes of title and abstract. For instance, Gargari et al.[15] used GPT-3.5 for title and abstract screening achieving a sensitivity of 69%, whereas another GPT-3.5 study reported a higher sensitivity (97.1%) but low specificity (37.7%).[16] Similarly, GPT-4 has also been employed as a tool for title and abstract screening, reporting varying sensitivities, 42–50%[17] and 76%.[18] Overall, these studies illustrate substantial promise but also reveal considerable variability, with specificity values in some cases below 40%. A key limitation is the reliance on title and abstract information alone, which restricts an LLM's ability to fully capture study context, methodology, and findings—factors critical for accurate inclusion decisions. Although full-text–based classification could theoretically yield superior performance, its feasibility is constrained by LLM's context length limitations and associated computational costs.

In our previous work,[19] we demonstrated the feasibility of LLMs to extract key insights—including research questions, methodologies, and findings—from full-text articles, achieving 97% agreement with human experts (relevance score). This suggests that a key-insight-driven workflow offers a viable alternative to sole reliance on titles and abstracts. Building on this foundation, the present study aims to explore the potential of LLMs in managing complex academic paper classification tasks, specifically addressing the semantic limitations of traditional automated methods. We propose a novel classification paradigm that transitions from surface-level metadata analysis to deep-text semantic synthesis. To this end, this study pursues the following objectives:

1. To develop a classification protocol centered on full-text "key-insight" extraction, utilizing LLMs to automatically distill research objectives, methodologies, and core findings. This overcomes the information bottleneck inherent in abstract-only classification and enables a more granular characterization of scientific literature.

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

2. To implement a novel confidence-weighted voting (CWV) mechanism that mitigates the stochastic instability of individual LLMs in multi-class settings. By aggregating decisions from multiple state-of-the-art models and assigning weights based on empirical validation performance, we establish a robust decision-making framework.
3. To establish a rigorous quantitative benchmark against expert-level "gold standards." Using manually curated classification schemes from published SRs, we evaluate the framework's performance in large-scale, real-world academic contexts to provide empirical evidence for the viability of LLM-based end-to-end automation.

2. Related work

In automated SR, paper classification constitutes a core component of evidence synthesis, directly influencing the performance of downstream tasks such as information extraction and trend analysis. Existing approaches to automated classification can be broadly categorized into three developmental stages: (i) rule-based and traditional ML methods, (ii) deep learning and pre-trained language models (PLMs), and (iii) emerging LLM methods. Table 1 outlines representative studies, including dataset size, features, classification labels, best-performing techniques, and results.

Early approaches predominantly relied on rule-based systems and conventional ML, using keyword matching, statistical feature extraction, and supervised learning. For example, Mustafa et al.[20] applied K-Nearest Neighbor to the ACM dataset with features from titles, abstracts, keywords, and common terms, achieving an $F1$-score of 0.88. Such methods depend heavily on metadata, which is often inconsistent or incomplete and have limited semantic understanding, making cross-domain classification difficult.

The second phase of development adopted deep learning and PLMs (e.g., BERT and its variants), enabling more effective semantic capture for classification and relevance prediction. Kandimalla et al.[21] employed a deep attention neural network to classify 45 million paper abstracts into 104 topics, achieving an $F1$-score of 0.76. Another study[22] combined SciBERT, decision trees, keyword matching, and support vector machine for a binary artificial intelligence (AI)-related classification, reaching 0.94 $F1$-score. These methods capture semantics more effectively but require substantial labeled data and have limited cross-domain generalization.

The emergence of LLMs has led to a third paradigm shift in SR automation. Contemporary approaches leverage LLMs' emergent capabilities in zero-shot learning and semantic reasoning to address longstanding challenges. Our previous work[19] demonstrated that GPT-4 achieves 97% relevance score in extracting key insights from scientific articles through structured prompting. In clinical research, similar promising results were observed in trial eligibility screening using hybrid human–AI workflows, achieving a prevalence- and bias-adjusted k of 0.96 against consensus-based human decisions.[18] Innovations in multimodal systems like Gemini-Pro,[23] which integrates text and citation graph analysis, have reduced manual screening time compared to traditional methods. However, methods employed by LLM like LLaMA3, which achieved 77.5% sensitivity and 91.4% specificity using only titles and abstracts, often remain limited to surface-level features lacking deeper semantic and methodological insights from full texts.[24] This limitation constrains both classification performance and interpretability. Frameworks such as TnT-LLM[25] operationalize these capabilities through a phased architecture. In systematic review applications, their two-stage pipeline first generates domain-specific ontologies using GPT-4's multi-step reasoning (achieving 87% taxonomy coverage) and then trains lightweight classifiers achieving 0.92 $F1$-score with 94% cost reduction versus pure-LLM classification. These advancements align with clinical SR requirements where dynamic label systems must adapt to evolving evidence hierarchies.

Despite these advancements, critical limitations remain. Most implementations concentrate on document-level processing rather than cross-document knowledge synthesis.[19] This sequential, paper-by-paper workflows overlook the higher order relations among studies.[26] Furthermore, existing evaluation frameworks tend to prioritize operational metrics (e.g., time savings) over epistemological

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

Table 1. *Overview of methods for automated paper classification in automated SR.*

Work	Sample size	Features	Classification labels	Best method	Results
Mustafa et al.[20]	86,166	Title, abstract, keywords, general terms	11 top-level ACM categories	K-Nearest neighbor	*F*1-score: 0.88
Tzika-Kostopoulou et al.[41]	2,671	Title, abstract, keywords,	8 transportation research topics	Naïve Bayes	Accuracy: 0.91
Kandimalla et al.[21]	~45,000,000	Abstract	104 subject categories	Deep attentive neural networks	*F*1-score: 0.76
Lu et al.[22]	4,000	Title, abstract, keywords, WoS categories	Binary classification	Ensemble model (SciBERT + decision tree + keyword matching + support vector machine)	*F*1-score: 0.94
Raja et al.[42]	1,000	Title, abstract	19 categories summarized by experts	Bidirectional and auto-regressive transformers	*F*1-score: 0.85
Rao et al.[43]	788	Title, abstract, keywords	A total of 44 hierarchical classes	Fine-tuned recurrent neural network	Accuracy: 0.36
López-Pineda et al.[24]	984	Title, abstract	Binary classification	Lamma 3	Sensitivity: 0.77 Specificity: 0.91
Ali et al.[44]	31,175	Title, abstract	Binary classification	Active learning	WSS@95%:0.75
Dai et al.[45]	113	Title, abstract, full text	Binary classification	GPT–4o with revised chain-of-thought prompt	AUC: 1.00
Colangelo et al.[46]	33,057	Title, abstract	Binary classification	Claude 3 Haiku	*F*1-score: 0.85–1.0

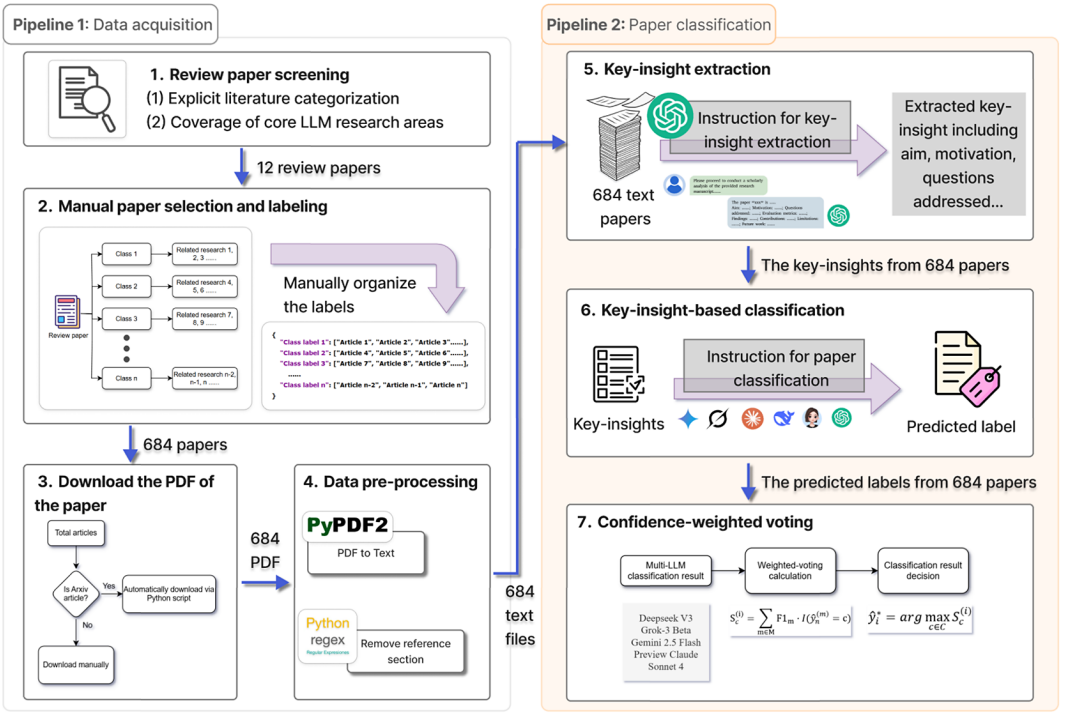


Figure 1. *System architecture.*

soundness. This gap is particularly evident in complex review contexts where meaningful synthesis arises from the aggregation of insights across a large corpus. These limitations motivated our proposed system, which integrates key-insight extraction with classification to automatically generate literature structures akin to those produced by domain experts.

3. Methods

3.1. Overview

This study addressed the prevailing challenges of inefficiency and limited automation in conducting SRs by introducing a novel framework that leverages LLMs to integrate automated paper classification with key-insight extraction. To operate this framework, we constructed a two-stage pipeline. The pipeline combines expert-annotated labels with automated LLM-based processing. As illustrated in Figure 1, the overall workflow was divided into two main pipelines: data acquisition and paper classification. The left part of the figure details the manual and automated steps for assembling a labeled dataset. The right part visualizes the process of extracting key insights and performing classification using LLMs. Specifically, let $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ denote the labeled dataset, where N denotes the total number of papers, x_i represents the textual content of the ith article, and $y_i \in C$ is its expert-assigned category label, with C being the set of all possible categories.

Our proposed method introduced a two-step mapping process. First, we defined a key-insight extraction function $g : x \mapsto k$, where k is a structured representation of the paper's academic key insights—such as research objectives, question addressed, and findings—systematically summarized from the full text. Second, based on these extracted key insights, we employed a classification function $f' : k \mapsto y$ to predict the category label y. Accordingly, the automated classification is formalized as

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

the composite mapping:

$$f(x) := f'(g(x)). \tag{1}$$

The two-stage pipeline proceeded as follows: (i) extraction of expert-provided classification labels from review articles (where the cited literature is manually categorized in a structured JSON format; see Figure 1); (ii) automation of key-insight extraction from each paper using LLMs; and (iii) utilization of extracted key-insight input by final LLM-based classification, producing predictions that aimed to maximize the agreement with human expert labels, as quantified by the following metric:

$$\text{Performance} = \frac{1}{N}\sum_{i=1}^{N} I\left(f'\left(g\left(x_i\right)\right) = y_i\right), \tag{2}$$

where $I(\cdot)$ is the indicator function.

3.2. Dataset

3.2.1. Data acquisition

Obtaining labeled datasets is challenging in automated SRs. Traditional manual annotation methods require domain experts to spend substantial time screening and classifying literature. Meanwhile, unsupervised learning may produce clustering outcomes that are misaligned with the actual needs of academic fields.[27] To address this issue, we utilized classification schemes extracted from existing literature review papers as ground-truth labels for evaluating algorithm performance.

Let $\mathcal{S} = \left\{s_j\right\}_{j=1}^{K}$ denote the set of selected survey papers, where K denotes the total number of survey papers, each providing a manually constructed classification scheme $C_| = \left\{c_k^{(j)}\right\}_{k=1}^{n_j}$. Each paper s_j further specified a set of cited publications and their assigned categories as pairs $\mathcal{R}_| = \left\{\left(r_i^{(j)}, c_{k(i)}^{(j)}\right)\right\}_{i=1}^{m_j}$, where $\mathrm{r}_i^{(\mathrm{j})}$ is a cited work in survey j, $c_{\mathrm{k(i)}}^{(j)}$ its assigned category, m_j denotes the number of primary research papers that are both cited and explicitly categorized in the jth survey paper s_j. By aggregating all surveys, we obtain a labeled dataset:

$$\mathcal{D} = \bigcup_{j=1}^{K} \mathcal{R}_| = \{(x_n, y_n)\}_{n=1}^{N}, \tag{3}$$

where x_n is the full text of the n-th cited paper, y_n its category label, and N denotes the total number of papers.

Survey paper selection criteria: We identified relevant survey papers through Google Scholar using the search queries "LLM review" and "large language model literature review," restricting the publication period to 2021–2025 and limiting the results to papers published in English. Survey papers were defined as structured surveys that provide explicit literature categorization, rather than strictly systematic reviews. The selected surveys met the following criteria: (i) Explicit literature categorization: Papers must provide structured classifications with clearly labeled citation-category mappings. (ii) Coverage of core LLM research areas: Surveys should focus on key technical aspects, applications, or challenges in LLMs. This process yielded 17 eligible survey papers and 900 cited research papers for dataset construction. Notably, we did not deliberately seek systematic reviews that flowing PRISMA standard. Among the 17 survey papers, only 2 qualify as systematic reviews with PRISMA-like rigor, reflecting the nascent stage of LLM literature.

Data collection: We extracted author-assigned category labels from the surveys, representing generalized classifications of literature within specific domains. We described all 17 review papers and their corresponding dataset information in Supplementary Table 1. For instance, Kaddour et al.[28] defined 11 distinct categories—Chatbot, Computational Biology, Computer Programming, Creative Work, Knowledge Work, Law, Medicine, Reasoning, Robotics, Social Science, and Synthetic Training

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

Data—each linked to relevant citations. Similarly, we organized these citations and their corresponding categories into our labeled dataset (Figure 1). These author-assigned categories were subsequently used as the ground truth to evaluate the performance of our multi-class classification. For each (x_n, y_n) in our dataset, the full-text PDF of x_n was collected either via (1) automated retrieval using the Python ArXiv library[29] or (2) manual retrieval for non-ArXiv sources. This dual sourcing ensured comprehensive coverage of all cited works.

3.2.2. Data preprocessing

To facilitate automated key-insight extraction and ensure data quality, multiple preprocessing steps were applied to the collected dataset. To define the dataset for analysis, let $\mathcal{X} = \left\{x_n^{\text{raw}}\right\}_{n=1}^{N}$ denote the set of text documents, where x_n^{raw} represents the full raw text extracted from the PDF $[(x_n^{\text{raw}} = f_{\text{PDF}}(x_n)]$. Full raw text content was extracted from the full-text PDFs using the PyPDF2 library.[30]

To improve insight extraction efficiency and reduce irrelevant content, cleaning function g_{clean} was applied to each document, removing reference section:

$$x_n' = g_{\text{clean}}\left(x_n^{\text{raw}}\right), 1 \le n \le N, \tag{4}$$

where x_n' is the filtered main body text.

Regular expressions were used for this cleaning step, substantially shortening the average length of article text strings from 76,941.6 to 59,144.3 characters, thereby removing redundant content and improving both the performance and efficiency of subsequent key-insight extraction.[19]

The final curated dataset was defined as $\mathcal{D}_{\text{final}} = \left\{\left(x_n', y_n\right)\right\}_{n=1}^{N}$, derived from 17 survey papers that collectively defined 107 unique categories (averaging 8.3 per survey), with each category containing on average 7.5 articles. Out of the 900 research papers collected, approximately 46.3% was sourced from ArXiv.

3.3. Key-insight extraction

Key-insight extraction was the process of identifying valuable information contained in scientific articles,[31,32] including research questions, methodologies, and stated objectives. From the SR perspective, key insights reflected focal areas of interest to researchers, such as organizing literature reviews based on research methodologies or research questions.

Mathematically, the key-insight extraction process can be formulated as a mapping $g_\theta : \mathcal{X}' \to \mathcal{K}$, where $\mathcal{X}' = \left\{x_n'\right\}_{n=1}^{N}$ is the preprocessed full-text corpus and $\mathcal{K}$ denotes the space of key-insight representations. For each article, $k_n = g_\theta\left(x_n'\right)$ is a structured sequence of key insights summarizing that article, that is,

$$k_n = \left\{k_n^{(1)}, k_n^{(2)}, \ldots, k_n^{(n)}\right\}, \tag{5}$$

where each $k_n^{(i)}$ corresponds to a specific insight, encompassing the aim, motivation, methodology, experimental results, contributions, and limitations.

Previous studies that employed manual comparison reported up to 97% similarity between the key insights extracted by GPT-4o and human experts from scientific articles.[19] Accordingly, in this study, we utilized GPT-4o to extract key insights from all the selected articles, as detailed in Supplementary Table 2.

3.4. Key-insight-based classification

Classifying research papers remains challenging due to ambiguous category boundaries and overlapping research themes. Many papers span multiple topics, such as combining model development with

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

Table 2. *LLMs used in this study.*

Model name	Abbreviation	Company	Capacity	Max context length
claude-sonnet–4	Claude	Anthropic	Text generation, reasoning	200k
deepseek-v3–250324	Deepseek	Deepseek, Inc	Text generation	128k
doubao–1.5-pro–32k–250115	Doubao	ByteDance	Text generation	128k
gemini–2.5-flash-preview–05–20	Gemini	Google	Text generation, reasoning, multi-modal input	1024k
gpt–4o	GPT	OpenAI	Text generation, multi-modal input	128k
grok–3-beta	Grok	X-AI	Text generation, reasoning, multi-modal input	1024k

application studies, making single-label assignment difficult. Additionally, variations in writing style and terminology increase the complexity of identifying key themes. Prior methods relied on titles, abstracts, or keywords, which may have overlooked critical methodological or conceptual details. As a result, classification based solely on limited textual cues can be shallow and inaccurate.

To address these limitations, we proposed a key-insight-based classification approach and hypothesized that it offers advantages over traditional methods by utilizing structured representations of each paper's core contributions. Unlike conventional abstracts that often overlook methodological details, limitations, or application contexts, key insights provided a more comprehensive and structured summary of each paper. By explicitly capturing elements such as research objectives and findings, they offered a richer semantic foundation that may enhance model interpretability and classification performance.

Formally, this step implemented a two-stage mapping. For each preprocessed article x'_n, its key-insight representation $\boldsymbol{k_n} = \boldsymbol{g_\theta}\left(\boldsymbol{x'_n}\right)$ was first extracted as outlined in Section 3.3. Classification was then performed by an LLM-based classifier, which can be defined as a function:

$$\widehat{y}_n = f'_{\boldsymbol{\Phi}}\left(k_n\right) = \arg\max_{c \in C} \Pr\left(c | k_n, \boldsymbol{\Phi}\right), \tag{6}$$

where Φ represents the LLM's parameters and prompt instructions, c is a candidate category, and C denotes the set of all possible categories. This approach used k_n, a structured key insight extracted in the previous step, as the classification basis, rather than shallow cues such as the title or abstract.

To perform this classification, we employed six state-of-the-art LLMs as shown in Table 2. Each of these LLMs possessed distinct characteristics and capabilities that could affect classification performance. Model invocations were uniformly processed by OpenRouter,[33] with the temperature parameter set to 0.7 and top-p set to 0.95 for each model to attain optimal performance through a balanced approach to predictability and creativity.[34] A lower temperature value constrains the model's randomness and creativity, whereas a higher temperature may result in excessively unstable outputs.

Specific instructions given to each model are described in Supplementary Table 3. The instruction's design was optimized for robust classification through a two-fold strategy: the "System instruction" established a consistent interpretive framework by defining the LLM's role as a meticulous literature researcher with strong thematic analysis capabilities, while the "User instruction" provided precise, contextualized input (title, key insight, category candidates, and descriptions) and mandated

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

a structured JSON output. This methodological precision aimed to mitigate LLM interpretation variability and standardize response formats for subsequent analysis.

3.5. Confidence-weighted voting

To further enhance the robustness and trustworthiness of paper classification, we introduced a confidence-weighted voting framework informed by validation set performance. Instead of treating each model's output as equally reliable, we incorporated evidence-based weighting that reflects the actual performance of each language model for our specific task.

Let $M = m_1, m_2 \ldots, m_p$ denote the set of selected LLMs and let $\mathcal{V} = \{(x_v, y_v)\}_{v=1}^{V}$ denote a small, labeled validation set representative of the classification task. Each model m in M is tasked with predicting labels for V. We then calculated the classification performance (macro $F1$-score) of each model and used this performance as the model's overall reliability score. To avoid degradation in ensemble quality, due to poorly performing models, we referenced Section 4.2's comparative experiments and excluded LLMs with significantly lower macro $F1$-scores on the validation set. Specifically, only models with macro $F1$-scores above a practical threshold were retained in the voting ensemble; this led us to select GPT, Grok, Gemini, and Claude. Deepseek and Doubao, which had notably lower macro $F1$-scores under our evaluation, were omitted from the final voting pool.

During the classification stage, each LLM was assigned a predicted category $\widehat{y}_n^{(m)}$ for each article x_i'. Rather than simple majority voting, we assigned each model's vote a weight, specifically its validation macro $F1$-score, $F1_m$. The aggregated score for assigning article x_i' to category $c \in C$ was defined as follows:

$$S_c^{(i)} = \sum_{m \in M} \text{Macro } F1 - \text{score}_m \cdot I\left(\widehat{y}_n^{(m)} = c\right), \tag{7}$$

where $\widehat{y}_n^{(m)}$ denotes the predicted category assigned to $\boldsymbol{x_n'}$ and $S_c^{(i)}$ represents the aggregated weighted score for assigning class c to the ith article.

The final category assignment for x_i' is then determined as follows:

$$\widehat{y}_i^* = \arg\max_{c \in C} S_c^{(i)}. \tag{8}$$

This strategy ensured that predictions from more accurate models exerted proportionally greater influence on the outcome while still preserving the benefits of model diversity. In case of tie between categories after weighted voting, the output of the single model with the highest $S_c^{(i)}$ was chosen as the tiebreaker.

4. Performance evaluation

4.1. Evaluation metrics

In this study, the paper classification task was framed as a multi-class classification problem. During the performance evaluation phase, our primary focus was to assess whether the proposed system produced results like those provided by the authors of the review papers. To this end, we used the accuracy, macro precision, macro recall, and macro $F1$-score as the evaluation metric. The definition of these metrics are as follows:

For each class i in a multi-class classification task, we compute its precision P_i, recall R_i, and F1 score $F1_i$:

$$P_i = \frac{\text{TP}_i}{\text{TP}_\text{i} + \text{FP}_i}, \tag{9}$$

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

Table 3. *Classification performance of different LLMs on the small-scale validation set.*

Model	Accuracy	Precision	Recall	$F1$-score
GPT	0.877	0.891	0.846	0.848
Grok	0.860	0.895	0.830	0.817
Gemini	0.842	0.838	0.816	0.800
Claude	0.842	0.880	0.799	0.799
Deepseek	0.842	0.868	0.792	0.787
Doubao	0.807	0.857	0.739	0.738

Note: The above accuracy values reflect overall accuracy. Precision, Recall, and $F1$-score are computed as macro average metrics.

$$R_i = \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN_i}}, \tag{10}$$

$$F1_i = 2 \times \frac{P_i \times R_i}{P_i + R_i}, \tag{11}$$

where TP refers to true positives, FP to false positives, and FN to false negatives. The macro precision, macro recall score, and macro $F1$-score are then defined as follows:

$$\text{Macro precision} = \frac{1}{n}\sum_{i=1}^{n} P_i, \tag{12}$$

$$\text{Macro recall} = \frac{1}{n}\sum_{i=1}^{n} R_i, \tag{13}$$

$$\text{Macro } F1 - \text{score} = \frac{1}{n}\sum_{i=1}^{n} F1_i, \tag{14}$$

where n denotes the total number of classes.

4.2. Confidence estimation on small-sample evaluation

To ensure the scientific rigor and effectiveness of the confidence-weighted voting mechanism, we conducted a detailed evaluation of several mainstream LLMs on a small-scale validation set, with a particular focus on employing the macro $F1$-score as the confidence weight. We used the key insight from 57 randomly selected papers as independent classification inputs to compare the multi-class performance of six prominent LLMs. To prevent data leakage, the test dataset used in the results, as presented in Section 4.3, does not include these 57 papers.

Evaluation metrics included accuracy, macro precision, macro recall, and macro $F1$-score. The detailed results are presented in Table 3. The assessment results revealed discernible differences in macro $F1$-scores among the various models. The GPT model achieved the best performance with a macro $F1$-score of 0.848. Grok, Gemini, and Claude also performed robustly, each yielding macro $F1$-scores close to or above 0.8. In contrast, Deepseek and Doubao exhibited lower macro $F1$-scores, registering 0.787 and 0.738, respectively.

To improve the overall robustness and quality of the weighted voting mechanism, we used each model's macro $F1$-score on the small validation set as its confidence score. The ensemble then assigned weights to each model based on these confidence scores. Due to considerably inferior macro $F1$-scores compared to the other models, Deepseek and Doubao were excluded from the final weighted voting,

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

aiming to prevent degradation of the ensemble's performance. Only the classification outputs from GPT, Grok, Gemini, and Claude were retained for the final ensemble.

4.3. Comparative analysis of classification approaches

4.3.1. Overview of comparative methods

In recent years, natural language processing (NLP) tasks have increasingly favored zero-shot approaches that do not require parameter training or fine-tuning, to enhance generalizability and convenience. Traditional methods[20–22] required model training before deployment and were less adaptable to new domains or classification schemes. Accordingly, all comparative methods in this study—including abstract-based LLM classification, key-insight-based LLM classification, and unsupervised K-means clustering—were implemented without additional training or fine-tuning, thus providing a realistic evaluation of the proposed system's applicability in current automated systematic reviews.

To evaluate the performance of the proposed system, we introduced several comparative methods as benchmarks:

- *K-means clustering (KMC)*: K-means is a widely adopted unsupervised clustering algorithm,[35] commonly used in literature clustering tasks. The performance of this method served as a baseline for the classification task and was used to evaluate the performance of other methods. The paper abstracts were encoded using Salesforce's SFR-Embedding-2_R model, followed by dimensionality reduction using principal component analysis,[36] retaining 90% of the variance. Clustering was performed separately for each review's dataset using scikit-learn library, with k set to the number of categories in that review (e.g., $k = 11$ for Review 1), and other parameters at defaults. As KMC does not yield predefined class labels, we employed the Hungarian matching algorithm[37] to align the resulting clusters with the ground truth labels based on maximum overlap. KMC provides a conventional unsupervised baseline, emphasizing the value of our zero-shot LLM approaches over purely data-driven clustering without expert labels.
- *Abstract-based classification (ABC)*: This method utilized LLMs to classify each paper based solely on its title and abstract, as presented in Supplementary Table 4.
- *Key-insight-based classification (KBC)*: In this approach, the LLM performed classification based on extracted key insights from each paper, as presented in Supplementary Table 3. Compared to the abstract-based classification, this approach enabled a more systematic examination of the potential advantages conferred by using key insights in place of original abstracts.
- *Confidence-weighted voting method (CWV)*: The method proposed in Section 3.5.

4.3.2. Results

Table 4 represents the results for the four approaches. For ABC, accuracy ranged from 0.688 (Grok) to 0.726 (Claude), with macro $F1$-scores between 0. 654 and 0. 704, respectively. The average macro $F1$-score was 0.676. This indicates that titles and abstracts alone offer limited detail for accurate classification. With KBC, accuracies increased to range of 0.734–0.768, and macro $F1$-scores improved to between 0.708 and 0.749. The richer context from key insights improved classification quality. KMC achieved 0.524 accuracy and 0.446 macro $F1$-score, indicating that unsupervised clustering here fails to capture category boundaries effectively. The CWV method reached 0.797 accuracy, 0.805 precision, 0.809 recall, and a 0.796 macro $F1$-score. Weighted ensembles of strong LLMs delivered the best overall performance.

Figure 2 summarizes the overall performance trends from different approaches. As shown in Figure 2a, for all candidate LLMs, the KBC consistently outperformed the ABC, indicating that key insights provided a more informative basis for classification than full abstracts. Figure 2b shows that, in terms of the macro $F1$-score, CWV achieved the highest score (0.796), substantially surpassing both single-model classification approaches (KBC: 0.732; ABC: 0.676) and the unsupervised K-means

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

Table 4. *Performance of paper classification methods.*

Methods	Models	Accuracy	Precision	Recall	*F*1-score
ABC	Deepseek	0.693	0.693	0.705	0.664
	Grok	0.688	0.678	0.697	0.654
	Gemini	0.702	0.722	0.716	0.683
	Claude	0.726	0.73	0.736	0.704
	GPT	0.696	0.725	0.705	0.669
	Doubao	0.708	0.729	0.715	0.679
	Average	0.702	0.713	0.712	0.676
KBC	Deepseek	0.734	0.736	0.755	0.708
	Grok	0.766	0.766	0.79	0.745
	Gemini	0.752	0.754	0.773	0.734
	Claude	0.768	0.762	0.783	0.749
	GPT	0.749	0.757	0.774	0.732
	Doubao	0.759	0.759	0.771	0.73
	Average	0.755	0.756	0.774	0.732
KMC		0.524	0.475	0.484	0.446
CWV		**0.797**	**0.805**	**0.809**	**0.796**

Note: The above accuracy values reflect overall accuracy. Precision, Recall, and *F*1-score are computed as macro average metrics. ABC: abstract-based classification; CWV: confidence-weighed voting method; KBC: key-insight-based classification; KMC: K-means clustering.

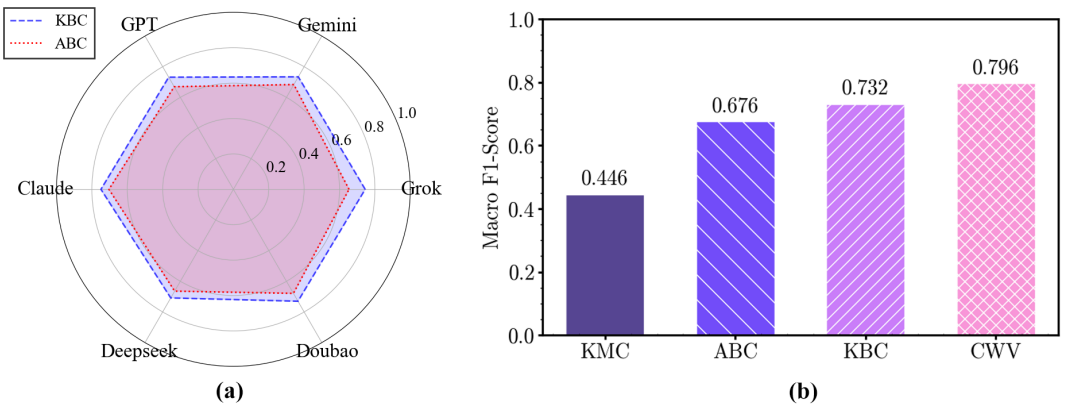


Figure 2. *(a) Macro F1-score differences of candidate LLMs: key insight versus abstract classification. (b) Comparison of the macro F1-scores across the four methods. ABC: abstract-based classification; CWV: confidence-weighed voting method; KBC: key-insight-based classification; KMC: K-means clustering.*

clustering baseline (KMC: 0.446). These results demonstrated a clear and consistent improvement when combining multiple LLMs through confidence-weighted voting, highlighting the advantage of integrating complementary model strengths.

Figure 3 illustrates the average macro *F*1-scores of the KMC, ABC, KBC, and CWV methods across datasets corresponding to 16 survey papers (where the results for ABC and KBC represent averages across all models). Overall, KMC exhibited significantly lower performance than LLM-based methods on all datasets, with substantial variability, indicating that purely unsupervised clustering struggles to stably capture the complex and fine-grained classification boundaries across different surveys.

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

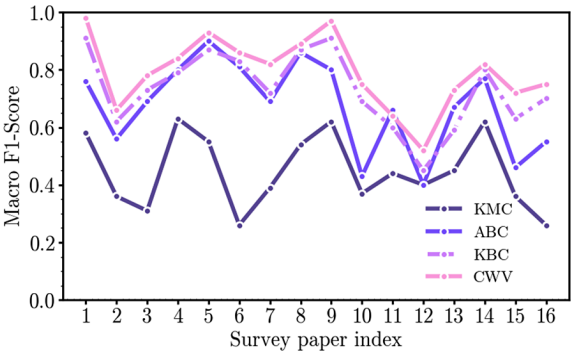


Figure 3. *Average macro F1-scores of the KMC, ABC, KBC, and CWV methods on datasets corresponding to different survey papers, where ABC and KBC results represent averages across all models. ABC: abstract-based classification; CWV: confidence-weighed voting method; KBC: key-insight-based classification; KMC: K-means clustering.*

Compared to KMC, ABC showed marked improvements; however, it still suffers from performance deficiencies on multiple datasets (e.g., surveys 7, 10, and 15), underscoring the limitations of relying solely on the representational capacity of titles and abstracts. KBC outperformed ABC on most survey datasets (except surveys 6, 11, and 13), highlighting the beneficial role of key insights in cross-survey classification tasks. The CWV method performed comparably to KBC, slightly surpassing it on all datasets, which demonstrates the advantages of multi-agent voting approaches.

4.3.3. Statistical significance of results

Given the proximity of the observed performance scores, it was essential to assess whether these differences were statistically meaningful. Due to the presence of natural boundaries in the data—potentially resulting in skewed distributions—we employed the Wilcoxon signed-rank test to evaluate differences across LLM outputs. This non-parametric test was selected because it does not assume normality, making it better suited for data with distributional irregularities. The null hypothesis posited no statistically significant differences in macro $F1$-score, while the alternative hypothesis suggested that such differences did exist. Supplementary Table 5 summarizes the test results across all evaluated methods. All pairwise comparisons yielded statistically significant differences ($p < 0.05$), indicating that each method differs meaningfully from the others. These results confirmed that subtle variations in performance were not due to random chance but a reflection of consistent methodological distinctions.

4.3.4. Error overlap analysis across classification methods

To further investigate the similarities and differences in error patterns among the classification approaches, we performed an intersection analysis of misclassified samples using an UpSet plot (Figure 4). For the LLM-based settings, Claude was selected as the representative model for the ABC and GPT for the KBC. The results revealed that the KMC method produced the largest number of unique misclassifications (417), with substantial overlap with the errors of other methods, indicating its limited ability to capture nuanced and complex category boundaries without supervision. Among the zero-shot LLM approaches, ABC yielded 243 uniquely misclassified cases, whereas KBC reduced this number to 208, reflecting the advantage of leveraging structured key insights over raw abstracts in providing richer contextual grounding for classification. In contrast, the proposed CWV approach further reduced misclassifications beyond both ABC and KBC, demonstrating its ability to integrate complementary strengths of multiple high-performing LLMs and mitigate errors that persisted in single-model settings.

Notably, the intersection analysis also revealed a core set of misclassified samples (58) common to all four methods. These papers frequently span multiple research themes or exhibit substantial content

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

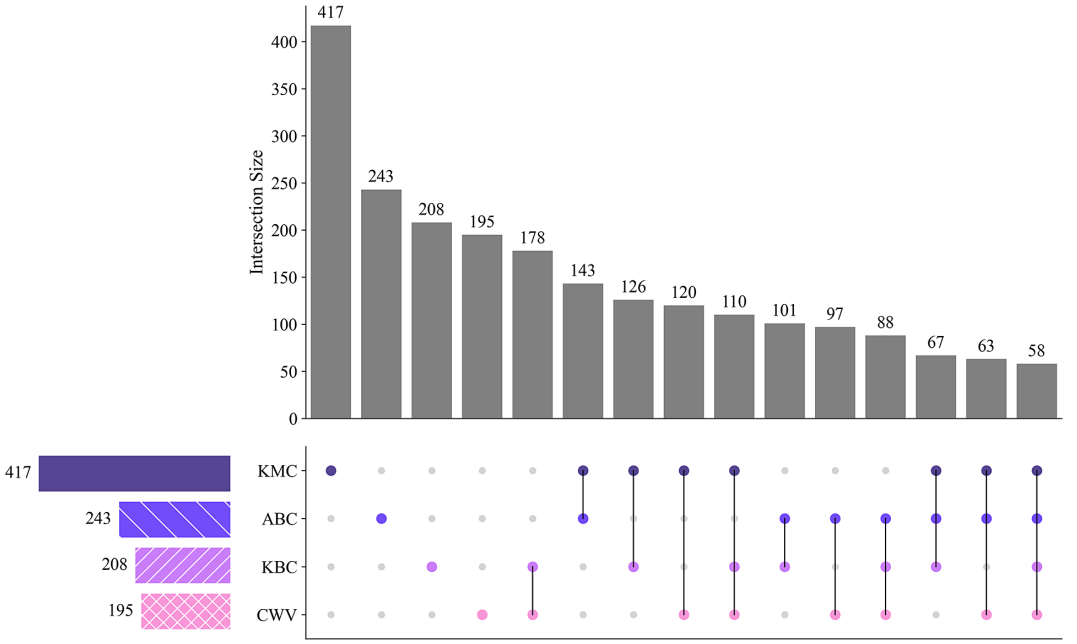


Figure 4. *UpSet plot of misclassification overlaps among classification approaches. ABC: abstract-based classification; CWV: confidence-weighed voting method; KBC: key-insight-based classification; KMC: K-means clustering.*

overlapping across categories, making them difficult to assign to a single class unambiguously. This finding highlighted an inherent limitation of single-label multi-class approaches when applied to cross-domain or thematically overlapping literature, suggesting that multi-class classification or category ambiguity modeling may be more appropriate in such cases.

5. Discussion

5.1. Overview of core findings

This study presents a novel LLM-based framework designed to automate the complex task of paper classification within SRs. Our approach primarily leveraged the generalization capabilities of LLMs. Compared with traditional language models, LLMs achieved significant advancements in both the performance and span of contextual understanding.[38] These improvements made it feasible to perform deeper and more meaningful analyses of article content. Unlike prior research such as ChatPDF,[39] which applied LLMs for content-based querying, or other emerging tools like Abstrackr[11] that focused on predicting researchers' interest in academic papers, our study leverages LLMs to generate reviewer-like outputs, facilitating expert-aligned evaluations in rigorous SRs.

Our empirical evaluation across a manually curated benchmark of 900 articles from 17 survey papers yielded several critical insights:

First, KBC outperforms ABC. As shown in Table 4, moving from metadata-level analysis to full-text synthesis improved the macro $F1$-score by 8.2% on average. In specialized domains like LLM research, abstracts frequently lack the granular methodological details required for precise categorization. By extracting and structuring core findings into key insights, the classifiers gain access to the high-density evidence needed to resolve category ambiguity.

Second, the CWV mechanism effectively mitigates the inherent instability of individual zero-shot LLMs. While single-model performance varied, the aggregated CWV approach achieved a macro

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

*F*1-score of 0.796, approaching human-expert levels. The effectiveness of a static weighting scheme across 17 heterogeneous datasets suggests that the different reasoning patterns of various LLMs are complementary. This ensemble-based approach provides a reliable validation mechanism for SR workflows, where classification accuracy is critical for downstream synthesis.[40]

Finally, the comparison with the KMC baseline highlights a shift in automation paradigms. Despite using state-of-the-art semantic embeddings, unsupervised clustering correlates poorly with expert-defined taxonomies (macro *F*1: 0.446). This disparity underscores that scientific classification requires alignment with specific, objective-driven ontologies rather than just identifying statistical patterns. Our zero-shot framework addresses this by using pre-defined category definitions as semantic references, enabling it to adapt to diverse review goals without the need for manual labeling or domain-specific fine-tuning.

5.2. The superiority of key-insight extraction

The central finding of this study is the measurable performance advantage of KBC over ABC. This superiority is consistently observed across 13 out of 16 survey datasets (Figure 3), suggesting that the benefit of using distilled full-text data is systemic rather than dataset-specific.

The performance gap between ABC and KBC (averaging an 8.2% improvement in macro *F*1-score) can be analyzed through the lens of information sufficiency. While abstracts are designed to provide a high-level summary, they are often subject to length constraints imposed by journals (typically 200–400 words), leading to the omission of specific methodological parameters, secondary findings, or tool-specific implementation details. In our dataset, the average input string length for ABC was 3879.5 ± 1260.3 characters, whereas KBC provided 5867.8 ± 1284.8 characters. However, this gain is not merely a product of increased token count. Abstracts are typically highly compressed due to strict word limits, which often forces the omission of critical technical implementations, such as specific algorithmic variants or detailed benchmark results. The KBC mechanism overcomes this inherent information scarcity by retrieving and reconstructing research objectives, technical methodologies, specific metrics, and core findings from the full-text body. This process does not simply reduce text; rather, it introduces high-value details that are absent from metadata and reorganizes them into a structured format, significantly increasing the density of discriminative information.

This structural refinement is particularly effective when navigating categories with high semantic overlap. In Survey-1 (Supplementary Table 1), which focuses on Parameter-Efficient Fine-Tuning (PEFT), numerous papers are described at the abstract level using generic terms like "optimizing large-scale models," making it difficult for ABC to distinguish between specific technical pathways. We provide a concrete example in Supplementary Table 6, comparing the abstract and key insights of the paper "Raise a Child in Large Language Model: Towards Effective and Generalizable Fine-tuning." In this instance, while the abstract refers broadly to "strategically masking" a subset of parameters, the key insights successfully retrieve crucial details from the full text, such as the two distinct variants: the task-free Child-TuningF (utilizing a Bernoulli distribution) and the task-driven Child-TuningD (utilizing Fisher Information). Crucially, the key insights also capture the method's divergence from model pruning—noting that it retains the full network during inference—and its orthogonality to existing techniques.

These specific features retrieved from the full-text body provide the LLM with unambiguous evidence for categorization. By transforming implicit features into explicit structural insights, the KBC framework significantly enhances the model's ability to differentiate between fine-grained technological categories that appear indistinguishable in a metadata-only analysis. Furthermore, the structured evidence layer provided by KBC enhances the transparency of the automated classification process. In an SR workflow, researchers can directly audit the logical consistency between the extracted "key insights" and the final categorical assignment. By transforming implicit full-text features into explicit, structured insights, the KBC framework not only compensates for the poverty of information in abstracts but also provides an auditable chain of evidence, making automated decision-making more robust and aligned with the rigorous requirements of scientific evidence synthesis.

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

5.3. Collective decision logic and deployment in systematic reviews

This study integrated predictions from multiple LLMs through CWV, significantly enhancing classification stability. Although a static weighting strategy based on validation set $F1$ scores was employed, the performance gains remain substantial. This improvement stems primarily from the complementary bias of different models when processing complex semantics. Due to divergences in pre-training corpora and reasoning logic, individual models often exhibit specific inferential biases when handling interdisciplinary or semantically overlapping literature categories. The CWV mechanism effectively leverages the consensus of high-performance models, such as GPT-4o and Claude 3.5, mitigating classification errors caused by stochasticity or context-window limitations of a single model through weighted allocation. In practical deployment, the application of this ensemble mode must balance precision against computational cost; thus, we propose the CWV framework as an extensible solution that can be dynamically regulated based on task stages and user requirements.

For individual researchers or small teams, the primary value of CWV lies in providing a low-cost, automated "cross-check" mechanism, aimed at mitigating systematic misjudgments by a single model that are often difficult to detect due to limited human resources. In such lightweight scenarios, users are not required to maintain multi-model operations throughout the entire workflow; instead, a phased strategy can be implemented. During the initial screening stage of a SR, when faced with thousands of retrieved records, a low-cost ABC paired with a single model can be utilized to maximize recall while controlling inference overhead. Only upon transition to the full-text analysis stage—where the sample size has been pruned to hundreds and requires high-precision adjudication—is the multi-model decision-making process, based on key-insight extraction, activated. This on-demand invocation logic ensures that individual researchers can achieve expert-level reliability and consistency with controllable API expenditures.

For large-scale academic institutions or commercial developers, the engineering significance of this framework is manifested in its optimization of classification stability and throughput for complex literature streams. In commercial deployments, the "semantic compression layer" facilitated by key-insight extraction can structure raw PDFs—which often contain tens of thousands of characters—into compact representations of approximately 5,000 characters. This directly reduces the total token consumption of concurrent multi-model adjudications and bypasses the computational bottlenecks associated with long-context processing. Furthermore, developers of large-scale systems can monitor the consensus level of voting distributions to identify disciplinary fields with low classification consistency, thereby routing controversial samples arising from LLM uncertainty to human experts for manual review. This modular design, tailored for late-stage precision classification and label alignment, allows researchers to switch flexibly between KBC and CWV methods based on the specific scale and budget of the SR. This ensures that the framework maintains both scientific rigor and engineering feasibility when processing massive volumes of academic literature.

5.4. Limitations and directions for future research

Despite demonstrating the efficacy of key-insight extraction and weighted voting mechanisms, this study is subject to several limitations. First, regarding dataset construction, although we expanded the expert-annotated set to include 900 papers, approximately 46.3% of the data originated from the ArXiv platform. While this distribution reflects the rapid iteration of research in the LLM field, ArXiv abstracts often exhibit a concise style characteristic of specific disciplines (e.g., computer science and mathematics). This may differ from the more structured and formal abstracts typical of biomedical or social science journals. This limitation in data sources suggests that future research should apply the framework to disciplines with traditions of longer, more complex abstracts to evaluate its generalizability across diverse writing norms. Second, this study is predicated on the classification criteria of previously published survey papers; consequently, the upper bound of our findings is constrained by the inter-annotator agreement of the original authors. Human multi-class classification inherently involves

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

significant subjective variance. The stabilization of macro $F1$ scores for literature classification between 0.7 and 0.8 largely reflects the blurred boundaries caused by interdisciplinary overlaps—a challenge that cannot be eliminated through technical optimizations alone. On the other hand, although the proposed system showed strong alignment with human-assigned categories, it currently assumes a static and mutually exclusive set of classification labels. Future research could explore hierarchical or multi-label classification frameworks to accommodate the complex and overlapping nature of scholarly work.

In light of these challenges, future research should shift from static evaluation toward more interactive and dynamic systems. The current framework assumes a predefined and static classification labeling system; however, in actual SR workflows, taxonomies often evolve as evidence accumulates. Future work could explore the introduction of an ontology-driven closed-loop mechanism, allowing LLMs to automatically propose new categories or restructure classification hierarchies when a significant portion of the literature fails to align with existing labels. Furthermore, to address resource-balancing pressures in large-scale applications, the development of adaptive model-selection logic holds substantial engineering value. Specifically, the system could dynamically determine whether to invoke high-cost, high-tier models or multi-model voting based on initial confidence scores. This would enable an efficiency optimization whereby "low-uncertainty samples follow a single-model pipeline, while highly controversial samples trigger collective decision logic." Finally, establishing a human-in-the-loop evaluation standard that incorporates interaction between human experts and LLMs will be critical. Researching how LLM-generated key insights can assist researchers in more rapidly correcting classification biases will provide a key pathway for transitioning automated systematic reviews from mere tool-assisted tasks to a paradigm of human–AI collaboration.

6. Conclusion

This study presented a novel framework for automated paper classification in SRs using LLMs to extract key insights from full-text articles. The proposed CWV method approach significantly outperformed traditional methods, achieving a macro $F1$-score of 0.796 compared to 0.676 for ABS. CWV mechanism successfully integrated multiple LLMs, reducing classification errors and enhancing robustness. Statistical validation confirmed these performance improvements, demonstrating that key insights provide superior classification foundations than surface-level features. Although computational costs and annotation subjectivity still remain as challenges, this research provides a promising foundation for more efficient automated systematic review systems. Future work should explore hierarchical classification frameworks to enhance scalability across diverse academic domains.

Author contributions. Conceptualization: Zihan Song; Formal analysis: Shan Huang, Jie Lu; Investigation: Ngeemasara Thapa, Wenyang Li; Validation: Jie Lu, Wenyang Li, Wenbin Liu; Visualization: Xin Zhang, Byung-Kwon Park; Writing—original draft: Zihan Song; Writing—review and editing: Ngeemasara Thapa, Xin Zhang, Byung-Kwon Park, Bei Zhan, Jianfei Li.

Competing interest statement. The authors declare that no competing interests exist.

Data availability statement. The data that support the findings of this study are openly available at https://doi.org/10.5281/zenodo.19221294.

Ethical approval. There is no ethical situation related to this study.

Funding statement. This work was supported by the Dong-A University research fund.

Supplementary material. To view supplementary material for this article, please visit http://doi.org/10.1017/rsm.2026.10094.

References

[1] Avenali A, Daraio C, Di Leo S, Matteucci G, Nepomuceno T. Systematic reviews as a metaknowledge tool: caveats and a review of available options. *Int Trans Oper Res*. 2023;30(6): 2761–2806. https://doi.org/10.1111/itor.13309.

[2] Beller EM, Chen JKH, Wang ULH, Glasziou PP. Are systematic reviews up-to-date at the time of publication? *Syst Rev*. 2013;2(1): 36. https://doi.org/10.1186/2046-4053-2-36.

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

[3] Bin RI, Naqvi SAA, Hasan B, Murad MH. Future of evidence synthesis: automated, living, and interactive systematic reviews and meta-analyses. *Mayo Clinic Proc*. 2024;2(3): 361–365. https://doi.org/10.1016/j.mcpdig.2024.05.023.
[4] Yao X, Kumar MV, Su E, Flores Miranda A, Saha A, Sussman J. Evaluating the efficacy of artificial intelligence tools for the automation of systematic reviews in cancer research: a systematic review. *Cancer Epidemiol*. 2024;88(102511). https://doi.org/10.1016/j.canep.2023.102511.
[5] Spijker R, Dinnes J, Glanville J, Eisinga A. Searching for and selecting studies. In: *Cochrane Handbook for Systematic Reviews of Diagnostic Test Accuracy*. Wiley; 2023: 97–129. https://doi.org/10.1002/9781119756194.ch6.
[6] Page MJ, Moher D, Bossuyt PM, et al. PRISMA 2020 explanation and elaboration: updated guidance and exemplars for reporting systematic reviews. *BMJ*. Published online March 29, 2021: n160. https://doi.org/10.1136/bmj.n160.
[7] Higgins JPT, Thomas J, Chandler J, et al. , eds. *Cochrane Handbook for Systematic Reviews of Interventions*. 2nd ed. John Wiley & Sons; 2019.
[8] Bannach-Brown A, Przybyła P, Thomas J, et al. Machine learning algorithms for systematic review: reducing workload in a preclinical review of animal studies and reducing human screening error. *Syst Rev*. 2019;8(1): 23. https://doi.org/10.1186/s13643-019-0942-7.
[9] Matsui K, Utsumi T, Aoki Y, Maruki T, Takeshima M, Takaesu Y. Human-comparable sensitivity of large language models in identifying eligible studies through title and abstract screening: 3-layer strategy using GPT-3.5 and GPT-4 for systematic reviews. *J Med Internet Res*. 2024;26: e52758. https://doi.org/10.2196/52758.
[10] Dennstädt F, Zink J, Putora PM, Hastings J, Cihoric N. Title and abstract screening for literature reviews using large language models: an exploratory study in the biomedical domain. *Syst Rev*. 2024;13(1): 158. https://doi.org/10.1186/s13643-024-02575-4.
[11] Carey N, Harte M, Mc Cullagh L. A text-mining tool generated title-abstract screening workload savings: performance evaluation versus single-human screening. *J Clin Epidemiol*. 2022;149: 53–59. https://doi.org/10.1016/j.jclinepi.2022.05.017.
[12] Scherhag J, Burgard T. Performance of semi-automated screening using Rayyan and ASReview: a retrospective analysis of potential work reduction and different stopping rules. *ZPID (Leibniz Institute for Psychology)*. Preprint posted online May 2023. https://doi.org/10.23668/psycharchives.12843.
[13] Lund B, Ma J. A review of cluster analysis techniques and their uses in library and information science research: *k-means* and *k-medoids* clustering. *Perform Meas Metrics*. 2021;22(3): 161–173. https://doi.org/10.1108/PMM-05-2021-0026.
[14] Liu S, Li Y, Li J, Yang S, Lan Y. Unleashing the power of large language models in zero-shot relation extraction via self-prompting. Paper presented at: *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics; 2024: 13147–13161. https://doi.org/10.18653/v1/2024.findings-emnlp.769.
[15] Kohandel Gargari O, Mahmoudi MH, Hajisafarali M, Samiee R. Enhancing title and abstract screening for systematic reviews with GPT-3.5 turbo. *BMJ Evid Based Med*. 2024;29(1): 69–70. https://doi.org/10.1136/bmjebm-2023-112678.
[16] Tran VT, Gartlehner G, Yaacoub S, et al. Sensitivity and specificity of using GPT-3.5 turbo models for title and abstract screening in systematic reviews and meta-analyses. *Ann Intern Med*. 2024;177(6): 791–799. https://doi.org/10.7326/M23-3389.
[17] Khraisha Q, Put S, Kappenberg J, Warraitch A, Hadfield K. Can large language models replace humans in systematic reviews? Evaluating GPT-4's efficacy in screening and extracting data from peer-reviewed and grey literature in multiple languages. *Res Synth Methods*. 2024;15(4): 616–626. https://doi.org/10.1002/jrsm.1715.
[18] Guo E, Gupta M, Deng J, Park YJ, Paget M, Naugler C. Automated paper screening for clinical reviews using large language models: data analysis study. *J Med Internet Res*. 2024;26(e48996). https://doi.org/10.2196/48996.
[19] Song Z, Hwang GY, Zhang X, Huang S, Park BK. A scientific-article key-insight extraction system based on multi-actor of fine-tuned open-source large language models. *Sci Rep*. 2025;15(1): 1608. https://doi.org/10.1038/s41598-025-85715-7.
[20] Mustafa G, Usman M, Afzal MT, Shahid A, Koubaa A. A comprehensive evaluation of metadata-based features to classify research paper's topics. *IEEE Access*. 2021;9: 133500–133509. https://doi.org/10.1109/ACCESS.2021.3115148.
[21] Kandimalla B, Rohatgi S, Wu J, Giles CL. Large scale subject category classification of scholarly papers with deep attentive neural networks. *Front Res Metr Anal*. 2021; 5. https://doi.org/10.3389/frma.2020.600382.
[22] Lu M, Tang L, Zhou X. An ensemble approach for research article classification: a case study in artificial intelligence. *PeerJ Comput Sci*. 2024;10: e2521. https://doi.org/10.7717/peerj-cs.2521.
[23] Ye A, Maiti A, Schmidt M, Pedersen SJ. A hybrid semi-automated workflow for systematic and literature review processes with large language model analysis. *Future Internet*. 2024;16(5): 167. https://doi.org/10.3390/fi16050167.
[24] López-Pineda A, Nouni-García R, Carbonell-Soliva Á, Gil-Guillén VF, Carratalá-Munuera C, Borrás F. Validation of large language models (Llama 3 and ChatGPT-4o mini) for title and abstract screening in biomedical systematic reviews. *Res Synth Methods*. 2025;16(4): 620–630. https://doi.org/10.1017/rsm.2025.15.
[25] Wan M, Safavi T, Jauhar SK, et al. TnT-LLM: text mining at scale with large language models. Paper presented at: *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM; 2024: 5836–5847. https://doi.org/10.1145/3637528.3671647.
[26] Zou A, Yu W, Zhang H, et al. DocBench: a benchmark for evaluating LLM-based document reading systems. Paper presented at: *Proceedings of the 4th International Workshop on Knowledge-Augmented Methods for Natural Language Processing*, 359–373. Association for Computational Linguistics. https://doi.org/10.18653/v1/2025.knowledgenlp-1.29.

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press

[27] Griffiths TL, Steyvers M. Finding scientific topics. *Proc Natl Acad Sci*. 2004;101: 5228–5235. https://doi.org/10.1073/pnas.0307752101.
[28] Kaddour J, Harris J, Mozes M, Bradley H, Raileanu R, McHardy R. Challenges and applications of large language models. Published online July 19, 2023. Accessed January 5, 2026. arXiv:2307.10169.
[29] Schwab L. Python arxiv. Accessed January 5, 2026. https://pypi.org/project/arxiv/.
[30] Martin T. pubpub-zz, j-t-1. PyPDF2. Accessed January 5, 2026. https://github.com/talumbau/PyPDF2.
[31] Kovačević A, Konjović Z, Milosavljević B, Nenadic G. Mining methodologies from NLP publications: a case study in automatic terminology recognition. *Comput Speech Lang*. 2012;26(2): 105–126. https://doi.org/10.1016/j.csl.2011.09.001.
[32] Tateisi Y, Ohta T, Miyao Y, Pyysalo S, Aizawa A. Typed entity and relation annotation on computer science papers. Paper presented at: *Proceedings of the 10th International Conference on Language Resources and Evaluation, LREC 2016*. Published online 2016: 3836–3843.
[33] OpenRouter. Accessed August 6, 2025. https://openrouter.ai/.
[34] Nhat Minh N, Baker A, Neo C, Roush A, Kirsch A, Shwartz-Ziv R. Turning up the heat: min-p sampling for creative and coherent LLM outputs. Published online November 20, 2025. Accessed January 5, 2026. arXiv:2407.01082.
[35] Bide P, Shedge R. Improved document clustering using k-means algorithm. Paper presented at: *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*. IEEE; 2015: 1–5. https://doi.org/10.1109/ICECCT.2015.7226065.
[36] Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in python. *J Mach Learn Res*. 2011;12: 2825–2830. http://jmlr.org/papers/v12/pedregosa11a.html.
[37] Kuhn HW. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*. 1955;2(1–2): 83–97. https://doi.org/10.1002/nav.3800020109.
[38] Bubeck S, Chandrasekaran V, Eldan R, et al. Sparks of artificial general intelligence: early experiments with GPT-4. Published online March 22, 2023. https://www.microsoft.com/en-us/research/publication/sparks-of-artificial-general-intelligence-early-experiments-with-gpt-4/.
[39] Panda S. Enhancing PDF interaction for a more engaging user experience in library: introducing ChatPDF. *IP Indi J Libr Sci Informa Technol*. 2023;8(1): 20–25. https://doi.org/10.18231/j.ijlsit.2023.004.
[40] Boell SK, Cecez-Kecmanovic D. A hermeneutic approach for conducting literature reviews and literature searches. *Communications of the Association for Information Systems*. 2014; 34. https://doi.org/10.17705/1CAIS.03412.
[41] Tzika-Kostopoulou D, Nathanail E, Kokkinos K. Big data in transportation: a systematic literature analysis and topic classification. *Knowl Inf Syst*. 2024;66(8): 5021–5046. https://doi.org/10.1007/s10115-024-02112-8.
[42] Raja H, Munawar A, Mylonas N, et al. Automated category and trend analysis of scientific articles on ophthalmology using large language models: development and usability study. *JMIR Form Res*. 2024;8: e52462. https://doi.org/10.2196/52462.
[43] Rao SX, Tu Y, Egger PH. SAINE: scientific annotation and inference engine of scientific research. Paper presented at: *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics: System Demonstrations*. Association for Computational Linguistics; 2023: 41–58. https://doi.org/10.18653/v1/2023.ijcnlp-demo.6.
[44] Ali F, Tan ASC, Wang SJW. Can machine learning help accelerate article screening for systematic reviews? Yes, when article separability in embedding space is high. *Res Synth Methods*. 2025;16(1): 194–210. https://doi.org/10.1017/rsm.2024.16.
[45] Dai ZY, Wang FQ, Shen C, et al. Accuracy of large language models for literature screening in thoracic surgery: diagnostic study. *J Med Internet Res*. 2025;27: e67488. https://doi.org/10.2196/67488.
[46] Colangelo MT, Guizzardi S, Meleti M, Calciolari E, Galli C. Performance comparison of large language models for efficient literature screening. *BioMedInformatics*. 2025;5(2): 25. https://doi.org/10.3390/biomedinformatics5020025.

Cite this article: Song Z, Huang S, Thapa N, Zhang X, Park B.-K., Lu J, Li W, Liu W, Zhan B, Li J. Large language model-based paper classification framework with key-insight extraction and confidence-weighted voting. *Research Synthesis Methods*. 2026;17: 999–1017. https://doi.org/10.1017/rsm.2026.10094

https://doi.org/10.1017/rsm.2026.10094 Published online by Cambridge University Press