AgentSLR: Automating Systematic Literature Reviews in Epidemiology with Agentic AI

Anonymous Authors[1]

Abstract

Systematic literature reviews are essential for synthesizing scientific evidence but are costly, difficult to scale and time-intensive, creating bottlenecks for evidence-based policy. We study whether large language models can automate the complete systematic review workflow, from article retrieval, article screening, data extraction to report synthesis. Applied to epidemiological reviews of nine WHO-designated priority pathogens and validated against expert-curated ground truth, our open-source agentic pipeline (AgentSLR) achieves performance comparable to human researchers while reducing review time from approximately 7 weeks to 20 hours (a 58× speed-up). Our comparison of five frontier models reveals that performance on SLR is driven less by model size or inference cost than by each model's distinctive capabilities. Through human-in-the-loop validation, we identify key failure modes. Our results demonstrate that agentic AI can substantially accelerate scientific evidence synthesis in specialised domains.

1. Introduction

Modern AI systems, powered by large language models (LLMs), demonstrate the ability to answer expert-level scientific questions (Rein et al., 2024), support extended reasoning tasks (Kwa et al., 2025), interpret scientific figures (Roberts et al., 2024) and generate code for research problems (Tian et al., 2024). These advances suggest that LLMs might be able to automate complex, multi-stage scientific workflows that currently require substantial human expert effort (Wang et al., 2023; Zhang et al., 2025; Lu et al., 2024). Systematic literature reviews (SLRs)—comprehensive syntheses that require the retrieval, screening, extraction and analysis of up to thousands of scientific articles—represent a challenging test case (Zahavi & Einav, 2025). The benefits of such automation are significant, as traditional SLR workflows are resource-intensive, taking on average 67 weeks (Borah et al., 2017; Michelson & Reuter, 2019) and $141,000 in labour to complete (Michelson & Reuter, 2019).

Empirical studies of LLM-assisted scientific evidence-based workflows, akin to SLRs, suggest that LLMs can reduce the workload of title and abstract screening (Oami et al., 2024). Beyond screening, LLM failures relevant to scientific use have been documented previously: when summarising research articles, models often generalise conclusions, risking summaries that omit scope-limiting details (Peters & Chin-Yee, 2025). These risks can compound in multi-stage agentic pipelines, where large-scale evaluations of multi-agent LLM systems report frequent failures (Pan et al., 2025).

To ground these issues in a real scientific workflow, we focus on infectious disease epidemiology as an application domain. SLRs in epidemiology often require standardised extraction of a wide range of key parameters such as the basic reproduction number, serial interval, and case-fatality ratio across diverse study designs and data representations. We therefore select epidemiology to establish the technical feasibility of automating evidence synthesis in a specialised scientific domain.

Our contributions are summarised as follows:

- We introduce **AgentSLR**, a fully *open-source pipeline* that uses language reasoning models (LRMs) to automate real-world systematic reviews in epidemiology.
- We apply AgentSLR to a high-stakes domain, evaluating against expert labels from SLRs on WHO-designated priority pathogens. We demonstrate that agentic LRM systems can achieve comparable outputs to humans while processing articles 58 times faster.
- We conduct model ablations across five frontier reasoning models (`gpt-oss-120b`, `GPT-5.2`, `Kimi-K2.5`, `GLM-4.7`, `DeepSeek-V3.2`), analysing the performance-to-cost trade-off across pipeline stages and identifying where model choice most impacts systematic review automation.

2. AgentSLR Pipeline

We present AgentSLR, an open-source pipeline designed to streamline systematic literature reviews. It is comprised of six stages (Figure 1): article retrieval, abstract and full-text screening, OCR-based PDF to markdown conversion, structured data extraction using tools and report generation.

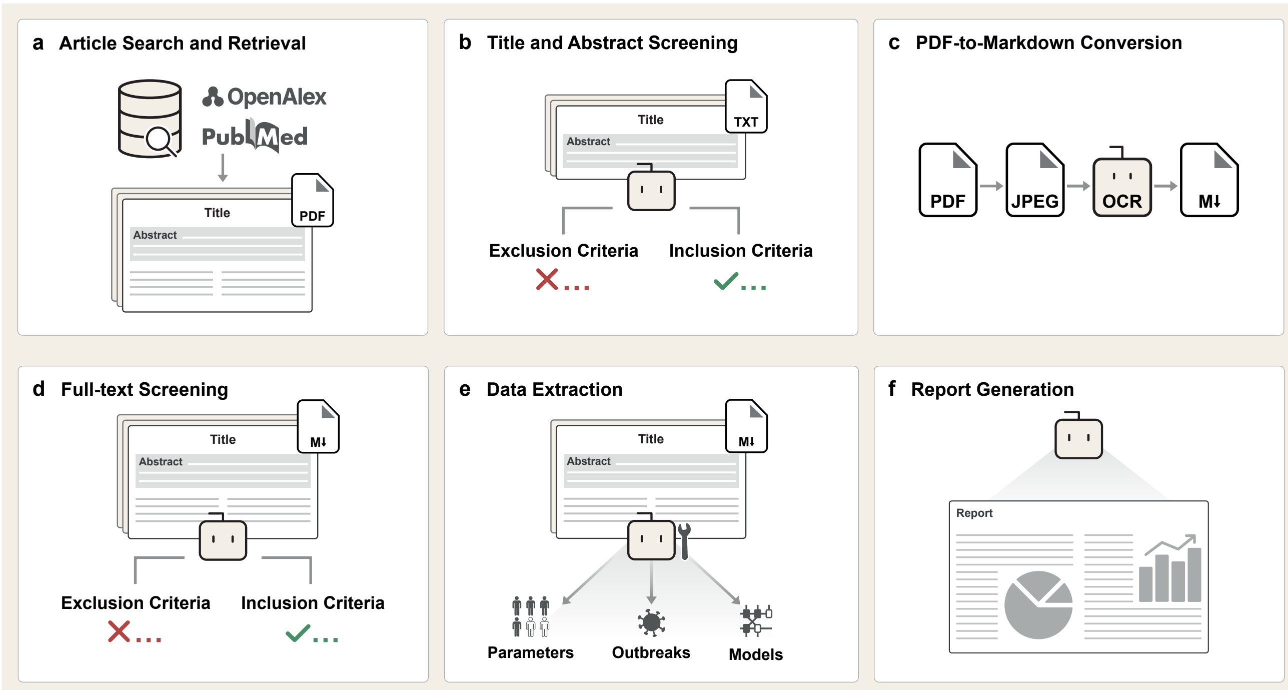


Figure 1. **AgentSLRfor assiting systematic literature reviews in epidemiology** (a) *Article Search and Retrieval* queries bibliographic databases with domain-specific Boolean searches and obtains PDF from open-access sources. (b) *Title and Abstract Screening* applies language reasoning models to filter articles using expert-designed inclusion/exclusion criteria. (c) *PDF-to-Markdown Conversion* uses an image-to-text OCR model to convert PDFs to machine-readable Markdown. (d) *Full-text Screening* applies stricter filtering criteria than (b). (e) *Data Extraction* employs multi-stage tool-calling with schema validation to extract structured epidemiological data (parameters, models, outbreaks). (f) *Report Generation* synthesises extracted data through programmatic descriptive generation followed by iterative LRM self-refinement (writing, critique and evidence grounding). For more details see Section 2.

Article Search and Retrieval AgentSLR queries three bibliographic databases (OpenAlex, PubMed, and Europe PMC) using domain-specific Boolean search strategies covering seven core epidemiological domains. Retrieved records are first deduplicated using identifier and bibliographic metadata-based matching, then full texts are automatically retrieved from open-access sources. The download pipeline incorporates caching, streaming and file validation, parallel execution and checkpointing. Full details are provided in Appendix C.

Title and Abstract Screening Initial screening is conducted using titles and abstracts based on predefined inclusion and exclusion criteria (Appendix D). We use large language reasoning models (LRMs), which enable inference-time scaling without fine-tuning on limited prior studies. Following the ScreenPrompt methodology (Cao et al., 2025b), we structure screening with five components: study objectives, inclusion/exclusion criteria, chain-of-thought instructions, article abstract and structured output format.

PDF-to-Markdown Conversion Each downloaded PDF is rendered page-by-page into high-resolution images, then processed with an OCR model to recover text while preserving document hierarchy, equations (LaTeX), and tables (HTML). The process produces 1 Markdown file per article.

Full-text Screening Converted articles undergo full-text screening using an LRM with a prompt structure analogous to abstract screening but with stricter criteria, requiring extractable quantitative epidemiological parameters (e.g. transmission rates, incubation periods and severity outcomes) while excluding literature reviews, meta-analyses, and case studies describing fewer than 10 infected individuals. We provide the full criteria and prompts in Appendix D.

Data Extraction We extract structured data for three categories (epidemiological parameters, transmission models and concluded outbreaks) using a multi-stage, schema-constrained framework. Extraction assumes the use of agentic models and provides tools that enforce field-level constraints and ensure structured outputs, mimicking human annotators who extract relevant data and fill out survey forms. For each data category, the pipeline first conducts *presence flagging* to identify articles containing relevant data, followed by targeted extraction using parameter-specific, model-specific, or outbreak-specific tool calls for validated outputs. For epidemiological parameters, extraction also

Table 1. **Total research articles processed for creating priority pathogen SLRs. PERG** indicates the total article count retrieved from `epireview` (R package) after deduplication; **AgentSLR** indicates articles downloaded after AgentSLR's Article Search and Retrieval stage; **Matched** indicates their intersection. The coloured symbols represent the progress of PERG's SLRs per priority-pathogen: published ●, conducting data extraction ●, and yet to begin screening ●, as of March 2026.

Pathogen	PERG*	AgentSLR	Matched
● Marburg virus	2,593	6,501	762 (29.4%)
● Ebola virus	11,605	23,226	3,938 (33.9%)
● Lassa fever	2,131	6,514	647 (30.4%)
● SARS-CoV-1	12,280	7,540	1,967 (16.0%)
● Zika virus	10,510	3,103	2,128 (20.2%)
● MERS-CoV	19,656	23,204	5,675 (28.9%)
● Nipah virus	1,458	5,103	664 (45.5%)
● Rift Valley fever virus	–	6,810	–
● CCHF virus	–	3,478	–
Total†	60,233	75,191	15,781 (26.2%)

*Articles post deduplication and empty abstract removal.
†Excludes Rift Valley fever virus and CCHF virus article counts.

involves population tagging (e.g. age groups, geographic locations and clinical severity), which enables subsequent aggregation of parameter estimates into summary statistics across population contexts. Complete schemas, tool definitions and validation rules are provided in Appendix E.

Report Generation Extracted data are converted into a report through a multi-stage process. Descriptive statistics are computed and visualised alongside standardised figures and evidence tables with an accompanying content manifest. An LRM generates an initial narrative synthesis, which then undergoes iterative self-refinement. Each iteration consists of a rubric-based critique assessing clarity, completeness and traceability, followed by targeted revision.

3. Methods

3.1. Data

To evaluate against ground truth, we used SLRs from the Pathogen Epidemiology Review Group (PERG) and corresponding data made available through the `epireview` and `priority-pathogen` R packages (Naidoo et al., 2025; Nash et al., 2026). PERG is conducting SLRs for nine "priority pathogens" identified by the WHO as having high epidemic or pandemic potential (World Health Organization, 2024). The group has published five peer-reviewed SLRs (Cuomo-Dannenburg et al., 2024; Doohan et al., 2024; Nash et al., 2024; Morgenstern et al., 2025; McCain et al., 2026) and two more are in the data extraction phase. For these seven pathogens, approximately 26.2% of the articles considered by PERG were available under open-access licensing through the bibliographic databases we queried (Appendix B and Appendix C). We verify that this open-access subset is not a systematically biased sample of the broader PERG corpus via a matched comparison against 1,004 closed-access articles, with performance differences being minimal (see Appendix B.2). We evaluated each AgentSLR stage with all pathogen data available, meaning the first seven are evaluated for screening, and four (Ebola, Lassa, SARS, and Zika) are evaluated for data extraction. After correspondence with PERG, we chose to exclude Marburg due to inconsistencies in data format, and MERS and Nipah because PERG's extraction phase is still in progress.

3.2. Models

AgentSLR is implemented to be compatible with both open- and closed-weight models, with tool calls and requests schematised through OpenAI's Responses and Chat Completions APIs. Unless otherwise stated, we evaluated OpenAI's open-weight `gpt-oss-120b` reasoning model for all primary results in Section 4 (OpenAI et al., 2025). To assess the robustness and generalisability of the pipeline across model families, we conducted ablations using OpenAI's `GPT-5.2`, Moonshot AI's `Kimi K2.5`, Z.AI's `GLM-4.7`, and DeepSeek's `DeepSeek-V3.2`. Attempts to evaluate Claude Opus 4.5 and Sonnet 4.5 resulted in streaming refusals.[1] All models had reasoning set to high where possible, with a maximum generation limit of 64K tokens per pass. Open-source models were hosted with `vllm` (Kwon et al., 2023) on a NVIDIA H200 cluster node.

For the PDF-to-Markdown conversion stage, we used the `mistral-ocr-2512` API endpoint (Mistral AI, 2025), a state-of-the-art OCR model well-suited to scanned documents with complex mathematical and tabular content. For reproducibility, AgentSLR is also configured to run with open-weight OCR models available on HuggingFace.

3.3. Metrics

Pipeline runtime. We evaluated AgentSLR first in terms of time efficiency relative to human annotators (Section 4.1). We recorded the total wall-clock time to complete the first five pipeline stages and compare this time to self-reported estimates by human experts per stage. The final stage – producing a final SLR for peer review – does not admit a reliable time estimate, as it includes deliberations over meta-analysis outside the scope of AgentSLR, so we omitted this stage from our comparison.

Individual pipeline stage evaluations. To assess pipeline quality, we validated against expert annotations on four

[1]We experience this refusal problem with all Claude models above version 4.0 (See Documentation from Anthropic). Potential causes and implications are discussed in Section 7.

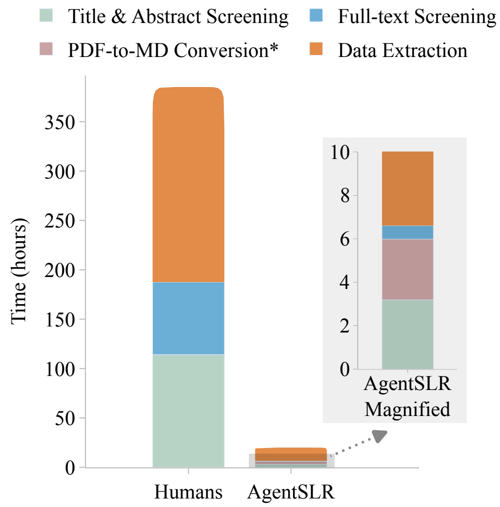


Figure 2. **SLR completion time.** AgentSLR (with `GPT-OSS-120B`) completes the workflow in 20 hours versus 385 hours taken for manual-conducted reviews (19.3× speed-up). Running continuously, this corresponds to less than 1 day (0.83) versus 48.1 human workdays (assuming 8-hour days), yielding 58× calendar-time savings. Times shown reflect processing of 9,132 articles at abstract screening, 1,102 at full-text screening and 395 at data extraction. For more information, see Appendix H.

priority pathogens: Ebola, Lassa, SARS and Zika. We designed stage-level evaluations for each of abstract screening, full-text screening and data extraction.

Each of the two article screening stages is framed as a binary classification task, so we considered classification metrics (precision, recall, and F_1) against ground-truth screening decisions. For data extraction, quantifying agreement with ground-truth annotations is less straightforward. Each article may contain arbitrarily many extractions, and individual extractions may differ across many metadata fields. For a holistic account of AgentSLR's performance, we designed three evaluation measures. *Flagging* assesses how reliably our pipeline identifies relevant parameter classes, models and outbreaks within an article. *Count* measures agreement in the number of extracted items per article. *Extraction* measures field-level accuracy by computing bipartite matches between our extractions and ground-truth extractions that maximise overall similarity. Appendix G provides formal definitions of the evaluation metrics and outlines pre-processing steps.

Human expert validation. Ground truth screening decisions and annotations provide clear standards for recall, but precision is less direct because additional extractions may be valid items omitted by human annotators. We therefore complement metric-based evaluation with a validation study involving six expert epidemiologists. Each expert is assigned a random subset of extracted data for parameters, models or outbreaks, along with the corresponding Markdown articles, and grades extraction correctness using a survey. Field-level correctness is grouped by category, such as temporal features or population context, and normalised so that each group receives equal weight. Experts also rate overall system capability from 1 to 7, where 4 is the threshold for a useful tool under human supervision. This validation is intended as a human centred check on metric construct validity, not as a duplicate annotation study, so we report expert judged correctness rather than inter-annotator agreement. Appendix G.3 describes the survey design and implementation.

4. Results

4.1. Full Pipeline Statistics

We demonstrate the base functionality of AgentSLR by running our pipeline for all nine priority pathogens using `gpt-oss-120b`. AgentSLR completes each report with an average wall clock time of 20 hours, processing 9,132 articles at title/abstract screening, 1,102 at full-text screening and 395 at data extraction.[2] Appendix K explains the report generation process in detail. Figure 2 shows the time comparison to human expert estimates, where AgentSLR's runtime ($\leq$ 1 day) represents a 19.3-times efficiency gain over the corresponding human processes (385 hours). Since AgentSLR runs continually, our runtime equates to a 58-times reduction in calendar days (assuming 8-hour workdays for humans). For full-text screening in particular, AgentSLR is 118 times faster than humans, reducing a 4-minute average down to below 2 seconds per article. The average cost of running AgentSLR per SLR varies by model and deployment: for example, self-hosting `gpt-oss-120b` on two Nvidia H200 GPUs costs approximately USD$137,[3] while using the OpenRouter API reduces the cost to USD$50 with higher latency as a trade-off. Appendix H.2 provides detailed comparisons across models and services.

4.2. Evaluation Against Ground-truth

Article screening. Figure 3 compares three article screening strategies tested across the seven evaluated pathogens. Under the default two-stage screening pipeline, AgentSLR achieves a recall of 0.81 against ground-truth screening decisions. To contextualise this performance, we consider two ablations. First, we condition full-text screening on ground-truth (human) abstract screening decisions, which improves recall to 0.92. Second, we omit abstract screening and process all full-texts directly, which improves recall to 0.89. Both trends are consistent across pathogens (Figure 3).

[2] We consider articles processed at each stage based on average across pathogens. See Appendix M for more details.

[3] This cost estimate includes OCR PDF-to-markdown conversion using `mistral-ocr-2512`.

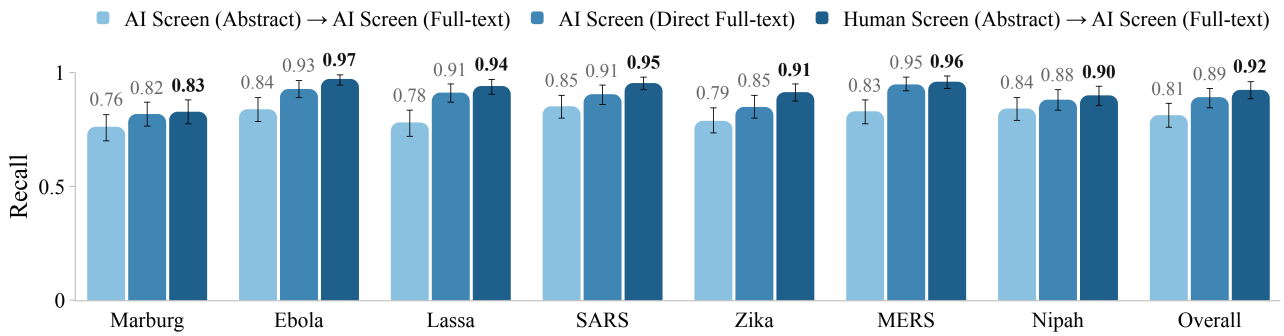


Figure 3. **Recall of article screening strategies across pathogens.** Two ablation screening strategies (human-conditioned, direct full-text) with AgentSLR (`GPT-OSS-120B`) offer better recall (or 'fetch rate') than performing traditional AI-based two stage screening, with bootstrapped confidence intervals (95% C.I.; 10,000 resamples) between the two ablations overlapping across most pathogens. Full article screening metrics along with individual title & abstract stage screening results are reported in Appendix I.1.

Direct full-text screening thus improves recall over the two-stage pipeline without human involvement, though at a $2.3\times$ increase in screening runtime (9.55 vs 4.16 hours) and a corresponding rise in OCR costs (USD$36.6 vs USD$303.2).

Data extraction. Table 2 presents our evaluation results for parameter, model and outbreak extraction. We report average classification measures (precision, recall and F_1) for each of our Flagging, Count and Extraction metrics. Across all data types, flagging achieves the highest average F_1 (0.75), with performance declining progressively through counting (0.65) and extraction (0.63), reflecting the compounding difficulty of each successive pipeline stage. See Appendix I.2 for complete disaggregated results across all data subtypes, pathogens and individual fields.

AgentSLR displays high recall (0.92) but moderate precision (0.51) for parameter class flagging. For parameter extraction counts, this trend reverses, suggesting that the agent identifies many parameter classes as potentially relevant, yet exercises more discretion when producing structured extractions. At the field level, performance is moderate across all pathogens. AgentSLR achieves near-perfect accuracy for method extraction and for specific uncertainty fields, while value fields and population context prove considerably more challenging (See Table 27).

AgentSLR's model extraction achieves strong flagging performance, with high recall (0.91) and precision (0.90). This high recall carries through to model counts (0.99), indicating that nearly all models from the ground truth data are recovered, albeit with lower precision for counting (0.52). At the field level, model extraction attains a precision of 0.63 and recall of 0.74: core structural characteristics (model type, stochastic vs. deterministic and code availability) are extracted reliably, while complex multi-value fields such as assumptions, interventions and transmission routes remain more challenging (See Table 28 for the complete results).

We evaluate outbreak extraction only for Lassa and Zika due to a lack of human annotation for Ebola and SARS. Article flagging shows moderate performance across both pathogens (precision 0.63, recall 0.76), and outbreak counting shows high variance (± 0.28 for recall) driven by pathogen-level differences. Despite this, field-level extraction is robust: outbreak extraction achieves the highest precision (0.85) amongst all data types, with particular strength in temporal features and case burden (See Table 30).

Table 2. **Evaluation metrics for data extraction stage (averaged across pathogens with ± deviation).** Results for AgentSLR (`gpt-oss-120b`) are reported separately for *Flagging*, *Count*, and *Extraction*, measuring presence identification, quantity accuracy, and value accuracy, respectively. Flagging achieves the strongest performance ($F_1 = 0.75$), followed by counting and extraction. For disaggregated metrics, see Appendix I.2.

Data Type	Precision	Recall	F_1 Score
Parameters			
Flagging	0.51 (± 0.07)	0.92 (± 0.06)	0.66 (± 0.06)
Count	0.83 (± 0.10)	0.47 (± 0.09)	0.59 (± 0.07)
Extraction	0.52 (± 0.03)	0.57 (± 0.04)	0.54 (± 0.02)
Models			
Flagging	0.90 (± 0.04)	0.91 (± 0.05)	0.91 (± 0.04)
Count	0.52 (± 0.05)	0.99 (± 0.01)	0.68 (± 0.04)
Extraction	0.63 (± 0.04)	0.74 (± 0.02)	0.67 (± 0.03)
Outbreaks			
Flagging	0.63 (± 0.06)	0.76 (± 0.05)	0.61 (± 0.09)
Count	0.66 (± 0.17)	0.72 (± 0.28)	0.69 (± 0.22)
Extraction	0.85 (± 0.00)	0.76 (± 0.02)	0.79 (± 0.01)
Average (across data types)			
Flagging	0.70 (± 0.05)	0.88 (± 0.04)	0.75 (± 0.06)
Count	0.67 (± 0.09)	0.73 (± 0.06)	0.65 (± 0.06)
Extraction	0.62 (± 0.07)	0.67 (± 0.03)	0.63 (± 0.05)

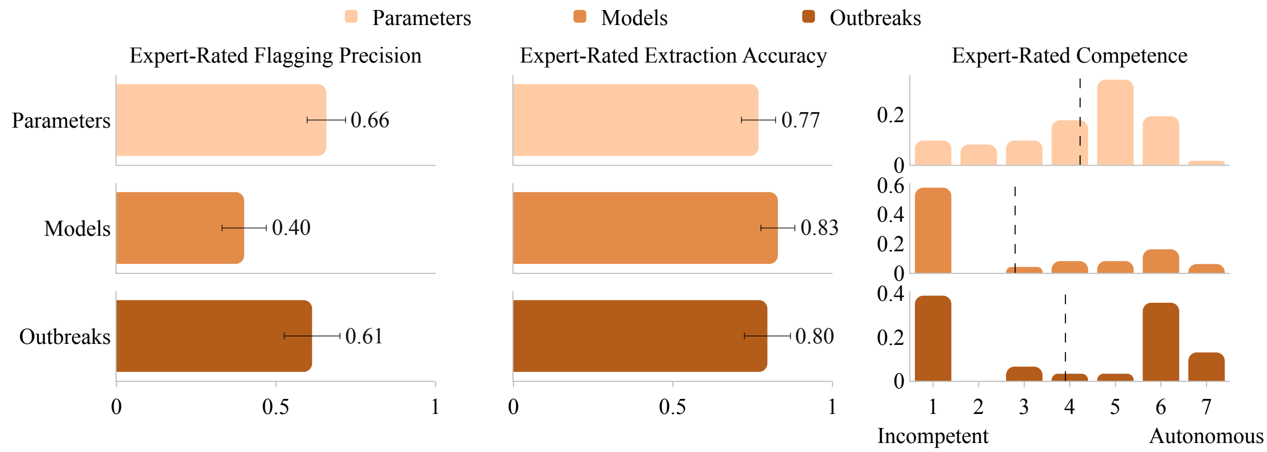


Figure 4. **Human expert evaluation of data extraction quality across stages.** We report expert-rated flagging precision, field-level extraction accuracy, and perceived AgentSLR (`gpt-oss-120b`) competence for parameter, model, and outbreak extractions, aggregated across six epidemiologists. Error bars denote standard errors, and dashed lines indicate mean competence ratings (4.2 for parameters, 2.8 for models, and 3.9 for outbreaks).

4.3. Human Expert Validation

Quantitative survey statistics. Figure 4 displays results from the human expert validations on AgentSLR (`gpt-oss-120b`) described in Section 3.3. Extraction Accuracy reports the average rate of field-level correctness conditional on the extraction being correctly flagged as relevant. In contrast, Expert-Rated Competence is assessed for every case, including incorrect flaggings. *Flagging Precision* reports the proportion of AgentSLR extractions judged relevant by experts and is higher for parameters (0.66) and outbreaks (0.61) than for models (0.40). *Extraction Accuracy* reports the average rate of field-level correctness. All extraction types achieve expert-rated correctness comparable to or exceeding the precision of our automated Extraction evaluation in Section 4.2 (0.77 for parameters; 0.83 for models; 0.80 for outbreaks).

Qualitative impressions. Based on survey feedback from six expert epidemiologists, AgentSLR was consistently reported to improve efficiency compared to fully manual extractions. Although false positives occur, these are typically easy to identify and remove, resulting in a net reduction in effort. Extraction difficulty varies across papers due to differences in complexity and reporting style, and in rare cases the system may increase effort in situations that are similarly challenging for human reviewers. Once an extraction is produced, individual fields are straightforward to validate, whereas multi-select fields pose more difficulty. Common errors arise from insufficient contextual information, limited use of document structure and cross-extraction constraints, and failures to infer fields apparent to human annotators when information is implicit. Additionally, the system struggles to understand provenance, occasionally mixing up newly reported findings with information cited from prior work.

5. Model Ablations

To contextualise the performance of `gpt-oss-120b`, we ran AgentSLR with four additional frontier LRMs, conducting ground-truth evaluations for each stage. We find variation in performance across stages with no clear best model (Figure 5). `Kimi-K2.5` and `gpt-oss-120b` perform best at the article screening stages, with the former excelling in title & abstract screening ($F_1 = 0.77$) and the latter in full-text screening($F_1 = 0.63$). All models struggle with parameter extraction, with the highest performance again achieved by `Kimi-K2.5` ($F_1 = 0.87$). `GLM-4.7` performs well specifically for extracting models, while `GPT-5.2` stands out when extracting outbreaks. `DeepSeek-V3.2` exhibits the most variable performance across stages. It is the worst-performing model by some margin during article screening, but becomes competitive during the extraction phase where it is enabled with function calling, most evidently in model and outbreak extraction.

The smallest model by size, `gpt-oss-120b`, performs within 4.5 percentage points of the best model across all stages. The next smallest model, `GLM-4.7`, is nearly 3 times larger at 358 billion parameters. Excluding outbreak extraction, where results are aggregated only across Zika and Lassa, `gpt-oss-120b` also exhibits one of the lowest variances across pathogens. The weaker outbreak extraction performance is driven by Zika ($F_1 = 0.60$), where `gpt-oss-120b` struggles consistently across stages. We

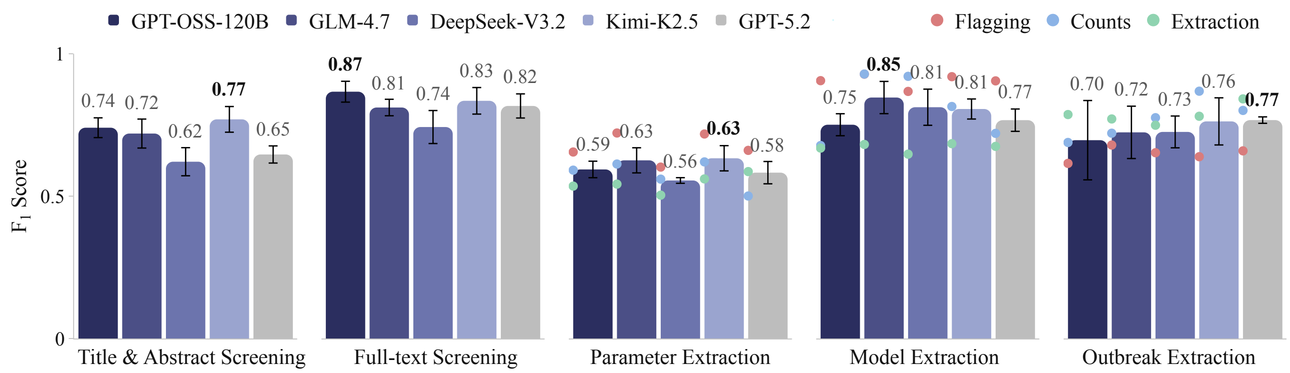


Figure 5. **Model ablation results with AgentSLR across all pipeline stages.** Averages are computed over the pathogens evaluated at each stage, following ground-truth availability described in Section 3.1. Error bars indicate one standard deviation across pathogens. For the three data extraction panels, coloured dots show the macro F1 of the Flagging ●, Counts ●, and Extraction ● sub-tasks, plotted to the left of each bar. No single model dominates across all stages: `Kimi-K2.5` and `gpt-oss-120b` lead screening, while extraction leaders vary by data type. Full pathogen-wise metrics are provided in Appendix J.

suspect this performance to be due to greater domain overlap and multi-pathogen co-occurrence, as Lassa outbreak extraction remains strong ($F_1 = 0.80$).

Consistent with Table 2, models find the successive data extraction sub-tasks progressively harder: Flagging typically outperforms both Counts and Extraction. Outbreaks remain the exception to this trend, consistently across models, with precision notably lower for Flagging. Outbreak events are typically reported once but repeated across many papers as disease background. Across all extraction stages, the gap between the best and worst performing models is also considerably narrower than in the screening stages.

We further contextualise the model ablations by cost[4] and aggregated performance across the full AgentSLR pipeline (Figure 6). Higher cost and larger models do not consistently yield higher performance. `gpt-oss-120b` achieves competitive average performance ($F_1 = 0.70$) at the lowest total cost ($13.9), over 96 times cheaper than `GPT-5.2` ($1,348). Despite being OpenAI's flagship closed-source model, `GPT-5.2` yields a lower average F_1 of 0.69. More broadly, the best-performing model, `Kimi-K2.5` ($F_1 = 0.74$), sits in the mid-cost range ($277), while `GLM-4.7` incurs the second-highest cost ($811) for a comparable average F_1 of 0.73. Variance across stages is also non-trivial for all models (ranging from ± 0.07 to ± 0.11), reflecting the uneven difficulty of pipeline stages as discussed above.

The substantial cost differences across models stem primarily from divergent per-article token usage, particularly at the parameter extraction stage. For example, `GPT-5.2` produces $91.1K$ output tokens per article compared to `DeepSeek-V3.2`'s $3.0K$. Parameter extraction dominates overall compute; full per-stage token and cost breakdowns are reported in Appendix H.2. Taken together with the stage-level capability differences observed in Section 5, these results suggest that the choice of LRM for AgentSLR involves a nuanced cost-performance trade-off, with no single model uniformly dominating across both dimensions.

[4] All costs reported are in USD at the time of our experiment.

6. Related Work

The human cost of conducting SLRs is concentrated in manual retrieval, screening, and evidence structuring (Page et al., 2021; Marshall & Wallace, 2019). Recent work with LLMs shows that prompt templates can transfer screening

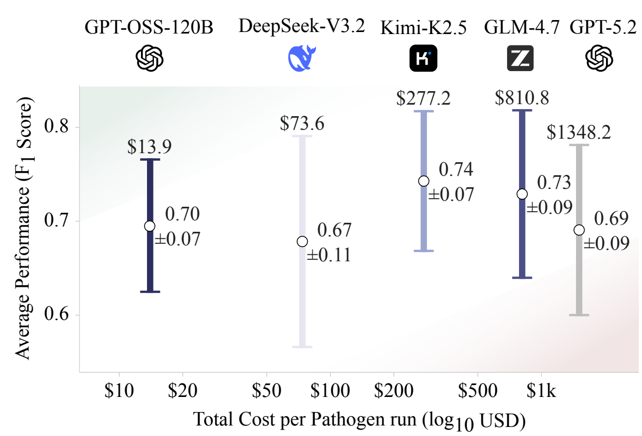


Figure 6. **Comparing total cost against average performance for an AgentSLR pathogen run.** Each point shows a model's average macro F_1 across all AgentSLR pipeline stages plotted against its estimated total cost per pathogen run (USD, $\log_{10}$ scale), with vertical bars indicating one standard deviation in F_1 across stages. Costs are estimated from mean per-article token usage across a funnel of 9,132 articles at abstract screening down to 395 at data extraction (see Figure 2 counts), multiplied by OpenRouter and OpenAI API pricing; full details in Appendix H.2.

logic across title, abstract and full-text stages without task-specific fine-tuning, achieving high sensitivity and specificity in multiple systematic reviews (Cao et al., 2025b; Homiar et al., 2025). For data extraction, LLMs perform well on constrained schemas but degrade on complex fields, with human-incorporated LLM workflows generally outperforming LLM-only approaches (Gartlehner et al., 2024; Mahmoudi et al., 2025; Lai et al., 2025). Building on these stage-specific advances, recent work has shifted towards end-to-end SLR pipelines that couple retrieval, screening, extraction and synthesis under agentic orchestration (Scherbakov et al., 2025). Some systems reproduce and update Cochrane-style intervention reviews, but rely on proprietary models and evaluate performance using LLM-as-a-judge with post-hoc “corrected” labels (Cao et al., 2025a). See Appendix A for extended comparison.

7. Discussion

AgentSLR achieves orders-of-magnitude efficiency gains while maintaining coverage. Our pipeline reduces active review time by a factor of 19.3, from 385 human labour hours to 20 hours, with full-text screening running 118 times faster than a human reviewer. These gains change the feasibility calculus for evidence synthesis on large, rapidly evolving corpora, with particular relevance where literature growth outpaces reviewer capacity (Bergstrom & Gross, 2026) or where timely synthesis is operationally critical (Orton et al., 2011; Clarke, 2017).

7.1. Implications

Human-in-the-loop deployment is the appropriate mode for automated SLR tools like AgentSLR. While AgentSLR lacks the contextual understanding to fully automate an epidemiological SLR, it delivers substantial efficiency gains within human-led processes. Manual review limits SLR scalability (Polanin et al., 2019), and full-text processing requires substantially more resources than abstract-only triage (Clark et al., 2020). Given our strong classification performance, AgentSLR is well-suited to expedite full-text screening after human abstract filtering. For data extraction, high recall ensures that relevant evidence persists for human validation, and experts report improved efficiency when provided with AgentSLR's outputs. By reducing the per-update burden that makes continuous curation infeasible, these capabilities could enable living systematic reviews for timely pandemic preparedness.

Our results also suggest that current open-weight models offer a viable foundation for scientific SLR deployment. Within our evaluation, open-weight models achieve performance comparable to closed-source frontier models while operating at substantially lower cost: `gpt-oss-120b` achieves similar performance ($F_1 = 0.70$) at over 96× lower cost than `GPT-5.2` ($F_1 = 0.69$), while `Kimi-K2.5` achieves the best overall performance ($F_1 = 0.74$) at a mid-range cost. Beyond cost, open-weight models permit version pinning and local deployment, properties that matter for long-running living reviews where reproducibility is a scientific requirement.

In addition, we encountered broad content restrictions from closed-source providers, which pose a risk for critical scientific applications. Attempts to evaluate AgentSLR using Claude Opus 4.5 and Sonnet 4.5 resulted in consistent streaming refusals, which we attribute to content filters triggered by epidemiological terminology being likened to bioweapons.[5] While such caution is understandable in consumer deployments, restrictions applied too broadly can render entire model families unavailable for legitimate public-health research, reinforcing the case for open-weight alternatives for both reproducibility and operational continuity.

7.2. Limitations

Our study has several limitations. Data coverage is restricted to open-access articles, matching roughly 26% of the ground-truth dataset, and English-only screening may introduce corpus-level bias. Our evaluation metrics prioritise recall over precision: parameter-class flagging achieves recall of 0.92 but precision of only 0.51, so downstream human filtering remains necessary. AgentSLR depends on long-context LRMs and OCR infrastructure, meaning deployment at scale is conditioned on compute availability and access to frontier models, which risks concentrating such capabilities within well-resourced institutions.

7.3. Future Work

This feasibility study suggests many exciting directions for future work. Most urgently, a proper human uplift study can be conducted to more robustly quantify the time savings and efficacy of a human-in-the-loop implementation. We are prototyping a human-in-loop annotation tool, explained in Appendix N, to be improved to production-grade and provided to epidemiologists conducting future SLRs. Human uplift is most compelling in the case of unknown or understudied diseases with serious epidemic potential (Mehand et al., 2018), or on priority pathogens where literature volume outpaces reviewer capacity, like COVID-19. More generally, while AgentSLR's implementation relies heavily on epidemiological domain knowledge, the framework it provides for SLR automation is extensible: future work could explore generalisation to additional scientific fields across the medical, social, and physical sciences, and investigate whether models can participate in defining their own extraction tools as domain knowledge shifts.

[5] See the Anthropic documentation on Sonnet 4.5 API safety filters.

References

Bean, A. M., Kearns, R. O., Romanou, A., Hafner, F. S., Mayne, H., Batzner, J., Foroutan, N., Schmitz, C., Korgul, K., Batra, H., Deb, O., Beharry, E., Emde, C., Foster, T., Gausen, A., Grandury, M., Han, S., Hofmann, V., Ibrahim, L., Kim, H., Kirk, H. R., Lin, F., Liu, G. K.-M., Luettgau, L., Magomere, J., Rystrøm, J., Sotnikova, A., Yang, Y., Zhao, Y., Bibi, A., Bosselut, A., Clark, R., Cohan, A., Foerster, J., Gal, Y., Hale, S. A., Raji, I. D., Summerfield, C., Torr, P. H. S., Ududec, C., Rocher, L., and Mahdi, A. Measuring what matters: Construct validity in large language model benchmarks. *The Thirty-ninth Annual Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2025.

Bergstrom, C. T. and Gross, K. Screening, sorting, and the feedback cycles that imperil peer review. *PLoS biology*, 24(2):e3003650, 2026.

Borah, R., Brown, A. W., Capers, P. L., and Kaiser, K. A. Analysis of the time and workers needed to conduct systematic reviews of medical interventions using data from the prospero registry. *BMJ open*, 7(2):e012545, 2017.

Cao, C., Arora, R., Cento, P., Manta, K., Farahani, E., Cecere, M., Selemon, A., Sang, J., Gong, L. X., Kloosterman, R., et al. Automation of systematic reviews with large language models. *medRxiv*, pp. 2025–06, 2025a.

Cao, C., Sang, J., Arora, R., Chen, D., Kloosterman, R., Cecere, M., Gorla, J., Saleh, R., Drennan, I., Teja, B., et al. Development of prompt templates for large language model–driven screening in systematic reviews. *Annals of Internal Medicine*, 178(3):389–401, 2025b.

Chai, K. E., Lines, R. L., Gucciardi, D. F., and Ng, L. Research screener: a machine learning tool to semi-automate abstract screening for systematic reviews. *Systematic Reviews*, 10(1):93, 2021.

Clark, J., Glasziou, P., Del Mar, C., Bannach-Brown, A., Stehlik, P., and Scott, A. M. A full systematic review was completed in 2 weeks using automation tools: a case study. *Journal of Clinical Epidemiology*, 121:81–90, 2020.

Clarke, M. Evidence aid: using systematic reviews to improve access to evidence for humanitarian emergencies., 2017.

Cuomo-Dannenburg, G., McCain, K., McCabe, R., Unwin, H. J. T., Doohan, P., Nash, R. K., Hicks, J. T., Charniga, K., Geismar, C., Lambert, B., Nikitin, D., Skarp, J., Wardle, J., Kont, M., Bhatia, S., Imai, N., van Elsland, S., Cori, A., Morgenstern, C., Morris, A., Forna, A., Dighe, A., Cori, A., Hamlet, A., Lambert, B., Whittaker, C., Morgenstern, C., Geismar, C., Nikitin, D., Jorgensen, D., Knock, E., Unwin, E., Cuomo-Dannenburg, G., Thompson, H., Routledge, I., Skarp, J., Hicks, J., Fraser, K., Charniga, K., McCain, K., Geidelberg, L., Cattarino, L., Kont, M., Baguelin, M., Imai, N., Moghaddas, N., Doohan, P., Nash, R., McCabe, R., van Elsland, S., Bhatia, S., Radhakrishnan, S., Cucunuba Perez, Z., and Wardle, J. Marburg virus disease outbreaks, mathematical models, and disease parameters: A systematic review. *The Lancet Infectious Diseases*, 24(5):e307–e317, 2024.

Doohan, P., Jorgensen, D., Naidoo, T. M., McCain, K., Hicks, J. T., McCabe, R., Bhatia, S., Charniga, K., Cuomo-Dannenburg, G., Hamlet, A., Nash, R. K., Nikitin, D., Rawson, T., Sheppard, R. J., Unwin, H. J. T., van Elsland, S., Cori, A., Morgenstern, C., Imai-Eaton, N., Morris, A., Forna, A., Dighe, A., Vicco, A., Hartner, A.-M., Cori, A., Hamlet, A., Lambert, B., Cracknell Daniels, B., Whittaker, C., Morgenstern, C., Santoni, C., Geismar, C., Nikitin, D., Jorgensen, D., Dee, D., Knock, E., Unwin, E., Cuomo-Dannenburg, G., Thompson, H., Dorigatti, I., Routledge, I., Wardle, J., Skarp, J., Hicks, J., Parchani, K., Fraser, K., Charniga, K., McCain, K., Drake, K., Geidelberg, L., Cattarino, L., Kusumgar, M., Kont, M., Baguelin, M., Imai-Eaton, N., Guzman, P. P., Doohan, P., Lietar, P., Christen, P., Nash, R., Fitzjohn, R., Sheppard, R., Johnson, R., McCabe, R., van Elsland, S., Bhatia, S., Leuba, S., Ruybal-Pesantez, S., Radhakrishnan, S., Rawson, T., Naidoo, T., and Cucunuba Perez, Z. Lassa fever outbreaks, mathematical models, and disease parameters: A systematic review and meta-analysis. *The Lancet Global Health*, 12(12):e1962–e1972, 2024.

Gartlehner, G., Kahwati, L., Hilscher, R., Thomas, I., Kugley, S., Crotty, K., Viswanathan, M., Nussbaumer-Streit, B., Booth, G., Erskine, N., Konet, A., and Chew, R. Data extraction for evidence synthesis using a large language model: A proof-of-concept study. *Research Synthesis Methods*, 15(4):576–589, 2024.

Gates, A., Johnson, C., and Hartling, L. Technology-assisted title and abstract screening for systematic reviews: a retrospective evaluation of the abstrackr machine learning tool. *Systematic Reviews*, 7(1):45, 2018a.

Gates, A., Vandermeer, B., and Hartling, L. Technology-assisted risk of bias assessment in systematic reviews: a prospective cross-sectional evaluation of the robotreviewer machine learning tool. *Journal of Clinical Epidemiology*, 96:54–62, 2018b.

He, W., Yi, G. Y., and Zhu, Y. Estimation of the basic reproduction number, average incubation time, asymptomatic infection rate, and case fatality rate for COVID-19: Meta-analysis and sensitivity analysis. *Journal of Medical Virology*, 92(11):2543–2550, 2020.

Homiar, A., Thomas, J., Ostinelli, E. G., Kennett, J., Friedrich, C., Cuijpers, P., Harrer, M., Leucht, S., Miguel, C., Rodolico, A., et al. Development and evaluation of prompts for a large language model to screen titles and abstracts in a living systematic review. *BMJ Mental Health*, 28(1), 2025.

Jonker, R. and Volgenant, A. A shortest augmenting path algorithm for dense and sparse linear assignment problems. *Computing*, 38(4):325–340, 1987.

Khraisha, Q., Put, S., Kappenberg, J., Warraitch, A., and 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. *Research Synthesis Methods*, 15(4):616–626, 2024.

Kwa, T., West, B., Becker, J., Deng, A., Garcia, K., Hasin, M., Jawhar, S., Kinniment, M., Rush, N., von Arx, S., Bloom, R., Broadley, T., Du, H., Goodrich, B., Jurkovic, N., Miles, L. H., Nix, S., Lin, T., Parikh, N., Rein, D., Sato, L. J. K., Wijk, H., Ziegler, D. M., Barnes, E., and Chan, L. Measuring AI ability to complete long tasks. *CoRR*, abs/2503.14499, 2025.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Lai, H., Liu, J., Bai, C., Liu, H., Pan, B., Luo, X., Hou, L., Zhao, W., Xia, D., Tian, J., et al. Language models for data extraction and risk of bias assessment in complementary medicine. *npj Digital Medicine*, 8(1):74, 2025.

Lu, C., Lu, C., Lange, R. T., Foerster, J., Clune, J., and Ha, D. The AI Scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

Mahmoudi, H., Chang, D., Lee, H., Ghaffarzadegan, N., and Jalali, M. S. Critical assessment of large language models'(ChatGPT) performance in data extraction for systematic reviews: Exploratory study. *JMIR AI*, 4(1): e68097, 2025.

Marshall, I. J. and Wallace, B. C. Toward systematic review automation: a practical guide to using machine learning tools in research synthesis. *Systematic Reviews*, 8(1):163, 2019.

McCain, K., Vicco, A., Morgenstern, C., Rawson, T., Naidoo, T. M., Bhatia, S., Dee, D. P., Doohan, P., Fraser, K., Hartner, A.-M., et al. A systematic review and meta-analysis of Zika virus epidemiology. *Nature Health*, pp. 1–13, 2026.

Mehand, M. S., Al-Shorbaji, F., Millett, P., and Murgue, B. The WHO R&D blueprint: 2018 review of emerging infectious diseases requiring urgent research and development efforts. *Emerging Infectious Diseases*, 24(9):e1–e8, 2018.

Michelson, M. and Reuter, K. The significant cost of systematic reviews and meta-analyses: a call for greater involvement of machine learning to assess the promise of clinical trials. *Contemporary Clinical Trials Communications*, 16:100443, 2019.

Mistral AI. Mistral OCR 3, 2025.

Morgenstern, C., Rawson, T., Routledge, I., Kont, M., Imai-Eaton, N., Skarp, J., Doohan, P., McCain, K., Johnson, R., Unwin, H. J. T., Naidoo, T., Dee, D. P., Parchani, K., Cracknell Daniels, B. N., Vicco, A., Drake, K. O., Christen, P., Sheppard, R. J., Leuba, S. I., Hicks, J. T., McCabe, R., Nash, R. K., Santoni, C. N., Cuomo-Dannenburg, G., van Elsland, S., Bhatia, S., Cori, A., Morris, A., Forna, A., Dighe, A., Vicco, A., Hartner, A.-M., Cori, A., Hamlet, A., Lambert, B., Cracknell Daniels, B., Whittaker, C., Morgenstern, C., Santoni, C., Geismar, C., Nikitin, D., Jorgensen, D., Dee, D., Knock, E., Unwin, E., Cuomo-Dannenburg, G., Thompson, H., Dorigatti, I., Routledge, I., Wardle, J., Skarp, J., Hicks, J., Parchani, K., Fraser, K., Charniga, K., McCain, K., Drake, K., Geidelberg, L., Cattarino, L., Kusumgar, M., Kont, M., Baguelin, M., Imai-Eaton, N., Perez Guzman, P., Doohan, P., Lietar, P., Christen, P., Nash, R., Fitzjohn, R., Sheppard, R., Johnson, R., McCabe, R., van Elsland, S., Bhatia, S., Leuba, S., Ruybal-Pesantez, S., Radhakrishnan, S., Rawson, T., Naidoo, T., and Cucunuba Perez, Z. Severe acute respiratory syndrome (SARS) mathematical models and disease parameters: A systematic review. *The Lancet Microbe*, 6 (9), 2025.

Naidoo, T., Nash, R., Morgenstern, C., Doohan, P., McCabe, R., Lambert, J., Sheppard, R., Santoni, C., Rawson, T., Ruybal-Pesántez, S., Unwin, J. H., Cuomo-Dannenburg, G., McCain, K., Hicks, J., Cori, A., and Bhatia, S. *Epireview: Tools to Update and Summarise the Latest Pathogen Data from the Pathogen Epidemiology Review Group (PERG)*, 2025.

Nash, R., Morgenstern, C., Bhatia, S., Sheppard, R., Hicks, J., Cuomo-Dannenburg, G., McCabe, R., McCain, K., Vicco, A., Doohan, P., and Naidoo, T. *Priority-Pathogens*, 2026.

Nash, R. K., Bhatia, S., Morgenstern, C., Doohan, P., Jorgensen, D., McCain, K., McCabe, R., Nikitin, D., Forna, A., Cuomo-Dannenburg, G., Hicks, J. T., Sheppard, R. J., Naidoo, T., van Elsland, S., Geismar, C., Rawson, T., Leuba, S. I., Wardle, J., Routledge, I., Fraser, K., Imai-Eaton, N., Cori, A., and Unwin, H. J. T. Ebola virus disease mathematical models and epidemiological parameters: A systematic review. *The Lancet Infectious Diseases*, 24(12):e762–e773, 2024.

Oami, T., Okada, Y., and Nakada, T.-a. Performance of a large language model in screening citations. *JAMA Network Open*, 7(7):e2420496–e2420496, 2024.

OpenAI, Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A., Arbus, E., Arora, R. K., Bai, Y., Baker, B., Bao, H., Barak, B., Bennett, A., Bertao, T., Brett, N., Brevdo, E., Brockman, G., Bubeck, S., Chang, C., Chen, K., Chen, M., Cheung, E., Clark, A., Cook, D., Dukhan, M., Dvorak, C., Fives, K., Fomenko, V., Garipov, T., Georgiev, K., Glaese, M., Gogineni, T., Goucher, A., Gross, L., Guzman, K. G., Hallman, J., Hehir, J., Heidecke, J., Helyar, A., Hu, H., Huet, R., Huh, J., Jain, S., Johnson, Z., Koch, C., Kofman, I., Kundel, D., Kwon, J., Kyrylov, V., Le, E. Y., Leclerc, G., Lennon, J. P., Lessans, S., Lezcano-Casado, M., Li, Y., Li, Z., Lin, J., Liss, J., Lily Liu, Liu, J., Lu, K., Lu, C., Martinovic, Z., McCallum, L., McGrath, J., McKinney, S., McLaughlin, A., Mei, S., Mostovoy, S., Mu, T., Myles, G., Neitz, A., Nichol, A., Pachocki, J., Paino, A., Palmie, D., Pantuliano, A., Parascandolo, G., Park, J., Pathak, L., Paz, C., Peran, L., Pimenov, D., Pokrass, M., Proehl, E., Qiu, H., Raila, G., Raso, F., Ren, H., Richardson, K., Robinson, D., Rotsted, B., Salman, H., Sanjeev, S., Schwarzer, M., Sculley, D., Sikchi, H., Simon, K., Singhal, K., Song, Y., Stuckey, D., Sun, Z., Tillet, P., Toizer, S., Tsimpourlas, F., Vyas, N., Wallace, E., Wang, X., Wang, M., Watkins, O., Weil, K., Wendling, A., Whinnery, K., Whitney, C., Wong, H., Yang, L., Yang, Y., Yasunaga, M., Ying, K., Zaremba, W., Zhan, W., Zhang, C., Zhang, B., Zhang, E., and Zhao, S. Gpt-oss-120b & gpt-oss-20b Model Card, 2025.

Orton, L., Lloyd-Williams, F., Taylor-Robinson, D., O'Flaherty, M., and Capewell, S. The use of research evidence in public health decision making processes: systematic review. *PloS one*, 6(7):e21704, 2011.

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., et al. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, 372, 2021.

Pan, M. Z., Cemri, M., Agrawal, L. A., Yang, S., Chopra, B., Tiwari, R., Keutzer, K., Parameswaran, A., Ramchandran, K., Klein, D., et al. Why do multiagent systems fail? In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*, 2025.

Parkinson, R. H., Cerbone, H., Mieskolainen, M., Cao, S., Wilson, A. D., Albacete, S., Armstrong, E. B., Bass, C., Botías, C., Brown, A., et al. Metabeeai: an AI pipeline for full-text systematic reviews in biology. *bioRxiv*, pp. 2025–11, 2025.

Peters, U. and Chin-Yee, B. Generalization bias in large language model summarization of scientific research. *Royal Society Open Science*, 12(4):241776, 2025.

Polanin, J. R., Pigott, T. D., Espelage, D. L., and Grotpeter, J. K. Best practice guidelines for abstract screening large-evidence systematic reviews and meta-analyses. *Research Synthesis Methods*, 10(3):330–342, 2019.

Przybyła, P., Brockmeier, A. J., Kontonatsios, G., Le Pogam, M.-A., McNaught, J., von Elm, E., Nolan, K., and Ananiadou, S. Prioritising references for systematic reviews with RobotAnalyst: a user study. *Research Synthesis Methods*, 9(3):470–488, 2018.

Rein, D., Hou, B. L., Stickland, A. C., Petty, J., Pang, R. Y., Dirani, J., Michael, J., and Bowman, S. R. GPQA: A graduate-level Google-proof Q&A benchmark. In *First Conference on Language Modeling*, 2024.

Roberts, J., Han, K., Houlsby, N., and Albanie, S. SciFIBench: Benchmarking large multimodal models for scientific figure interpretation. *Advances in Neural Information Processing Systems*, 37:18695–18728, 2024.

Scherbakov, D., Hubig, N., Jansari, V., Bakumenko, A., and Lenert, L. A. The emergence of large language models as tools in literature reviews: a large language model-assisted systematic review. *Journal of the American Medical Informatics Association*, 32(6):1071–1086, 2025.

Syriani, E., David, I., and Kumar, G. Screening articles for systematic reviews with ChatGPT. *Journal of Computer Languages*, 80:101287, 2024.

Thomas, J., McDonald, S., Noel-Storr, A., Shemilt, I., Elliott, J., Mavergames, C., and Marshall, I. J. Machine learning reduced workload with minimal risk of missing studies: development and evaluation of a randomized controlled trial classifier for cochrane reviews. *Journal of Clinical Epidemiology*, 133:140–151, 2021.

Tian, M., Gao, L., Zhang, S., Chen, X., Fan, C., Guo, X., Haas, R., Ji, P., Krongchon, K., Li, Y., et al. SciCode: A research coding benchmark curated by scientists. *Advances in Neural Information Processing Systems*, 37: 30624–30650, 2024.

Wallace, B. C., Trikalinos, T. A., Lau, J., Brodley, C., and Schmid, C. H. Semi-automated screening of biomedical citations for systematic reviews. *BMC bioinformatics*, 11 (1):55, 2010.

Wang, H., Fu, T., Du, Y., Gao, W., Huang, K., Liu, Z., Chandak, P., Liu, S., Van Katwyk, P., Deac, A., Anandkumar, A., Bergen, K., Gomes, C. P., Ho, S., Kohli, P., Lasenby, J., Leskovec, J., Liu, T.-Y., Manrai, A., Marks, D., Ramsundar, B., Song, L., Sun, J., Tang, J., Veličković, P., Welling, M., Zhang, L., Coley, C. W., Bengio, Y., and Zitnik, M. Scientific discovery in the age of artificial intelligence. *Nature*, 620(7972):47–60, 2023.

Ward, J., Gressani, O., Kim, S., Hens, N., and Edmunds, W. J. The epidemiology of pathogens with pandemic potential: A review of key parameters and clustering analysis. *Epidemics*, 54:100882, 2026.

World Health Organization. Pathogens prioritization: A scientific framework for epidemic and pandemic research preparedness. Technical report, World Health Organization, 2024.

Zahavi, I. and Einav, S. How large language models can help us write a systematic review. *Intensive Care Medicine*, 2025.

Zhang, Y., Khan, S. A., Mahmud, A., Yang, H., Lavin, A., Levin, M., Frey, J., Dunnmon, J., Evans, J., Bundy, A., et al. Exploring the role of large language models in the scientific method: from hypothesis to discovery. *npj Artificial Intelligence*, 1(1):14, 2025.

Appendix

A. Extended Related Work

Prior LLM-based SLR work has addressed individual pipeline stages in isolation (Cao et al., 2025b; Gartlehner et al., 2024), and more recent end-to-end systems have demonstrated feasibility in clinical and biological domains (Cao et al., 2025a; Parkinson et al., 2025; **?**). Each existing system is bespoke to its target domain, uses incompatible ground-truth schemas, and cannot be transferred to epidemiological parameter extraction. Direct quantitative comparison across systems is therefore methodologically ill-posed. Table 3 provides a structured qualitative comparison across domain, methodology, and evaluation rigour.

Table 3. **Comparison against related work.** End-to-end LLM-based SLR pipelines differ substantially in domain, extraction methodology, and evaluation rigour, making direct quantitative comparison methodologically ill-posed. ✓ and × denote presence and absence of each property. *Open code*: source code is publicly released. *Open weight*: compatible with open-weight models. *Human reference*: evaluation uses expert-curated annotations as ground truth. *Independent evaluation*: performance is not assessed using LLM-as-a-judge. *Stage evaluation*: evaluation is decomposed into stage-specific metrics enabling failure attribution. AgentSLR is the only system satisfying all five properties.

	otto-SR (Cao et al., 2025a)	**A4SLR (?)**	**MetaBeeAI (Parkinson et al., 2025)**	**AgentSLR (ours)**
Domain	Clinical evidence	Health economics	Biology	Epidemiology
Methodological focus	Cochrane-style intervention reviews	HEOR and HTA use cases	Bee ecotoxicology	WHO-priority pathogen epidemiology
Extraction method	LLM extraction with post-hoc human correction	Prompt-based with domain templates	Multi-pass chunk retrieval	Tool-calling with JSON schema constraints
Open code	×	×	✓	✓
Open weight	×	×	×	✓
Human reference	✓	✓	✓	✓
Independent evaluation	×	✓	✓	✓
Stage evaluation	×	✓	×	✓

AgentSLR is distinguished on three axes that no existing pipeline jointly satisfies. It is the first system to target epidemiological SLRs on WHO-priority pathogens, the first end-to-end pipeline compatible with open-weight models with disclosed source code, and the first to report stage-isolated evaluation against expert-curated annotations without LLM-as-a-judge. Each property has direct operational consequences. Open-weight compatibility enables version pinning and local deployment, which are requirements for living reviews where reproducibility is a scientific constraint. During our evaluation, all Claude 4.0+ models consistently refused to process epidemiological content, rendering entire closed-source model families unavailable for legitimate public-health research. Stage-isolated evaluation enables failure attribution at the component level, supporting targeted improvement without confounding from cross-stage error propagation.

B. Evaluation Data

We evaluate AgentSLR against systematic literature reviews (SLRs) produced by the Pathogen Epidemiology Review Group (PERG), whose curated article sets and extracted data are made available through the `epireview` and `priority-pathogen` R packages (Naidoo et al., 2025; Nash et al., 2026). The evaluation spans up to seven WHO priority pathogens (World Health Organization, 2024), with screening evaluated across all seven and structured extraction evaluated across four (Ebola, Lassa, SARS-CoV-1, and Zika) for which PERG's published SLRs and extraction artefacts were both complete and consistently formatted. This section details the article corpus underlying these evaluations and provides evidence that the open-access subset used for benchmarking is representative of the broader PERG population corpus.

B.1. Corpus Composition

Article retrieval was performed via the OpenAlex bibliographic database (see Appendix C for queries used). After deduplication and empty-abstract removal, AgentSLR retrieved a total of 75,191 articles across nine priority pathogens. Of the 60,233 articles in the corresponding PERG corpora (excluding pathogens not yet screened), 15,781 (26.2%) were matched to articles accessible through open-access licensing at the time of retrieval. Table 1 reports per-pathogen counts and match rates.

The open-access match rate varies across pathogens, ranging from 16.0% (SARS-CoV-1) to 45.5% (Nipah virus), reflecting heterogeneity in publication practices and licensing across disease areas and time periods. Marburg was excluded from evaluation due to inconsistencies in PERG's data format; MERS-CoV and Nipah virus were excluded because PERG's extraction phase remains in progress.

B.2. Data Representativeness (Ecological Validity)

A potential concern is whether the 26.2% open-access overlap constitutes a biased sample of the broader PERG corpus, which would affect the validity of reported performance estimates. To assess this directly, we manually retrieved (through institutional access) and processed 1,004 closed-access articles across four pathogens (Ebola, Lassa, SARS-CoV-1, and Zika) that appear in the broader PERG population corpus but were not accessible through OpenAlex at the time of retrieval. We then constructed a matched stratified random sample from the AgentSLR open-access subset and ran the full `gpt-oss-120b` pipeline on both groups under identical conditions.

Table 4. **Representativeness of the open-access evaluation subset across pipeline stages.** Macro F1 with 95% bootstrap confidence intervals for the AgentSLR open-access evaluation sample and a matched sample of 1,004 closed-access articles from the broader PERG population corpus, retrieved via institutional access across Ebola, Lassa, SARS-CoV-1, and Zika. Both groups are processed using `gpt-oss-120b` under identical conditions. Δ F1 denotes the signed difference (open-access minus population). Differences are small across all stages (range: -5.8 to $+3.1$ pp), with no difference statistically distinguishable from zero at the 5% level, indicating the open-access subset is not a systematically biased sample of the broader corpus.

Stage	AgentSLR (Open-Access)	PERG Population	Δ F1
Abstract Screening	0.877 [0.853, 0.899]	0.906 [0.878, 0.930]	−0.028 [−0.062, 0.010]
Full-text Screening	0.860 [0.830, 0.886]	0.918 [0.890, 0.942]	−0.058 [−0.093, −0.017]
Parameter Extraction	0.728 [0.654, 0.799]	0.697 [0.607, 0.781]	+0.031 [−0.079, 0.142]

Table 4 reports stage-level macro F1 with 95% bootstrap confidence intervals for the AgentSLR open-access sample and the PERG population sample (closed-acess). Performance is broadly comparable across all evaluated stages. The largest gap is in full-text screening (-5.8 percentage points), while parameter extraction shows a marginal improvement in the population sample ($+3.1$ pp). All confidence intervals overlap substantially, and no difference is statistically distinguishable from zero at the 5% level. The two subsets are temporally similar, with median publication years of 2018 (open-access) and 2016 (population).

C. Article Search and Retrieval

This section details the search query construction, database-specific adaptations, and PDF retrieval strategy used for article acquisition across priority pathogens in the AgentSLR pipeline. Following the Pathogen Epidemiology Review Group (PERG) methodology[6], we developed a standardised base query structure that captures core epidemiological domains including transmission dynamics, disease severity, temporal parameters, transmission heterogeneity, and evolutionary characteristics.

Different bibliographic databases support different search capabilities, requiring tailored query implementations. We maintain two versions of each pathogen query: one for PubMed and Europe PMC (which support wildcard truncation operators using $\star$), and another for OpenAlex (which requires fully expanded term variants).

Base Search Query (PubMed and Europe PMC) The base query for PubMed and Europe PMC uses Boolean operators with truncation symbols to capture morphological term variations:

```
[PATHOGEN_IDENTIFIER] AND (
   (transmissi* OR epidemiolog*) OR
   (model* NOT imag*) OR
   (severity OR "case fatality ratio*" OR CFR OR "case fatality rate*"
      OR "mortality rate*" OR "attack rate*") OR
   ("infectious period*" OR "serial interval*" OR "incubation period*"
      OR "generation time*" OR "generation interval*" OR "latent period*"
      OR latency) OR
   (heterogeneit* OR superspread* OR "super spread*" OR super-spread*
      OR overdispersion OR overdispersed OR over-dispersion OR over-dispersed
      OR "over dispersion" OR "over dispersed") OR
   (infectivity OR infectiousness OR "growth rate*" OR "reproduction number*"
      OR "reproductive number*" OR R0 OR "reproduction ratio*"
      OR "reproductive rate*") OR
   ("pre-existing immunity" OR serological OR serology OR serosurvey*) OR
   (evolution* OR mutation* OR substitution*) OR
   (outbreak* OR cluster* OR epidemic*) OR
   ("risk factor*")
   [ADDITIONAL_TERMS]
) [EXCLUSION_CRITERIA]
```

OpenAlex Adapted Queries

Because the OpenAlex API does not support wildcard operators[7] and strips these characters during query processing, we expanded all truncated terms into their common morphological variants:

```
[PATHOGEN_IDENTIFIER] AND (
   (transmission OR transmissibility OR transmissible OR transmitted
      OR transmitting OR transmit OR epidemiology OR epidemiological
      OR epidemiologic) OR
   (model OR models OR modeling OR modelling OR modeled OR modelled
      NOT (image OR images OR imaging)) OR
   (severity OR "case fatality ratio" OR "case fatality ratios" OR CFR
      OR "case fatality rate" OR "case fatality rates" OR "mortality rate"
      OR "mortality rates" OR "attack rate" OR "attack rates") OR
   ("infectious period" OR "infectious periods" OR "serial interval"
      OR "serial intervals" OR "incubation period" OR "incubation periods"
      OR "generation time" OR "generation interval" OR "generation intervals"
      OR "latent period" OR "latent periods" OR latency) OR
   (heterogeneity OR heterogeneous OR superspread OR superspreader
      OR superspreaders OR superspreading OR "super spread"
      OR "super spreader" OR "super spreaders" OR "super spreading"
      OR overdispersion OR overdispersed OR "over dispersion"
      OR "over dispersed") OR
   (infectivity OR infectiousness OR "growth rate" OR "growth rates"
```

[6] https://github.com/mrc-ide/priority-pathogens/wiki/Search-terms

[7] https://docs.openalex.org/how-to-use-the-api/get-lists-of-entities/search-entities

```
      OR "reproduction number" OR "reproduction numbers"
      OR "reproductive number" OR "reproductive numbers" OR R0
      OR "reproduction ratio" OR "reproduction ratios"
      OR "reproductive rate" OR "reproductive rates"
      OR "basic reproduction number") OR
   ("pre-existing immunity" OR serological OR serology OR serosurvey
      OR serosurveys OR seroprevalence OR serosurveillance) OR
   (evolution OR evolutionary OR evolving OR evolved OR mutation
      OR mutations OR mutant OR mutants OR mutate OR mutated
      OR substitution OR substitutions) OR
   (outbreak OR outbreaks OR cluster OR clusters OR clustering
      OR epidemic OR epidemics OR pandemic OR pandemics) OR
   ("risk factor" OR "risk factors")
   [ADDITIONAL_TERMS]
) [EXCLUSION_CRITERIA]
```

Pathogen-Specific Query Modifications

Table 5 summarises the pathogen-specific modifications applied across all database implementations. Most pathogens require only customised identifiers to ensure relevant literature retrieval. However, the queries for SARS explicitly exclude COVID-19 literature to prevent cross-contamination with SARS-CoV-2 studies. Similarly, queries for Zika include vector-specific epidemiological parameters (extrinsic incubation period, vector competence) that are essential for capturing mosquito-borne transmission dynamics. For Rift Valley fever, Crimean-Congo hemorrhagic fever (CCHF) and MERS, we incorporated additional virus-specific identifiers and spelling variants to enhance retrieval comprehensiveness. Despite these modifications, all databases share consistent pathogen identifiers and exclusion criteria, differing only in their use of wildcard forms (PubMed/Europe PMC) versus expanded term variants (OpenAlex).

Table 5. **Pathogen-specific modifications to the standardised search query**. All databases share consistent pathogen identifiers and exclusion criteria; PubMed/Europe PMC use wildcard forms while OpenAlex uses expanded variants.

Pathogen	**PATHOGEN_IDENTIFIER**	**ADDITIONAL_TERMS**	**EXCLUSION_CRITERIA**
Marburg virus	Marburg virus	—	—
Ebola virus	Ebola	—	—
Lassa virus	Lassa	—	—
SARS-CoV-1	SARS OR SARS-CoV-1 OR “Severe acute respiratory syndrome"	—	NOT (COVID-19 OR SARS-CoV-2)
Zika virus	zika	OR (“extrinsic incubation period" OR “EIP" OR “vector competence" OR “vectorial capacity")[†]	—
Nipah virus	Nipah	—	—
MERS-CoV	MERS OR MERS-CoV OR “Middle East respiratory syndrome" OR “Middle East Respiratory Syndrome Coronavirus"[‡]	—	—
Rift Valley fever virus	“Rift valley fever" OR RVF OR “Rift Valley Fever Virus" OR RVFV[‡]	—	—
CCHF virus	“Crimean Congo haemorrhagic fever" OR “Crimean-Congo hemorrhagic fever" OR CCHF OR “CCHF virus" OR CCHFV[‡]	—	—

[†]Vector-specific terms capture mosquito transmission parameters unique to arboviral epidemiology.
[‡]Expanded identifiers include alternative spellings (American/British English), virus-specific nomenclature, and common abbreviations for comprehensive coverage.

Metadata Extraction and Deduplication We extract bibliographic metadata from each database as summarised in Table 6. OpenAlex provides direct PDF URLs and internal work identifiers, PubMed supplies standardised medical literature identifiers (PMID: PubMed ID; PMCID: PubMed Central ID), and Europe PMC offers full-text availability metadata. The Digital Object Identifier (DOI) serves as a persistent identifier across databases.

We implement a hierarchical five-level deduplication strategy:

1. **DOI-based**: Normalised DOI strings (case-insensitive, URL prefixes stripped);
2. **PMID-based**: Numeric PMID extraction and normalisation;
3. **PMCID-based**: Normalised PMC identifiers (uppercase, "PMC" prefix standardised);
4. **OpenAlex ID-based**: Internal OpenAlex work identifiers;
5. **Title-year combination**: Normalised title strings (lowercase, alphanumeric only) paired with publication year.

When duplicate records are detected, identifier fields (DOI, PMID, PMCID, OpenAlex ID, URLs) preserve all non-null values while narrative fields (title, abstract, journal) retain the first non-null value. Source provenance is marked as "Both" when records appear in multiple databases.

Table 6. **Metadata fields extracted during article search.** PMID: PubMed ID; PMCID: PubMed Central ID; DOI: Digital Object Identifier.

Field	Description
article_id	Generated unique identifier
source	Database origin
pmid	PubMed Identifier
pmcid	PubMed Central Identifier
doi	Digital Object Identifier
title	Article title
authors	Semicolon-delimited author list
journal	Publication venue
year	Publication year
abstract	Article abstract
url	Canonical article URL
openalex_id	OpenAlex work identifier
openalex_pdf_url	Direct PDF link from OpenAlex
pathogen	Target pathogen
query	Search query used
harvested_at	ISO 8601 timestamp

Table 7. **Additional fields populated during PDF retrieval attempts.**

Field	Description
downloaded	Boolean success flag
downloaded_path	Filesystem path to PDF
download_source	Source that provided PDF
download_attempted_at	ISO 8601 timestamp
download_error	Error messages from attempts

PDF Retrieval We attempt PDF downloads through multiple open access sources using a cascading retrieval strategy. Before attempting downloads, available identifiers (PMID, PMCID, DOI) are cross-referenced using NCBI's PMC ID Converter API[8] to maximise source compatibility. The system then attempts downloads from up to four sources in priority order (Table 8), stopping at the first successful retrieval.

Implementation Details

Downloads employ HTTP streaming to temporary files with 64 KB chunks and validate each file through two stages: (1) magic byte verification (`%PDF` header), and (2) content inspection for HTML access denial pages. Files exceeding 500 MB or failing validation are immediately discarded. Thread-pool parallelism with 16 workers processes downloads concurrently while respecting per-source rate limits. In-memory caches keyed by normalised identifiers store both successful PDF URLs and negative markers to eliminate redundant API calls. Progress is checkpointed every 50 records for crash recovery.

Successfully validated PDFs are saved with standardised filenames following identifier priority (PMID → PMCID → DOI hash → title hash). Metadata is augmented with download provenance including source, timestamp, and error diagnostics.

Final Quality Control After retrieval, we applied deduplication and quality filtering that removes: records lacking abstracts, duplicate article IDs, duplicate DOIs (retaining first occurrence) and records with file validation failures.

[8] https://www.ncbi.nlm.nih.gov/pmc/tools/id-converter-api/

Table 8. **PDF retrieval sources in cascading priority order.** Sources are queried sequentially until success or exhaustion. Identifier cross-referencing via NCBI PMC ID Converter API precedes all download attempts (10 req/s, cached).

Priority	Source & Endpoint	Rate Limit	Cached
1	**OpenAlex Direct PDF URL** Metadata field `openalex_pdf_url`	30 req/s	No
2	**Europe PMC Fulltext API** `ebi.ac.uk/europepmc/webservices/rest/search`	20 req/s	Yes
3	**Unpaywall API** `api.unpaywall.org/v2/{DOI}?email={EMAIL}`	50 req/s	Yes
4	**OpenAlex DOI Lookup** `api.openalex.org/works/https://doi.org/{DOI}`	30 req/s	Yes

D. Article Screening Criteria and Prompts

Following article search and retrieval, the articles are screened for relevance to the study. The screening is conducted on abstracts, and then on full-text articles. We present the study objectives, inclusion and exclusion criteria, along with the detailed prompts used to screen for relevant priority pathogen articles. We take inspiration from (Cao et al., 2025b), and their ScreenPrompt structure to build our article screening prompts. The prompts follow a structured format: basic instruction, study objectives, inclusion/exclusion criteria, article content, and chain-of-thought screening instructions with parsable output request.

Study Objectives

This systematic review aims to collate transmission and modelling parameters for `{pathogen_name}`. The review seeks to:

1. Provide estimates of key infectious disease metrics (reproduction number, CFR, generation time, serial interval, incubation period, etc.)
2. Document historical outbreak characteristics (size, location, duration, deaths)
3. Identify mathematical/statistical models of transmission
4. Collate risk factors for infection, severe disease, and death
5. Summarize seroprevalence data
6. Support infectious disease modelling and outbreak response efforts

This information enables effective outbreak preparedness, resource targeting, and mathematical modelling for nowcasting and forecasting of `{pathogen_name}`.

Inclusion Criteria

ALL must be met:

1. Pathogen: Must be about `{pathogen_name}`
2. Language: English only
3. Study type: Peer-reviewed, original research (note systematic reviews/meta-analyses for special consideration)
4. Population: Human subjects (animal studies acceptable if reporting EITHER: (a) transmission parameters: R_0, R_t, R_e, r, growth rate, mutation rate, OR (b) vector parameters: extrinsic incubation period, vector reproduction numbers, vector competence, mosquito delays)
5. Content: Must contain AT LEAST ONE of:
   (a) Quantitative details of concluded/ongoing human outbreak (size, year, location, duration, spatial scale)
   (b) Mathematical or statistical model of disease transmission
   (c) Measures/estimates of transmission parameters: R, R_0, R_t, r, R_e, growth rate, doubling time
   (d) Measures/estimates of timing parameters: generation time, serial interval, incubation period, latent period, infectious period
   (e) Measures/estimates of severity: CFR, IFR, hospitalization rate, mortality rate, attack rate
   (f) Measures/estimates of genetic evolution: mutation rate, substitution rate, evolutionary rate
   (g) Measures of overdispersion or superspreading (k parameter, transmission heterogeneity)
   (h) Seroprevalence data or serological surveys
   (i) Risk factors for infection, severe disease, death, or hospitalization (with statistical measures)
   (j) Measures/estimates of vector parameters: extrinsic incubation period (EIP), mosquito reproduction numbers, vector competence, mosquito delays, or relative transmission contributions (human-to-human vs vector-borne/zoonotic)

Full-text only

6. Data Extraction Requirement: Must contain extractable mathematical models, transmission models, or quantitative parameter estimates (with values or ranges) for disease modeling. This includes: reproduction numbers, transmission rates, incubation periods, case fatality ratios, model structures, intervention effects, or other modeling parameters. Articles without extractable quantitative parameters or models should be excluded.

Title & Abstract Screening Prompt

You are an expert epidemiologist screening abstracts for a systematic review on the target pathogen.

Study Objectives

[See Study Objectives above]

Screening Criteria

The following is an excerpt of 2 sets of criteria. A study is considered included if it meets ALL inclusion criteria. If a study meets ANY exclusion criteria, it should be excluded. Here are the 2 sets of criteria:

Inclusion Criteria

[See Inclusion Criteria 1–5 above]

Exclusion Criteria

Exclude if ANY apply:

1. Pathogen: Not about `{pathogen_name}` (excludes studies on other pathogens)
2. Language: Non-English
3. Publication type: Conference proceedings, abstract-only, posters, correspondence
4. Study design: *In-vitro* studies only (no human or animal component)
5. Study design: Solely animal studies AND animal studies that do not report transmission parameters (R_0, R_t, R_e, r, growth rate, mutation rate)
6. Outbreak type: Accidental laboratory outbreaks (not natural disease transmission)

Abstract (To Screen)

```
Title: {{title}}
Abstract: {{abstract}}
```

Screening Instructions

We now assess whether the paper should be included in the systematic review by evaluating it against each and every predefined inclusion and exclusion criterion. First, we will reflect on how we will decide whether a paper should be included or excluded. Then, we will think step by step for each criterion, giving reasons for why they are met or not met.

Studies that may not fully align with the primary focus of our inclusion criteria but provide data or insights potentially relevant to our review deserve thoughtful consideration. Given the nature of abstracts as concise summaries of comprehensive research, some degree of interpretation is necessary.

Our aim should be to inclusively screen abstracts, ensuring broad coverage of pertinent studies while filtering out those that are clearly irrelevant.

We will conclude by outputting (on the very last line) `<decision>EXCLUDE</decision>` if the paper warrants exclusion, or `<decision>INCLUDE</decision>` if inclusion is advised or uncertainty persists.

Finally, the articles that pass the abstract screening have their full text screened as follows.

Full-Text Screening Prompt

You are an expert epidemiologist screening abstracts for a systematic review on the target pathogen.

Study Objectives

[See Study Objectives above]

Screening Criteria

The following is an excerpt of 2 sets of criteria. A study is considered included if it meets ALL inclusion criteria. If a study meets ANY exclusion criteria, it should be excluded. Here are the 2 sets of criteria:

Inclusion Criteria

[See Inclusion Criteria 1–6 above, including full-text criterion]

Exclusion Criteria
Exclude if ANY apply:

1. Not about `{pathogen_name}` (excludes other pathogens)
2. Non-English language
3. Conference proceedings, abstract-only, posters, correspondence, Literature reviews, meta-analyses
4. *In-vitro* studies only (no human or animal component)
5. Animal studies without transmission parameters (R_0, R_t, R_e, r, growth rate, mutation rate) or solely animal studies.
6. Case studies/reports with <10 human cases
7. Accidental laboratory outbreaks

Full-Text Article (To Screen)

```
Title:  {{title}}
Full Text:  {{fulltext}}
```

Screening Instructions

We now assess whether the paper should be included in the systematic review by evaluating it against each and every predefined inclusion and exclusion criterion. First, we will reflect on how we will decide whether a paper should be included or excluded. Then, we will think step by step for each criterion, giving reasons for why they are met or not met.

Critically evaluate: Does this paper contain extractable quantitative data, models, or parameters relevant to disease transmission and outbreak response? This is essential for inclusion.

We will conclude by outputting (on the very last line) `<decision>EXCLUDE</decision>` if the paper warrants exclusion, or `<decision>INCLUDE</decision>` if inclusion is advised or uncertainty persists.

E. Agentic Data Extraction Process

After screening, the finalised pool of relevant articles underwent rigorous data extraction. This extraction stage employs a structured tool-calling framework to extract three categories of data: epidemiological parameters, transmission models and outbreak data from full-text articles. Each category followed a multi-stage workflow with validation on each tool output.

E.1. Parameters

Valid Epidemiological Parameters for Extraction Epidemiological parameters are quantitative summaries of how an infection behaves in a population, such as its rate of spread, the delays between key stages of infection, the infection and fatality rates, and risk factors across demographic groups. We used PERG's data entry tool, a REDCap survey, as the reference list of epidemiological quantities that human reviewers would extract from the literature.[9] This gave a fixed catalogue of 47 *parameter types* that cover mutation processes, transmission intensity, delay distributions in humans and mosquitoes, severity, seroprevalence, and risk factors. These higher-order groupings are labelled *parameter classes*, and AgentSLR defines data extraction criteria at the parameter class-level. Table 9 lists all parameter types targeted by our pipeline, together with brief definitions that match the guidance given to human experts.

Table 9. **Valid parameters for extraction, according to PERG's process.**

Parameter type	Parameter class	Description
Attack rate	Attack rate	Proportion of a population that becomes infected during an outbreak.
Secondary attack rate	Attack rate	Proportion of contacts of a primary case who become infected.
Doubling time	Doubling time	Time required for the number of infections to double.
Growth rate	Growth rate	Exponential rate at which new infections increase over time.
Evolutionary rate	Mutations	Rate of genetic change in a population over time, typically substitutions per site per year.
Mutation rate	Mutations	Frequency at which new genetic mutations arise per site per replication cycle.
Substitution rate	Mutations	Speed at which mutations become fixed in a population's genome.
Generation time	Human delay	Average interval between infection in a case and infection in a secondary case.
Serial interval	Human delay	Time between symptom onset in a primary and secondary case.
Latent period	Human delay	Time from infection to becoming infectious.
Incubation period	Human delay	Time from infection to symptom onset.
Infectious period	Human delay	Duration during which an infected person can transmit the pathogen.
Time in care	Human delay	Average duration of hospitalisation or clinical care.
Symptom onset → admission to care	Human delay	Time from symptom onset to hospital or clinical admission.
Symptom onset → discharge / recovery	Human delay	Time from symptom onset to recovery or discharge.
Symptom onset → death	Human delay	Time from symptom onset to death.
Admission → discharge / recovery	Human delay	Time from hospital admission to recovery or discharge.
Admission → death	Human delay	Time from hospital admission to death.
Other human delay	Human delay	Other reported delays related to human infection or response.
Overdispersion	Overdispersion	Measure of variation in the distribution of individual infectiousness.
Human-to-human	Relative contribution	Proportion of total transmission attributable to human-to-human spread.

[9] https://redcap.imperial.ac.uk/surveys/?s=CEX3YKW8W47NMFA4

Zoonotic-to-human	Relative contribution	Proportion of total transmission from animal or vector sources to humans.
Basic (R0)	Reproduction number	Average number of secondary cases from one case in a fully susceptible population.
Effective (Re)	Reproduction number	Average number of secondary cases in a population with partial immunity or interventions.
Case fatality rate (CFR)	Severity	Proportion of diagnosed cases that result in death.
Infection fatality rate (IFR)	Severity	Proportion of all infections (symptomatic and asymptomatic) that result in death.
Proportion of symptomatic cases	Severity	Proportion of infections that develop symptoms.
IgM	Seroprevalence	Proportion of individuals with detectable IgM antibodies, indicating recent infection.
IgG	Seroprevalence	Proportion of individuals with IgG antibodies, indicating past infection or immunity.
PRNT	Seroprevalence	Proportion with neutralising antibodies detected by plaque reduction neutralization test.
HAI/HI	Seroprevalence	Proportion with antibodies detected by hemagglutination inhibition assay.
IFA	Seroprevalence	Proportion with antibodies detected by immunofluorescence assay.
Unspecified	Seroprevalence	Seroprevalence reported without specifying assay type.
Risk factors	Risk factors	Host, environmental, or behavioural characteristics associated with infection risk.

Multi-Stage Parameter Extraction Pipeline Parameter extraction utilises a five-step workflow that mirrors how a careful human reader would process scientific articles. Starting from full-text contents, we identify relevant estimates in the text, extract them into a standardised format, and collect relevant metadata about population context and parameter uncertainty.

For our first step, we ask a reasoning language model with tool calling (in our implementation, `gpt-oss-120b`) to "screen" each article for each parameter class. The reasoning model is provided with a tool to extract (potentially discontiguous) quotations from the source text that relate to the parameter class. We provide specific details for each parameter class as displayed in Table 10, which are copied quotations from the parameter extraction documentation from the `priority-pathogens` codebase (Nash et al., 2026), accessible at https://github.com/mrc-ide/priority-pathogens/wiki/Parameter-Data.

Table 10. **Screening details for each parameter class.** This is inputted into the "Parameter Class Screening Details" section of the **Parameter Screening Prompt** below.

Parameter Class	Screening Details
Attack rate	The attack rate is the proportion of an at-risk population contracting the disease during a specified time interval. It is often reported as a percentage or rate, e.g. 52 people per 10,000 people.
Growth rate	The epidemic growth rate is a key metric that reflects how quickly the number of infections is changing day by day in a population. It is a time-dependent measure, usually expressed as a percentage or a rate per unit of time (e.g. per day), and is crucial for monitoring the speed and trajectory of an outbreak.
Human delay	These parameters all refer to time intervals in the natural history of infection of the host.
Mutation rate	Mutation rates, like substitution rate or evolutionary rate, describe the speed at which genetic changes accumulate in a population.
Relative contribution	This parameter is intended for pathogens (e.g. MERS) where there is both human to human (h2h) and animal to human (a2h) transmission, and aims to capture the relative magnitude of these two routes of infections in humans. We expect these to be proportions or percentages. E.g. a study might estimate 60% of infections in humans to be from h2h infection.
Reproduction number	We are extracting either the basic reproduction number R0 or the effective reproduction number Re.
Risk factors	We are extracting general information about risk factors in the included papers. We are extracting both univariate (naive) and multivariate (adjusted) risk factors, even if they are both available.

Seroprevalence	These parameters refer to estimations of seroprevalence in the paper. This may also be referred to as antibody prevalence. These parameters will all be expressed in a proportion or percentage of the population.
Severity	Severity refers to either the case fatality ratio or the infection fatality ratio. The case fatality ratio is the proportion of cases who end up dying of the disease. Note this depends on the case definition used, as the denominator is people identified as "cases". The infection fatality ratio is the proportion of infections who end up dying of the disease.

The model is also provided with the study objectives from Appendix D and instructed to only extract parameters "estimated from or fitted to actual data". If no relevant information is found, the model is told to refrain from calling the tool. The full prompt for this step is templatised as follows:

Parameter Screening Prompt

You are an expert epidemiologist extracting epidemiological parameters from scientific articles. You will be provided with the processed text of a scientific article. Your task is to extract information about epidemiological parameters according to the provided schema.

Study Objectives

See study objectives in Appendix D.

Summary Extraction Task Definition

For your first task, you will be provided with the full text of a scientific article and a specific type of parameter. We are only extracting parameters that are estimated from or fitted to actual data. For transmission models, if it is only a theoretical model and they have just chosen parameters from other studies/randomly, then please don't extract these.

Your task is to scan the provided text and determine whether this article estimates any parameters of the provided type. If it does, you must use the provided tool to extract relevant summaries from the text about this parameter. If the article makes no mention of the parameter, simply do not call the tool.

If there are multiple pieces of information about the same parameter, return them as separate list items. You will need to call the tool multiple times if there are multiple separate parameter estimates of the provided type.

In future steps, we will be using the provided summaries to extract structured information about the parameter, including:

(a) The estimated value
(b) Uncertainty intervals
(c) Sample study population

Please make sure your summaries provide all of this information if it is provided. Please be thorough: err on the side of extracting more information rather than less.

Parameter Class Screening Details

See the details provided for each parameter class in Table 10.

Full Text

```
Title: {{title}}
Full Text: {{fulltext}}
```

Our next steps are executed for each value of the `summaries` array returned by the model's tool call. We prompt the model in a new context, omitting the full text, to focus the model on the relevant text snippets from `summaries` and to save both inference time and API cost. If no relevant parameters are identified for a given article, `summaries` will be empty, and the extraction process will terminate.

Otherwise, we move to our second step, value extraction. At this step, the model utilises the `value_info` of the parameter to extract structured information about its value and uncertainty bounds. As before, we provide instructions for using the tool for each parameter class. These are listed below:

> **Value Extraction Details for Attack rate**
>
> If the attack rate is reported as a percentage, extract the percentage in the `value` field and set `unit` to `percentage`. If the attack rate is reported as a rate, extract the numerator in the `value` field and set `rate_denominator` to the denominator of the rate. Please extract attack rates as written in the paper.

> **Value Extraction Details for Growth rate**
>
> Please extract growth rates from the paper. Populate the `value` field with a numerical value as it is specified in the paper. If the paper provides a percentage value like 33%, record this value as `0.33`. Populate the `unit` field with one of the provided units according to the tool schema.

> **Value Extraction Details for Human delay**
>
> **Delay type**
>
> The `delay_type` field records the specific type of time interval. It can take one of the following values:
>
> - `generation_time`: The generation time is the time interval between infector exposure to infection and infectee exposure to infection. It may be used in reproduction number estimation, but given the difficulties in its observation, it may be replaced by the serial interval (see below).
> - `serial_interval`: The serial interval is the time interval between infector symptom onset and infectee symptom onset. It is frequently used in reproduction number estimation, as a substitute for the generation time.
> - `latent_period`: The latent period is the time interval between exposure to infection and becoming infectious. It is sometimes used interchangeably with the incubation period (see below). It may also be referred to as the latency period or the pre-infectious period.
> - `incubation_period`: The incubation period is the time interval between exposure to infection and symptom onset. It often coincides with the latent period, but may be shorter (symptom onset before infectiousness, e.g. SARS) or longer (infectiousness before symptom onset, e.g. Covid-19). It may also be referred to as the intrinsic incubation period (in the context of vector-borne diseases) or a subclinical infection.
> - `infectious_period`: The infectious period is the time interval during which the host remains infectious. It directly follows the latent period (see above). It may also be referred to as the infective period, the contagious period, the transmission period or the communicability period.
> - `time_in_care`: The time in care is the time interval between admission to care and discharge from care or death. Unless there is a delay in receiving care, it directly follows the time from symptom to careseeking. It may vary according to health outcome and is typically highly skewed. It may also be referred to as the length of stay (LOS).
>
> Human delays other than the six listed above may also be reported, for example the time from symptom onset to recovery, symptom onset to death, time from seeking care to admission to care etc. We allow `delay_type` to take on one of these other time interval values:
>
> - `admission__to__death`
> - `admission__to__discharge_or_recovery`
> - `symptom_onset__to__admission`
> - `symptom_onset__to__death`
> - `symptom_onset__to__discharge_or_recovery`
>
> In the case that *none* of the above values apply to a human delay parameter you have found, set `delay_type = 'other'` and record the type of delay in the `delay_type_note` field.
>
> **Value and unit**
>
> Use the `value` and `unit` fields to record the parameter estimate (e.g. x hours, days, weeks, or other).

> **Value Extraction Details for Mutation rate**
>
> For this task, we extract parameters estimated from pathogen genetic sequences. If no parameters were derived from genetic sequences, then this section can be skipped *even if sequencing was performed and reported*.
>
> `substitution_rate`, `evolutionary_rate`, and `mutation_rate` are different `parameter_type` values for describing the speed at which genetic changes accumulate in a population. When selecting the `parameter_type`, choose the value type and units based on the wording used by the authors in the article. If there are multiple terms used for the same measure (e.g.

substitution rate is used in the text, evolutionary rate is used in the table), choose either the most frequently used term or default to substitution_rate (if the units are substitutions per site per year). These values are often in the supplemental material. So if genetic sequences or phylogenetic analyses are presented, check the supplement. We are not extracting parameters associated with selection pressure or synonymous/nonsynonymous mutations, unless based on data or methodological limitations they have only been able to calculate substitution rate from nonsynonymous mutations (in that case specify this in the 'Gene' field, similar to *in vitro* experiments - see next bullet point). If substitution rates are calculated for subgroups (e.g. 'clades,' 'strains,' 'branches', etc), report the global estimate and indicate disaggregated data is available in the Parameter Disaggregation section.

As always, the unit value is very important for these parameters. The most common unit is substitutions_per_site_per_year. If units are not clear or they do not match the available options in the drop-down menu, set to unspecified.

Fill the genome_site field with the portion of the pathogen's genome used to estimate any extracted parameters (e.g. reproduction number, growth rate, substitution rate). This can be a gene, a gene segment, a codon position, or a more generic description (e.g. 'whole genome' or 'intergenic positions'). If parameter values are independently estimated for different portions of the genome, please enter each on a separate parameter value form. If a mutation rate is estimated by *in vitro* experiments of recombinant variants (for example, measuring the rate of mutation in an inserted gene, such as green fluorescent protein [GFP]), enter the name of the inserted gene used, even though this gene might not be naturally occurring in the virus's genome. In addition, they may measure different types of mutations (SNPs vs indels) during *in vitro* experiments. If this is the case, enter the type of mutation used to calculate the rate (ex. GFP-SNP, to signify that SNP mutations in the GFP gene were used to calculate the mutation rate).

Value Extraction Details for Severity

- parameter_type – we extract case fatality ratios (CFR), infection fatality ratios (IFR), and the proportion of cases that are symptomatic and asymptomatic.
  - Case fatality ratio (CFR) – the proportion of cases who end up dying of the disease. Note this depends on the case definition used, as the denominator is people identified as "cases". All CFRs should be extracted, even when a subset of the population is selected (e.g. severe cases); make sure to describe the population denominator in the context and notes.
  - Infection fatality ratio (IFR) – the proportion of infections who end up dying of the disease (harder to calculate but less context dependent).
  - Symptomatic proportion of infections – the proportion of total infections that are symptomatic.
  - Asymptomatic proportion of infections – the proportion of total infections that are asymptomatic.
- Parameter value – we don't do any calculation ourselves i.e. if a paper quotes number of deaths and number of cases, but not a CFR, we don't calculate the CFR.
- Ratio/prevalence values – please extract the numerator and denominator that generate the severity ratio. In line with the rule of 3, only extract the numerator and denominator of the central CFR value, even if disaggregated numerators and denominators are available. If there is no central value, do not extract any numerator or denominator. If the numerator and denominator are presented, but the percentage severity is not, extract the numerator, denominator and context, but leave the central value blank.
- method – we extract information about the method used to calculate CFR (or IFR), mainly whether it is:
  - a "naive" method, i.e. percentage mortality which computes total deaths divided by total cases (or infections); this is wrong because there may be many cases or infections who do not have final status information, so the naive estimate is typically an underestimate of true CFR (or IFR).
  - an adjusted method, which somehow accounts for infections or cases with unknown final status (e.g. calculates deaths / (deaths + recoveries) or does something more fancy).
  - an unknown method.
- value_type: mean, median, shape, etc. Please note that it may be the case that multiple measures of central tendency are provided, especially when the entire distribution of a parameter is presented. To avoid extracting multiple measures of centrality for the same parameter and to avoid bias, only one parameter value_type can be extracted. Central parameter types are prioritised based on the available uncertainty types in the following way:
  - When SD/variance/CIs are available: extract mean.
  - Else when only IQR/CrIs are available: extract median.
  - If mode is presented, this should be prioritised *after* the mean or median.
  - If Weibull distribution parameters are presented: prioritise extraction of the shape rather than mean/CIs or median/CrIs. We can get mean/CIs from shape/scale analytically but can only get shape/scale from mean/CIs numerically.

- `statistical_approach` – if the central parameter estimates are summarised directly from empirical data, select `observed_sample_statistic`. If the central parameter is estimated using a transmission model, select `estimated_model_parameter`. Due to limited data sources, the Oropouche systematic review *only* was extended to include `case_study` data.

The full prompt for the value extraction step is templatised below, incorporating text from both the parameter class screening details and the value extraction details.

Value Extraction Prompt

You are an expert epidemiologist extracting epidemiological parameters from scientific articles. You will be provided with the processed text of a scientific article. Your task is to extract information about epidemiological parameters according to the provided schema.

Study Objectives

See study objectives in Appendix D.

Value Extraction Task Definition

Value extraction task

For your next task, you will be provided with excerpts from a scientific article and a specific type of parameter. We are only extracting parameters that are estimated from or fitted to actual data. For transmission models, if it is only a theoretical model and they have just chosen parameters from other studies/randomly, then please don't extract these.

Scan the provided text and for the requested parameter and return all estimated parameter values using the provided tool. You will need to call the tool multiple times if there are multiple separate estimates.

Parameter Class Screening Details

{{`parameter_class`}}: parameter value extraction

{{*Screening details from Table 10*}}

Value Extraction Details for {`parameter_class`}

See the specific details of value extraction above.

Value Excerpts

The following are excerpts from the scientific article about parameter value:

{{`value_info`}}

The tool provided to the language model is distinct per parameter class. In Table 11, we specify the schemas utilised for these tool calls.

Table 11. **Schemas used for value extraction tool calls for each parameter class.** Here "–" means that any values of the correct type are allowed.

Parameter class	Variable	Type	Allowed values	Description
Attack rate	value	Float	–	The value of the attack rate.
	unit	Enum	percentage; rate	The unit of the provided attack rate.
	type	Enum	primary; secondary	Whether primary or secondary attack rate.
	rate denominator	Integer; Null	–	The denominator of the value, if the parameter is provided as a rate.
Doubling time	value	Float	–	The value of the doubling time, in days.
Growth rate	value	Float	–	The value of the growth rate.

	unit	Enum	per hour; per day; per week; per month; per year; other; unspecified	The unit of the provided growth rate.
Human delay	value	Float	–	The value of the human delay parameter.
	delay type	Enum	admission to death; admission to discharge or recovery; generation time; incubation period; infectious period; serial interval; symptom onset to admission; symptom onset to death; symptom onset to discharge or recovery; time in care; other	The specific delay parameter reported.
Mutation rate	value	Float	–	The value of the mutation rate parameter.
	type	Enum	evolutionary rate; mutation rate; substitution rate	The specific mutation rate parameter reported.
	unit	Enum	substitutions per site per year; mutations per genome per generation; percentage; other; unspecified	The unit of the mutation rate parameter value.
	genome site	String	–	The specific genome site or region associated with the mutation rate value.
Overdispersion	value	Float	–	The value of the overdispersion parameter
	unit	Enum	no units; max number of cases superspreading	The unit of the overdispersion parameter
Relative contribution	value	Float	–	The value of the relative contribution parameter.
	type	Enum	human-to-human; zoonotic-to-human	The type of relative contribution reported.
Reproduction number	value	Float	–	The value of the reproduction number parameter.
	type	Enum	basic R0; effective Re	The type of reproduction number reported.
	transmission	Enum	human; mosquito; unspecified; other	The type of transmission for this reproduction number estimate.
	method	Enum	branching process; growth rate; compartmental model; next generation matrix; empirical; genomic; other	The method used to obtain the reproduction number estimate.
Risk factors	name	List[Enum]	age; close contact; breastfeeding; comorbidity; contact with animal; environmental; funeral; hospitalisation; household contact; humidity; non-household contact; occupation; prior immunity to arboviruses; rainfall; sex; social gathering; temperature; other	The name of the risk factor.
	outcome	List[Enum]	death in general population; Guillain Barre Syndrome; infection; low birthweight; microcephaly; miscarriage or stillbirth; other neurological symptoms in general population; premature birth; serology; severe disease in general population; spillover risk; recovery; Zika congenital syndrome or other birth defects; other	The outcome for which the risk factor was evaluated.

	occupation	List[Enum]	abattoir services; correctional facilities; education; funeral and burial services; healthcare; laboratory; livestock and animal herders; public transport; quarantine facilities; veterinary; other; unspecified	If `name` is set to 'occupation', the occupation(s) that correspond(s) most closely to that described in the paper.
	significant	Enum	significant; not significant; unspecified	Whether the risk factor is significant or not.
	adjusted	Enum	adjusted; not adjusted; unspecified	Whether the estimates of the risk factors are adjusted or unadjusted.
Seroprevalence	value	Float	–	The seroprevalence value as a proportion between 0.0 and 1.0.
	parameter type	Enum	IgG; IgM; PRNT; HAI; IFA; unspecified	The type of seroprevalence parameter.
	numerator	Integer; Null	The numerator used to calculate the seroprevalence value. If not provided, set to `Null`.	
	denominator	Integer; Null	The denominator used to calculate the seroprevalence value. If not provided, set to `Null`.	
Severity	value	Float	–	The value of the severity parameter as a proportion between 0.0 and 1.0.
	numerator	Integer; Null	–	The numerator of the CFR or IFR parameter, if provided.
	denominator	Integer; Null	–	The denominator of the CFR or IFR parameter, if provided.
	parameter type	Enum	CFR; IFR; proportion of symptomatic cases; proportion of asymptomatic cases	The type of severity parameter reported.
	method	Enum; Null	naive; adjusted; unknown	The method used to calculate the CFR or IFR.

Following value extraction, all parameters move to our third step: population context extraction. We extract population context with the same prompt and tool for all parameter classes (see below).

Population Extraction Prompt

You are an expert epidemiologist extracting epidemiological parameters from scientific articles. You will be provided with the processed text of a scientific article. Your task is to extract information about epidemiological parameters according to the provided schema.

Study Objectives

See study objectives in Appendix D.

Population Extraction Task Definition

For your next task, you will be provided with excerpts from a scientific article and an estimated parameter that has been extracted from that article. Your task is to scan the provided text and extract relevant sample population information for the given parameter. You will use the provided tool, which sets the schema you should follow when returning population information.

Population Excerpts

The following are excerpts from the scientific article about parameter population context:

`{{population_info}}`

The population tool call is schematised as follows:

Table 12. **The schema for the population context extraction tool call.**

Variable	Type	Allowed values	Description
population sex	Enum	female; male; both; unspecified	The sex composition of your study population. If you have 99 men and 1 woman you would still put both in this option.
population sample type	Enum	community based; hospital based; household based; housing estate based; population based; school based; travel based; trade or business based; contact based; mixed settings; other; unspecified	How was the study conducted?
population group	Enum	healthcare workers; farmers; outdoor workers; animal workers; butchers; abattoir workers; pregnant women; children; sex workers; people who inject drugs; household contacts of survivors; persons under investigation; general population; persons with symptoms; mixed settings; unspecified; other	Demographic i.e. who was sampled?
population sample size	Integer; Null	–	Number of participants/samples tested etc.
population age min	Integer; Null	–	These must be number fields. If your sample is people over 18 you would put age min = 18 and leave age max blank.
population age max	Integer; Null	–	These must be number fields. If your sample is people over 18 you would put age min = 18 and leave age max blank.
population countries	List[String]	–	Where was the study undertaken?
population location	String	–	Location reported i.e. Kerry Town Ebola Treatment Centre.
method moment value	Enum	start outbreak; mid outbreak; end outbreak; post outbreak; endemic; unspecified	When in the outbreak was this study undertaken?

For our final step, if we have multiple extractions of the same class for an article, we ask the language model to aggregate parameters that should be reported as ranges over population disaggregations. Our aggregation logic follows PERG's *rule of three*, which specifies certain pathogen-specific conditions for when aggregated reporting is appropriate. These are detailed to the language model in the instruction prompt below, which is provided identically for all parameter classes.

Aggregation Prompt

You are an expert epidemiologist extracting epidemiological parameters from scientific articles. You will be provided with the processed text of a scientific article. Your task is to extract information about epidemiological parameters according to the provided schema.

Study Objectives

See study objectives in Appendix D.

Aggregation Task Definition

Aggregation task
For your next task, you will be provided with a list of parameters already extracted from an epidemiological study. Your task is to provide *aggregations* of these parameter values when suitable.

Aggregation context
Some epidemiological papers have a huge level of parameter disaggregation (e.g. age, sex, location) and so we have established different rules to ease our meta-review process. For non-location-related disaggregations, please remember the **rule of three**. If there are three or more disaggregations for a parameter, e.g. Rt values for three or more age groups, extract these as a **range** and specify that disaggregated data is available and what the parameter is disaggregated by.
Each pathogen has different rules on location, which we state here:

- marburg; ebola; MERS: Location is included within the rule of three.
- lassa; SARS; zika; nipah: Please *do not aggregate* values if the disaggregation is by location as much as possible and do not apply the rule of three for geographic regions down to admin level 2 (sub-regions) of a country. However, please respect the rule of three for estimates by neighborhood for example.

If the provided parameters do not contain adequate population information to perform an aggregation, then do not return any aggregated values.
If you decide that an aggregation is necessary, use the provided tool to submit aggregated values according to the tool's schema. Provide the `lower_bound` and `upper_bound` of the parameter values, and list the types of population disaggregation (like "age", "sex", etc.) in the `disaggregated_by` field. Fill the `aggregated_ids` list with all of the `ids` from the parameters you aggregated.

Extracted parameters

Extracted parameters: `{{parameters}}`

E.2. Models

Valid transmission models for extraction Epidemiological transmission models are mathematical frameworks that simulate how infectious diseases spread through populations by mechanistically describing the interactions between infected and susceptible individuals. We extract models that mechanistically represent disease transmission dynamics, excluding purely statistical analyses, regression-based forecasting without transmission mechanisms, and risk factor studies. Table 13 defines the categories of model characteristics extracted in this study, organised into structural properties, epidemiological features, assumptions, intervention categories, and reproducibility indicators.

Table 13. **Model characteristic categories targeted by the extraction pipeline.**

Category	Description
Structural Properties	Model type (compartmental, agent-based, branching process) and compartmental architecture (SIS, SIR, SEIR, etc.). Whether the model is stochastic or deterministic.
Epidemiological Features	Primary transmission routes (airborne, direct contact, vector-borne, sexual). Spatial heterogeneity and spillover dynamics from animal reservoirs.
Behavioural Assumptions	Mixing patterns (homogeneous or heterogeneous), age-dependent susceptibility, cross-immunity between pathogens, and temporal variation in transmission rates.
Theoretical vs. Fitted	Whether the model was fitted to actual data or uses parameters from literature or arbitrary values.
Intervention Categories	Control measures evaluated including vaccination, quarantine, vector control, treatment, contact tracing, behaviour changes, and various vector management strategies.
Reproducibility Indicators	Code availability, programming language used, data sharing status, and presence of documentation (README).

Model extraction schema Table 14 defines the complete extraction schema with field specifications, data types, allowed values, and descriptions. The schema uses controlled vocabularies to ensure consistency and enable structured analysis of modelling approaches across the literature.

Table 14. **Model extraction schema with field specifications, data types, allowed values, and descriptions.**

Field Name	Type	Allowed Values	Description
model_type	Enum	Compartmental; Branching process; Agent/Individual based; Other; Unspecified	Primary modeling framework used for transmission dynamics.
compartmental_type	Enum	SIS; SIR; SEIR; SEIR-SEI; SAIR-SEI; Not compartmental; Other compartmental	Specific compartmental model architecture if applicable. Use "Not compartmental" for non-compartmental models.
stoch_deter	Enum; Null	Stochastic; Deterministic	Whether the model is stochastic or deterministic. Only extract if explicitly stated. Null if not specified.
transmission_route	List[Enum]	Airborne or close contact; Human to human (direct contact); Human to human (direct non-sexual contact); Vector/Animal to human; Sexual; Unspecified	Primary pathway(s) through which pathogen transmission occurs. Multiple routes can be selected.
uncertainty_was_considered	Boolean; Null	True; False	Whether uncertainty was considered through stochasticity, multiple models, or parameter variation (e.g. sensitivity analyses, Bayesian analysis). Null if not specified.

spatial_model	Boolean; Null	True; False	Whether the model explicitly incorporated spatial or geographic heterogeneity. Null if not specified.
spillover_included	Boolean; Null	True; False	Whether the model explicitly modelled spillover (e.g. animal reservoir component, contribution to force of infection from zoonosis). Null if not specified.
assumptions	List[Enum]	Homogeneous mixing; Latent period is same as incubation period; Heterogeneity in transmission rates (between human groups; between groups; between human and vector; over time); Age dependent susceptibility; Cross-immunity between Zika and dengue; Other; Unspecified	Key structural and behavioural assumptions. Should be explicitly described in the paper or clear from model equations. Multiple assumptions can be selected.
theoretical_model	Boolean	True; False	Whether the model was NOT fitted to data (parameters from literature or arbitrary values). True if not fitted; False if fitted to actual data.
interventions_type	List[Enum]	Vaccination; Quarantine; Vector/Animal control; Treatment; Contact tracing; Hospitals; Treatment centres; Safe burials; Behaviour changes; Wolbachia replacement/suppression; Genetically modified mosquitoes; Mechanical removal of breeding sites; Pesticides/larvicides; Insecticide-treated nets; Indoor residual spraying; Other; Unspecified	Categories of control measures evaluated or incorporated in the model(s). Multiple interventions can be selected.
code_available	Boolean	True; False	Whether model implementation code was made publicly available.
coding_language	Enum; Null	R; Python; Matlab; Julia; C++; Other	Programming language(s) used for model implementation if code is available. Null if not specified.
is_data_used_available	Enum; Null	Yes (as an attachment; with a DOI; on Github; on another platform); Not available; Unspecified	Whether input data used for the model was shared and how it was shared. Null if not specified.
readme_included	Boolean; Null	True; False	Whether a README or detailed documentation accompanied the code repository. Null if not applicable.
notes	String; Null	–	Additional context or notes about the extracted model. Free text field.

Multi-stage model extraction pipeline Model extraction employs a two-stage agentic workflow operating on full-text article content. Unlike parameter extraction, which requires fine-grained text excerpting and value parsing, model extraction focuses on identifying the presence of dynamic transmission models and characterising their structural properties using controlled vocabularies.

In the first stage, a binary screening step identifies articles containing dynamic transmission models while excluding purely statistical analyses, regression-based forecasting, and risk factor studies without transmission dynamics (see the "Model Screening Prompt" below). The language model returns a simple "True" or "False" response indicating whether the article contains models suitable for extraction.

For articles passing this screen, the extraction stage deploys a structured tool-calling approach where the language model iteratively invokes an `extract_model_data` function once per distinct model identified in the article (see the "Model Extraction Prompt" below). Each tool call populates the standardised schema defined in Table 14.

The schema enforces controlled vocabularies for all fields through strict JSON validation. Multiple-select fields (`transmission_route`, `assumptions`, `interventions_type`) accept arrays of values from predefined enumerations, while single-select fields enforce unique values or null for optional characteristics. Validation logic rejects outputs violating vocabulary constraints or logical rules. For example, a non-compartmental `model_type` must have `compartmental_type` set to "Not compartmental", this prompts the model to correct errors before proceeding.

The complete extraction workflow is coordinated by the `ModelExtractionRunner` class, which loads full-text data, applies screening decisions, manages iterative tool calls with validation feedback, and logs all outputs to structured CSV files.

Model Screening Prompt

You are an epidemiologist specializing in infectious disease modeling. Determine if this article contains dynamic transmission models for infectious disease.

Screening Task Definition

Include (respond "True"):

- Compartmental models (SIR, SEIR, etc.)
- Agent-based or individual-based models
- Branching process models
- Network transmission models

Exclude (respond "False"):

- Pure statistical/regression analyses
- Time series forecasting without mechanistic transmission
- Risk factor analyses without transmission dynamics
- Seroprevalence studies without modeling

Respond with only "True" or "False".

Full Text

```
Title: {{title}}
Full Text: {{fulltext}}
```

Model Extraction Prompt

You are an epidemiologist specializing in infectious disease modeling. Extract information about transmission models from scientific articles.

Study Objectives

Study Objectives

This systematic review collates transmission models, outbreaks and parameters for {{pathogen}}.

Extraction Task Definition

Model extraction task

Extract **ALL dynamic transmission models** described in the article that were actually used or implemented.

Do **not** extract:

- Models only mentioned as possible alternatives without implementation
- Regression-only analyses
- Purely statistical forecasting

Tool Calling:

- Call `extract_model_data` **once per model** identified in the article
- After extracting all model/s, stop calling the tool (no completion call needed)

Schema Requirements:

- `transmission_route`, `assumptions`, `interventions_type` are **arrays** (multiple-select)
- All other fields are **single values** (single-select)
- Use `null` for optional single-select fields when not stated
- Use `["Unspecified"]` for required arrays when not stated

Full Text

```
Title:  {{title}}
Full Text:  {{fulltext}}
```

The language model uses the `extract_model_data()` tool (provided to it) to populate the schema defined in Table 14. The tool enforces strict JSON validation with controlled vocabularies for all fields, rejecting invalid outputs and prompting corrections. The complete tool specification follows standard OpenAI function calling conventions with enum constraints for single-select fields and array validation for multiple-select fields.

E.3. Outbreaks

Valid outbreak data for extraction Outbreak data capture the epidemiological characteristics of concluded epidemic events, including temporal bounds, geographic scope, transmission sources, case counts stratified by confirmation status, and demographic breakdowns. We extracted outbreak information as stated in articles, without performing additional calculations or inferring missing values.

Following extraction guidelines suggested by PERG,[10] outbreak inclusion criteria varied by pathogen based on reporting completeness and literature volume. For Marburg and Lassa, all reported outbreaks were captured regardless of size. For Zika, only outbreaks with at least 10 confirmed, probable, or suspected cases were extracted, reflecting the assumption that smaller events may not be systematically documented and contribute minimally to population-level immunity estimates. Table 15 defines the outbreak characteristics and their meanings in natural language.

Table 15. **Outbreak field descriptions and meanings.**

Outbreak Characteristic	Description
Outbreak start day	Day of outbreak onset (1–31). Extracted as stated in paper.
Outbreak start month	Month of outbreak onset. Extracted as stated in paper.
Outbreak start year	Year of outbreak onset. Extracted as stated in paper.
Outbreak end day	Day of outbreak conclusion (1–31). Extracted as stated in paper.
Outbreak end month	Month of outbreak conclusion. Extracted as stated in paper.
Outbreak end year	Year of outbreak conclusion. Extracted as stated in paper.
Outbreak duration (months)	Duration of outbreak in months. ONLY extracted if explicitly stated in paper; not calculated.
Outbreak is currently ongoing	Whether outbreak was concluded or ongoing at time of publication.
Outbreak country	Country where outbreak occurred, using WHO standard country names. Refers to reporting country rather than infection source for imported cases.
Outbreak location	Specific geographic location within country (city, district, province) as written in paper. Multiple locations separated by semicolons.
Outbreak location type	Administrative or geographic unit type of outbreak location.
Outbreak source	Known or suspected source of outbreak introduction.
Mode of detection	Primary method(s) used to identify and confirm cases.
Method of case definition	Criteria used to classify cases. Extracted as described in paper.
Pre-outbreak baseline	Baseline disease status in affected area prior to outbreak. Rarely reported.
Cases confirmed	Number of laboratory-confirmed cases (e.g. via molecular testing).
Cases probable	Number of probable cases as defined in paper. Definition may vary across studies.
Cases suspected	Number of suspected cases as defined in paper. Definition may vary across studies.
Cases unspecified	Number of cases where confirmation status was not specified.
Cases asymptomatic	Number of asymptomatic infections as defined in paper.
Cases severe	Number of severe cases or hospitalizations as stated in paper.
Deaths	Number of deaths attributed to outbreak.
Asymptomatic transmission described	Whether article explicitly described or discussed asymptomatic transmission.
Population size (geographical area)	Total population of affected geographic area. Rarely reported.
Type of cases (sex disaggregation)	Case type for which sex disaggregation was reported.
Male cases	Number of cases in males for specified case type.
Proportion male cases	Proportion (0.0–1.0) or percentage (0–100) of cases in males.
Female cases	Number of cases in females for specified case type.
Proportion female cases	Proportion (0.0–1.0) or percentage (0–100) of cases in females.
Notes	Additional context or clarifications about outbreak characteristics.

Outbreak extraction schema Table 16 defines the complete extraction schema with field specifications, data types, allowed values, and descriptions. The schema uses controlled vocabularies to ensure consistency and enable structured analysis of outbreak characteristics across the literature.

[10] https://github.com/mrc-ide/priority-pathogens/wiki/Outbreak-data

Table 16. **Outbreak extraction schema with field specifications, data types, allowed values, and descriptions.**

Field Name	Type	Allowed Values	Description
outbreak_start_day	Integer; Null	1-31	Day of outbreak onset. Null if not provided.
outbreak_start_month	String (Enum); Null	Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec	Month of outbreak onset. Null if not provided.
outbreak_start_year	Integer; Null	Integer year	Year of outbreak onset. Null if not provided.
outbreak_end_day	Integer; Null	1-31	Day of outbreak conclusion. Null if not provided.
outbreak_end_month	String (Enum); Null	Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec	Month of outbreak conclusion. Null if not provided.
outbreak_end_year	Integer; Null	Integer year	Year of outbreak conclusion. Null if not provided.
outbreak_duration_months	Float; Null	Numeric value	Duration in months. ONLY if explicitly stated; not calculated. Null if not stated.
outbreak_is_currently_ongoing	Boolean	True; False	Whether outbreak was concluded (False) or ongoing (True) at publication.
outbreak_country	String (Enum)	WHO standard country names (195 member states)	Country where outbreak occurred. MUST match WHO standard names.
outbreak_location	String; Null	Free text	Specific location within country. Multiple locations separated by semicolons. Null if not provided.
outbreak_location_type	String; Null	Free text (e.g. district, province, county, state, hospital)	Type of administrative or geographic unit. Null if not specified.
outbreak_source	String (Enum); Null	Domestic animal; Wild animal; Date palm sap; Unknown; Other	Known or suspected source of outbreak introduction. Null if not provided.
mode_of_detection	String (Enum); Null	Molecular (PCR etc); Symptoms; Confirmed + Suspected; Unspecified	Primary method used to identify and confirm cases. Null if not provided.
method_of_case_definition	String; Null	Free text	Criteria used to classify cases as described in paper. Null if not provided.
pre_outbreak	String (Enum); Null	Disease-free baseline; Endemic equilibrium; Unspecified; Probable	Baseline disease status prior to outbreak. Null if not provided.
cases_confirmed	Float; Null	Non-negative numeric	Number of laboratory-confirmed cases. Null if not provided.

Continued on next page

Table 16 continued from previous page

Field Name	Type	Allowed Values	Description
cases_probable	Float; Null	Non-negative numeric	Number of probable cases. Null if not provided.
cases_suspected	Float; Null	Non-negative numeric	Number of suspected cases. Null if not provided.
cases_unspecified	Float; Null	Non-negative numeric	Number of cases with unspecified confirmation status. Null if not provided.
cases_asymptomatic	Float; Null	Non-negative numeric	Number of asymptomatic infections. Null if not provided.
cases_severe	Float; Null	Non-negative numeric	Number of severe cases or hospitalizations. Null if not provided.
deaths	Float; Null	Non-negative numeric	Number of deaths attributed to outbreak. Null if not provided.
asymptomatic_transmission_described	Boolean	True; False	Whether article explicitly described or discussed asymptomatic transmission.
population_size_geographical_area	Float; Null	Non-negative numeric	Total population of affected geographic area. Null if not provided.
type_cases_sex_disagg	String (Enum); Null	Confirmed; Suspected; Other; Unspecified	Case type for which sex disaggregation was reported. Null if not provided.
male_cases	Float; Null	Non-negative numeric	Number of male cases for specified case type. Null if not provided.
prop_male_cases	Float; Null	Numeric (0.0-1.0 or 0-100)	Proportion or percentage of cases in males. Null if not provided.
female_cases	Float; Null	Non-negative numeric	Number of female cases for specified case type. Null if not provided.
prop_female_cases	Float; Null	Numeric (0.0-1.0 or 0-100)	Proportion or percentage of cases in females. Null if not provided.
notes	String; Null	Free text	Additional context or clarifications about outbreak characteristics. Null if not needed.

Multi-stage outbreak extraction pipeline Outbreak extraction employs a two-stage workflow operating on full-text article content. The first stage applies binary screening to identify articles reporting concluded, real-world outbreak events with defined case counts, excluding ongoing outbreaks, modelled scenarios, routine surveillance, and single case reports (see the "Outbreak Screening Prompt" below). The language model returns a simple "True" or "False" response indicating whether the article contains outbreaks suitable for extraction.

For articles passing this screen, the extraction stage deploys a structured tool-calling approach where the language model iteratively invokes an `extract_outbreak_data` function once per distinct outbreak identified in the article (see the "Outbreak Extraction Prompt" below). Outbreaks are considered distinct if they differ by location, time period, or both. Each tool call populates the standardised schema defined in Table 16.

The schema enforces controlled vocabularies for categorical fields through strict JSON validation. The required fields must be provided (`outbreak_is_currently_ongoing`, `outbreak_country`, `asymptomatic_transmission_described`), while all other fields accept null values when data are not stated in the article. The `outbreak_country` field enforces WHO standard country names, and the `outbreak_location` field prohibits commas, requiring semicolon separators for multiple locations to avoid parsing ambiguities. Validation logic rejects outputs violating vocabulary constraints or data type rules, prompting the model to correct errors before proceeding.

The complete extraction workflow is coordinated by the `OutbreakExtractionRunner` class, which loads full-text data, applies screening decisions, manages iterative tool calls with validation feedback, and logs all outputs to structured JSONL files for downstream analysis.

Outbreak Screening Prompt

You are an epidemiologist conducting systematic review of infectious disease outbreaks. Determine if this article reports concluded, real-world outbreak events with defined case counts.

Screening Task Definition

Include (respond "True"):

- Discrete outbreak events with ALL of: specific location, defined time period, and case counts
- Outbreak investigations describing a bounded epidemic event
- Case series (2 or more cases) from a specific outbreak

Exclude (respond "False"):

- Ongoing outbreaks at time of publication
- Modeled, simulated, or forecasted outbreaks
- Routine surveillance or annual disease burden (e.g., "X cases per year")
- Seroprevalence or risk factor studies without outbreak context
- Single case reports

Key Question: Can you identify a specific outbreak event (not general disease occurrence) with a start/end period and case count?

Respond with only "True" or "False".

Full Text

```
Title:  {{title}}
Full Text:  {{fulltext}}
```

Extraction Task Definition

Outbreak extraction task
Extract concluded outbreaks with defined case counts from the article. Call `extract_outbreak_data` once for each distinct outbreak (different location, time, or both).

Important Notes:
We are extracting everything as presented in the paper, even if you think it is an error by the author(s).
Extract data EXACTLY as stated in the paper. Do NOT perform calculations, convert units, or infer missing values.
DO NOT use commas in any field. If you need to separate items within a field, please use a semicolon.

Tool Calling Rules:

- Call `extract_outbreak_data` once per distinct outbreak
- Outbreaks are distinct if they differ by location, time, or both
- After extracting all outbreaks, stop calling the tool (no completion call needed)

Schema Requirements: Only three fields are required:

- `outbreak_is_currently_ongoing`: true or false

- `outbreak_country`: Must be valid WHO country name
- `asymptomatic_transmission_described`: true or false

All other fields: Use null when not stated in the paper.

Extraction Rules:

- Only select values that appear in the allowed lists for categorical fields
- Extract dates as separate components (day, month, year)
- Do NOT calculate `outbreak_duration_months`; only extract if explicitly stated
- If you receive a validation error message, correct the tool call and try again

Field-Specific Guidance:
Location:

- `outbreak_country`: MUST match WHO standard names exactly (e.g., "United States of America" not "USA", "Viet Nam" not "Vietnam")
- `outbreak_location`: Extract as written; use semicolons not commas (e.g., "Lagos; Abuja")

Case Counts: Extract all categories as reported

- `cases_confirmed`: Laboratory-confirmed cases
- `cases_probable`: Probable cases (clinical diagnosis)
- `cases_suspected`: Suspected cases under investigation
- `cases_unspecified`: Cases without clear classification
- `cases_asymptomatic`: Asymptomatic cases identified
- `cases_severe`: Severe cases OR hospitalizations (note if hospitalizations in notes)
- `deaths`: Reported deaths

Mode of Detection: Select ONE

- "Molecular (PCR etc)" Laboratory confirmation (PCR, ELISA, culture, etc.)
- "Symptoms": Clinical/syndromic diagnosis only
- "Confirmed + Suspected": Both lab-confirmed and clinical cases
- "Unspecified": Not clearly stated

Sex Disaggregation: When provided, extract:

- `male_cases` / `female_cases`: Counts
- `prop_male_cases` / `prop_female_cases`: Proportion/percentage as reported
- `type_cases_sex_disagg`: Which case type is disaggregated (Confirmed/Suspected/Other/Unspecified)

Pre-Outbreak Baseline:

- "Disease-free baseline": No previous cases
- "Endemic equilibrium": Disease was endemic
- "Probable": Suggested but not definitive
- "Unspecified": Not discussed

Dates: Provide as separate components (day, month, year). Partial dates are acceptable (e.g., only month and year).
Duration: ONLY extract if paper explicitly states duration. Do NOT calculate from dates.
Notes: Use this field for important context, data quality issues, or special circumstances.

Pathogen-Specific Rules:

- Zika, RVF: Only extract outbreaks with 10 or more cases
- Marburg, Lassa, Nipah: Extract all outbreaks
- OROV: Include even single case reports

Outbreak Extraction Prompt

You are an epidemiologist conducting systematic review of infectious disease outbreaks. Extract structured data about concluded outbreak events from scientific articles.

Study Objectives
This systematic review collates transmission models, outbreaks and parameters for {{pathogen}}.

Extraction Task Definition *See Extraction Task Definition details above.*

Full Text

```
Title: {{title}}
Full Text: {{fulltext}}
```

The language model uses the `extract_outbreak_data()` tool (provided to it) to populate the schema defined in Table 16. The tool enforces strict JSON validation with controlled vocabularies for categorical fields, rejecting invalid outputs and prompting corrections. The complete tool specification follows standard OpenAI function calling conventions with enum constraints for single-select fields and null acceptance for optional fields.

Provenance extraction Following successful extraction of parameters, models, and outbreaks, a provenance stage systematically mapped each extracted value to supporting textual excerpts from the article, ensuring complete traceability and grounding of all characteristics in source material. For each extracted record (parameter estimate, model descriptor, or outbreak summary), the provenance extraction invoked a dedicated tool (`extract_parameter_provenance`, `extract_model_provenance`, or `extract_outbreak_provenance`) that received the complete set of previously extracted characteristics and identified verbatim quotes, equation references, or table citations justifying each value selection. For multi-select fields (e.g. transmission routes, assumptions, interventions in models; multiple locations in outbreaks), each selected option required independent textual support. This additional stage enabled potential validation of extraction quality, provided transparency for subsequent data synthesis, and formed an audit trail linking structured outputs to primary literature, with all provenance traces logged to structured files for downstream analysis.

F. Report Generation: Building Systematic Living Reviews

F.1. Deterministic Report Assembly

Given a pathogen p, we generate human-readable reports directly from the extracted, structured datasets for outbreaks. The report build aggregates extraction records into descriptive summary tables and figures, then compiles a Markdown draft, and finally renders a PDF. The report build is lightweight relative to retrieval, screening, and extraction and is omitted from our main runtime breakdown (typically < 5 minutes per pathogen).

Inputs and derived artefacts

Inputs Let $\mathcal{D}_p^{\mathrm{O}}$ be the set of extracted outbreak records for pathogen p (one row per outbreak entity). Each record is schema-validated at extraction time (Appendix E), so report generation treats the datasets as structured inputs.

Content manifest. The manifest stores: pathogen identifier, timestamp, summary statistics (e.g. outbreak counts and geographic coverage), the list of narrative sections, and structured metadata for each figure and table (number, title, caption, path, and row or observation counts). This manifest is later used as part of the evidence packet in the LLM refinement stage (next subsection).

Table 17. **Artefact inventory for outbreak report generation (per pathogen p).** All artefacts are derived from the extracted outbreak dataset $\mathcal{D}_p^{\mathrm{O}}$.

Artefact	Path (relative to repo root)	Purpose
Outbreak report (Markdown)	`writeup/`p`/outbreaks_writeup.md`	Human-readable draft with embedded figures and tables.
Outbreak report (PDF)	`writeup/`p`/outbreaks_writeup.pdf`	Portable rendering for sharing and archiving.
Figures directory	`writeup/`p`/figures/`	Generated plots referenced by Markdown (e.g., temporal distribution, geographic spread, case counts).
Summary tables (embedded)	(in `outbreaks_writeup.md`)	Count and proportion tables computed from $\mathcal{D}_p^{\mathrm{O}}$.
Content manifest	`writeup/`p`/content_manifest.json`	Machine-readable inventory of figures, tables, and dataset statistics.

Evidence packet construction

Evidence packet For pathogen p and outbreak report type O, code constructs an evidence packet

$$E_p^{\mathrm{O}} = \left(\textsc{Stats}_p^{\mathrm{O}},\ \textsc{Figs}_p^{\mathrm{O}},\ \textsc{Tables}_p^{\mathrm{O}},\ W_p^{\mathrm{O},(0)} \right),$$

where $\textsc{Stats}_p^{\mathrm{O}}$ is a concise text summary of dataset counts and geographic breakdowns, $\textsc{Figs}_p^{\mathrm{O}}$ is the required figure list (paths and captions), $\textsc{Tables}_p^{\mathrm{O}}$ is the set of tables to be included (as Markdown blocks), and $W_p^{\mathrm{O},(0)}$ is the programmatic Markdown draft. The model is instructed to rely only on E_p^{O} and not to introduce external facts.

F.2. Evidence grounded narrative refinement

Report writing proceeds by an LLM revision stage that refines $W_p^{\mathrm{O},(0)}$ into a narrative synthesis, while enforcing evidence grounding and artefact presence.

Self-refinement loop and non-negotiable checks

Grounding and asset checks (non-negotiable). Two constraints are enforced for every refined version:

1. **Asset presence:** every required figure path from the manifest must appear at least once as a Markdown image line.
2. **Table preservation:** every table provided in the evidence packet must be present, with values unchanged (reformatting is allowed).

If either constraint is violated, we deterministically append missing figures or tables verbatim at the end of the Markdown so the final PDF always renders with the full artefact set.

Minimal formalisation Let $W^{(0)}$ denote the initial (programmatic) Markdown draft. Each iteration applies:

$$\text{critique}(W^{(k-1)}) \to C^{(k)}, \qquad \text{revise}(W^{(k-1)}, C^{(k)}) \to W^{(k)}.$$

This is only a notation convenience: in practice the evidence packet always accompanies both steps, and the critique output is structured JSON used to drive the next revision.

RUBRIC AND PROMPTS

Rubric We use an 8-dimension rubric, each scored from 1 (poor) to 5 (excellent). The dimensions are the same for both report types, except for the scope constraint.

Shared dimensions

1. `data_fidelity`: descriptive claims match the evidence packet; no invented statistics or outbreak characteristics.
2. `figure_table_presence`: all required figures and tables appear.
3. `traceability`: outside interpretation blocks, claims cite their source as (Figure X), (Table Y), or (Dataset Statistics).
4. `clarity`: consistent terminology, clear writing, minimal ambiguity.
5. `completeness`: covers the major patterns visible in the available figures and tables.
6. `interpretation_blocks`: interpretation is confined to dedicated blocks and labelled as such.
7. `formatting`: valid Markdown and sensible figure layout hints.

Interpretation policy Interpretation is allowed only inside blockquotes beginning with `> AI-Interpretation:`. Outside those blocks, the narrative must remain descriptive and evidence-linked; no new numbers may be introduced.

F.3. Report Generation Prompts

We present the exact prompts used for outbreak report generation and self-refinement, formatted consistently with the model report prompts. All prompts are instantiated programmatically by filling placeholders (e.g. `{EVIDENCE_PACKET}`) at runtime.

OUTBREAK REPORT PROMPTS

Outbreak Report: Initial Synthesis Prompt

You are a senior epidemiologist editing a living outbreak surveillance review. You are revising a first draft prepared by a research assistant who summarized extracted outbreak records.

Method Basis
Do not cite external sources; just follow these behaviors:

- Iterative critique→refine loop (Self-Refine).
- Rubric-based form-filling evaluation mindset (G-Eval).
- Attribution-first revision: every descriptive claim must be attributable to the provided evidence packet (RARR-style editing for attribution).
- Living review principles: explicitly describe what is present in the dataset snapshot and what is missing; avoid academic formatting.

Hard Scope Constraint

Focus on documented outbreak events and outbreak characteristics. Do not broaden into transmission modelling, pathogen biology, or clinical management beyond what is supported by the outbreak dataset.

Truthfulness Constraints

- Do not invent outbreak characteristics, case counts, geographic locations, or external facts.
- Outside of AI-Interpretation blocks, every numeric or categorical claim must be directly supported by the evidence packet and must cite its support as (Figure X), (Table Y), or (Dataset Statistics).
- Interpretation is allowed ONLY inside blockquotes starting with: `> AI-Interpretation:`
- Inside AI-Interpretation blocks, you may propose plausible implications for outbreak surveillance and preparedness, but you must label them as hypotheses and you must not introduce new numbers that are not in the evidence packet.

Figures and Tables Constraints

- All figures must appear as markdown images using their existing paths (e.g., `![Alt](figures/fig1_...png)`). Placement is free.
- Tables must all be present. You may reformat tables, but values must remain identical.

Formatting Agency

- You may include an OPTIONAL HTML comment immediately after any figure image line to suggest sizing for PDF rendering.
- Format: `<!- fig-layout:  width_in=5.5 max_height_in=7.5 ->`
- If absent, defaults will be used.

Output Requirements

- Produce a living outbreak surveillance review in Markdown.
- Use descriptive, report-like sections rather than academic paper structure.
- For each main section, include: (1) Evidence-based description, then (2) one AI-Interpretation blockquote.

Task Definition

Task: Produce Version 1 of the living outbreak surveillance review. Use the evidence packet below. Maintain honesty and verifiability.

Required structure (you may adapt headings, but keep these concepts):

1) Snapshot (dataset size, temporal coverage, geographic scope, what this review represents)
2) Outbreak temporal distribution (outbreak frequency over time, identification of major epidemic periods)
3) Geographic distribution and spread patterns (countries affected, spatial clustering, cross-border transmission)
4) Outbreak size and severity (case counts, fatality rates, outbreak durations)
5) Detection and reporting patterns (modes of detection, case definitions used, reporting delays if mentioned)
6) Demographic patterns (sex disaggregation, age patterns if available)
7) Data quality and gaps (completeness of reporting, missing information, asymptomatic transmission documentation)
8) Evidence-based recommendations (only tied to observed gaps in outbreak surveillance)
9) Change log stub (for future updates)

Evidence Packet

`{EVIDENCE_PACKET}`

Outbreak Report: Critique Prompt

You are a meticulous scientific editor. Return only valid JSON.

Critique Task Definition

You are a scientific editor evaluating a living outbreak surveillance review for faithfulness to the provided evidence packet. Return STRICT JSON only.

Evidence Packet Summary

`{DATASET_STATISTICS}`

Required Figure Paths

`All of the following must appear at least once:`

`{REQUIRED_FIGURE_PATHS}`

Report to Critique

`{CURRENT_REPORT}`

Evaluation Dimensions

Evaluate dimensions (score 1-5). Provide issues and concrete suggestions.

Dimensions:

1) `data_fidelity`: descriptive claims supported by evidence packet; no invented outbreak characteristics, case counts, or geographic information.
2) `outbreak_focus`: stays centered on documented outbreak events and outbreak surveillance rather than transmission modelling or pathogen biology.
3) `figure_table_presence`: all required figures present; all tables present.
4) `traceability`: outside AI-Interpretation blocks, claims cite support as (Figure X)/(Table Y)/(Dataset Statistics).
5) `clarity`: readable, precise, minimal ambiguity, consistent terminology for outbreak characteristics and surveillance metrics.
6) `completeness`: covers major patterns in outbreak temporal distribution, geographic spread, and detection practices described by available figures/tables.
7) `interpretation_blocks`: each main section includes a blockquote starting with `> AI-Interpretation:` and interpretation stays inside it.
8) `formatting`: figure layout directives used sensibly where needed; no broken markdown.

JSON Response Format

`Return JSON of the form:`

```
{
"dimensions": {
"data_fidelity": {"score": 1-5, "issues": [...], "suggestions": [...]},
"outbreak_focus": {"score": 1-5, "issues": [...], "suggestions": [...]},
"figure_table_presence": {"score": 1-5, "issues": [...], "suggestions": [...]},
"traceability": {"score": 1-5, "issues": [...], "suggestions": [...]},
"clarity": {"score": 1-5, "issues": [...], "suggestions": [...]},
"completeness": {"score": 1-5, "issues": [...], "suggestions": [...]},
"interpretation_blocks": {"score": 1-5, "issues": [...], "suggestions": [...]},
"formatting": {"score": 1-5, "issues": [...], "suggestions": [...]}
},
"priority_fixes": [...]
}
```

Outbreak Report: Revision Prompt

You are a senior epidemiologist performing an evidence-grounded revision.

Revision Constraints

- Follow an attribution-first editing approach: outside AI-Interpretation blocks, every claim must be supported by the evidence packet.
- Keep the document outbreak-focused.
- All figures must appear at least once as markdown images with their existing paths.
- All tables must be present; you may reformat, but values must not change.
- Interpretation is permitted only within blockquotes beginning with `> AI-Interpretation:`.
- You may add optional figure sizing directives as HTML comments immediately after image lines: `<!- fig-layout: width_in=5.5 max_height_in=7.5 ->`

Quality Scores

`{DIMENSION_SCORES}`

Priority Fixes

`{PRIORITY_FIXES}`

Evidence Packet

`{EVIDENCE_PACKET}`

Current Report

`{CURRENT_REPORT}`

Revision Requirements

- Fix all critique issues.
- Ensure each main section has (1) evidence-based description with citations (Figure/Table/Dataset Statistics), then (2) `> AI-Interpretation:` block.
- Remove or relabel any statement not supported by the evidence packet.
- Ensure outbreak-only framing (documented outbreak events and surveillance patterns).
- Keep document a living surveillance review (descriptive, update-ready), not an academic paper.

Return the complete revised Markdown.

G. Evaluation Constructs

Following standard perspectives on evaluation design and construct validity in LLM benchmarking (Bean et al., 2025), which emphasise aligning metrics with the underlying phenomenon a benchmark is intended to capture, we build our evaluation around metrics that reflect the specific construct each stage is designed to measure. In particular, our screening metrics target reliable inclusion/exclusion decisions at the article level, while our extraction metrics decompose performance into identifying relevant information, recovering the expected quantity of items, and matching extracted content to reference annotations.

G.1. Article Screening

We evaluate screening as a binary article-level decision $y \in \{\checkmark, \times\}$ ($\checkmark$ = include; $\times$ = exclude) against the PERG reference label. Let TP be articles correctly labelled as $\checkmark$, FP be those incorrectly labelled as $\checkmark$ and FN be those incorrectly labelled as $\times$; then

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \qquad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \qquad F_1 = \frac{2PR}{P+R},$$

where Precision measures the reliability of $\checkmark$ decisions and Recall measures how well we avoid assigning $\times$ to PERG-$\checkmark$ papers. We report macro-F_1 to weight $\checkmark$ and $\times$ performance equally, rather than letting the majority class dominate.

By default, article screening happens in two subsequent stages: first on the abstract and then on the full text. Full-text screening is therefore evaluated with different ablations so we can quantify both the stage-specific and holistic performance. We use three code-defined evaluation configurations for the ablations: (i) **AI abstract→AI full-text**, where any abstract $\times$ forces final $\times$; (ii) **Human abstract→AI full-text** (PERG-conditioned), where any PERG abstract $\times$ forces final $\times$; and (iii) **AI direct full-text**, which evaluates the AI full-text decision without filtering by abstract screening decisions.

G.2. Data Extraction

Schema validation and data quality Prior to evaluation, we validated and filtered our ground-truth extractions to ensure that only properly formatted annotations were compared. For fields typed as `Enums` in the schemas outlined in Appendix E, we defined acceptable values based on the PERG REDCap survey schema, which standardises entries through dropdown lists. Other fields provide a multi-select option—these we handled as `List[Enum]` types in the tool call schemas. We filter any ground-truth extractions where `Enum`-typed values do not agree with the schema, in order to avoid penalising AgentSLR for extractions it is not allowed to produce. Because of schema verification applied in the tool-calling stage, AgentSLR produces no such invalid extractions.

After validation, we aligned articles using shared identifiers, retaining only articles labelled $\checkmark$ by both PERG and AgentSLR. This intersection matches the ground-truth-labelled data to our article pool (Table 1), and thus avoids counting errors due to paper availability from the article screening stages.

Evaluation Framework We evaluated extraction performance according to three measures: *Flagging*, *Count*, and *Extraction*. All three measures are operationalised with standard classification metrics, specifically, we define and collect precision and recall for each.

Flagging measures whether AgentSLR correctly identifies the relevant data types to extract from each article. This measure considers all $\langle \text{article}, \text{data\_type} \rangle$ pairs, assigning labels with the functions

$$y(\langle \text{article}, \text{data\_type} \rangle) = \begin{cases} 1 & \text{There is a human extraction of data\_type from article} \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{y}(\langle \text{article}, \text{data\_type} \rangle) = \begin{cases} 1 & \text{AgentSLR identifies data\_type as relevant in article} \\ 0 & \text{otherwise} \end{cases}$$

and calculating precision and recall on these labels as in a standard binary classification task.

Count measures whether the overall volume of AgentSLR extractions agrees with those in the ground-truth-labelled data, irrespective of any agreement between the extraction contents. We operationalise this measure using a partial credit scheme:

if an article had n models in the reference and our extractor identified $\hat{n}$ models, we counted true positives as correctly matching counts

$$\mathrm{TP} = \min(n, \hat{n}),$$

false positives as excess extractions

$$\mathrm{FP} = \max(0, \hat{n} - n),$$

and false negatives as missed extractions

$$\mathrm{FN} = \max(0, n - \hat{n}).$$

For example, if the reference contained 2 data points but we extracted 5, we would receive credit for the 2 correct extractions ($\mathrm{TP} = 2$), be penalised for 3 spurious models ($\mathrm{FP} = 3$), and would receive no penalty for missed models ($\mathrm{FN} = 0$). We sum all of counts across all common articles and calculate precision and recall as standard.

Extractions faced a more complex matching challenge: while extractions can be trivially compared by raw count, they consist of many metadata fields, and lack unique identifiers to establish canonical correspondence. Matching every field value exactly is an unreasonably challenging task, and it provides no measure beyond absolute correspondence. To assess the field-level quality of our extractions, we first established optimal one-to-one correspondences between ground truth and AgentSLR extractions within each article by computing pairwise similarity.

For each extraction pair, we defined a subset of key fields $\mathcal{F}$ from the fields defined in Appendix E and compared these using normalised weights. The similarity between a true extraction E and an AgentSLR extraction $\hat{E}$ was computed as

$$s(E, \hat{E}) = \sum_{k \in \mathcal{F}} w_k \cdot d_k(E[k], \hat{E}[k]),$$

where w_k is the normalized weight for field k (with $\sum_{k \in \mathcal{F}} w_k = 1$) and d_k is the Jaccard similarity between fields in the extractions

$$d_k(v, \hat{v}) = J(v, \hat{v}) = \frac{|v \cap \hat{v}|}{|v \cup \hat{v}|}.$$

We then applied the modified Jonker–Volgenant algorithm (Jonker & Volgenant, 1987) using SciPy's `scipy.optimize.linear_sum_assignment()`[11] function to the cost matrix (cost $= 1 - s$), finding the matching that maximised total similarity.

Table 18 illustrates this optimal bipartite matching on an example. Suppose a single article has two reference models extracted by expert epidemiologists (PERG), while AgentSLR produces three extractions. Because the sets differ in size, no perfect bijection exists, and the algorithm must leave at least one AgentSLR extraction unmatched. For explanation purposes we restrict to two fields: `model_type`, a single-value field scored by exact match ($\delta_{\text{type}} \in \{0, 1\}$), and `interventions`, a multi-value field scored by Jaccard similarity ($J_{\text{int}} = |v \cap \hat{v}|/|v \cup \hat{v}|$). With equal weights, each pairwise cell reduces to $s_{ij} = 0.5\,\delta_{\text{type}} + 0.5\,J_{\text{int}}$.

The algorithm correctly recovers both reference correspondences – achieving total similarity 2.00 out of a maximum possible 2.00 – while the spurious AgentSLR M_2 is left unmatched and counted as a false positive under the Count metric. Crucially, this unmatched extraction incurs a Count penalty only; it does not contaminate field-level Extraction scores, ensuring over-extraction and extraction inaccuracy are penalised independently.

Once optimal correspondences are established, we evaluated each field within each matched pair to compute field-level precision and recall. For single-value fields, we counted true positives as sets of equal values, false positives as all AgentSLR values with no or an unequal match, and false negatives as all ground-truth values with no or an unequal match.

For multi-value fields, we defined

$$\mathrm{TP} = |v \cap \hat{v}|$$
$$\mathrm{FP} = |\hat{v} \setminus v|$$
$$\mathrm{FN} = |v \setminus \hat{v}|$$

where $v \in E$ and $\hat{v} \in \hat{E}$ are sets of values. Aggregating across all matched pairs and articles, we computed precision and recall as standard.

[11] https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.linear_sum_assignment.html

Table 18. **Optimal bipartite matching example: 2 PERG reference models, 3 AgentSLR-extracted models. (a)** Input field values (two fields shown for illustration). **(b)** Pairwise similarity matrix $\mathbf{S}$. **(c)** Optimal matching; AgentSLR M_2 is unmatched (FP). **(d)** Per-cell similarity calculations for entries of $\mathbf{S}$.

(a) Model Field Values

Model	Type	Interventions
PERG Reference		
PERG M_1	SIR	Vaccination
PERG M_2	SEIR	Quarantine; Vaccination
AI-Extracted		
AgentSLR M_1	SIR	Vaccination
AgentSLR M_2	SIR	Treatment
AgentSLR M_3	SEIR	Quarantine; Vaccination

(b) Pairwise Similarity Matrix S

$$\mathbf{S} = \begin{array}{c} \begin{array}{ccc} \text{AgentSLR M}_1 & \text{AgentSLR M}_2 & \text{AgentSLR M}_3 \end{array} \\ \begin{bmatrix} 1.00 & 0.50 & 0.25 \\ 0.25 & 0.00 & 1.00 \end{bmatrix} \end{array} \begin{array}{l} \leftarrow \text{PERG M}_1 \\ \leftarrow \text{PERG M}_2 \end{array}$$

(c) Optimal Matching

PERG M_1 $\leftrightarrow$ AgentSLR M_1		$s = 1.00$
PERG M_2 $\leftrightarrow$ AgentSLR M_3		$s = 1.00$
AgentSLR M_2 : unmatched	(FP)	—
Total similarity		2.00

(d) Similarity Calculations

$$s_{ij} = \underbrace{0.5\,\delta_{\text{type}}}_{\text{exact match on type}} + \underbrace{0.5\,J_{\text{int}}}_{\text{Jaccard on interventions}}$$

	Type match δ_{type}	**Jaccard** J_{int}	$\boldsymbol{s_{ij}}$
$\mathbf{S}_{1,1}$	**SIR = SIR**: 1.0	$J(\{V\},\{V\}) = 1/1 = 1.0$	**1.00**
$\mathbf{S}_{1,2}$	**SIR = SIR**: 1.0	$J(\{V\},\{T\}) = 0/2 = 0.0$	**0.50**
$\mathbf{S}_{1,3}$	**SIR ≠ SEIR**: 0.0	$J(\{V\},\{Q,V\}) = 1/2 = 0.5$	**0.25**
$\mathbf{S}_{2,1}$	**SEIR ≠ SIR**: 0.0	$J(\{Q,V\},\{V\}) = 1/2 = 0.5$	**0.25**
$\mathbf{S}_{2,2}$	**SEIR ≠ SIR**: 0.0	$J(\{Q,V\},\{T\}) = 0/3 = 0.0$	**0.00**
$\mathbf{S}_{2,3}$	**SEIR = SEIR**: 1.0	$J(\{Q,V\},\{Q,V\}) = 2/2 = 1.0$	**1.00**

V = Vaccination, Q = Quarantine, T = Treatment; $J(A,B) = |A \cap B| \,/\, |A \cup B|$.

DATA EXTRACTION: PARAMETERS

Parameter extraction is more varied than model and outbreak extraction. While there is only one data_type for each of model and outbreak extraction, parameters are broken down into nine distinct parameter *classes* (listed in Appendix E.1) each with different fields to extract. Therefore, we resolve nine parameter data_types at the level of parameter classes and calculate Flagging and Count metrics for each of these separately.

We defined our key parameter fields as

$$\begin{aligned} \mathcal{F} = \{ &\texttt{parameter\_class}, \texttt{parameter\_type}, \texttt{value}, \texttt{unit}, \texttt{method}, \texttt{value\_type}, \\ &\texttt{statistical\_approach}, \texttt{paired\_uncertainty}, \texttt{single\_type\_uncertainty}, \\ &\texttt{population\_sex}, \texttt{population\_group}, \texttt{population\_sample\_type}\}, \end{aligned}$$

ensuring that each sub-stage of value extraction, uncertainty extraction, and population context extraction are represented by multiple fields common across parameter classes. We normalise weights w_k so as to make each sub-stage equally important in determining similarity. Fields are grouped as follows:

- *Categorical fields* (2 fields): parameter class; parameter type;
- *Value fields* (3 fields): value; unit; method;
- *Uncertainty fields* (4 fields): value type; statistical approach; single type uncertainty; paired uncertainty;
- *Population fields* (3 fields): population sex; population group; population sample type.

DATA EXTRACTION: TRANSMISSION MODELS

Table 19 shows the filtering statistics across pathogens: ground-truth datasets had between 3.85% (Lassa) and 23.14% (Zika) invalid entries removed.

Table 19. **Validation statistics for PERG reference data and AI-extracted transmission model annotations across four pathogens.** PERG entries contained invalid field values due to manual data entry inconsistencies, while AI-extracted values showed no invalid entries due to structured schema enforcement during extraction.

Pathogen	PERG Total	PERG Invalid	Invalid (%)	AgentSLR Total
Lassa	52	2	3.85	19
Ebola	294	46	15.7	239
SARS	112	8	7.14	85
Zika	229	53	23.1	132

For data extraction for models, we defined our key fields as

$$\mathcal{F} = \{\texttt{model\_type}, \texttt{compartmental\_type}, \texttt{stoch\_deter}, \texttt{theoretical\_model}, \texttt{assumptions}, \texttt{interventions\_type}, \texttt{transmission\_route}\}.$$

DATA EXTRACTION: OUTBREAKS

Table 20 shows the filtering statistics across pathogens: PERG datasets had between 0% (Lassa) and 9.43% (Zika) invalid entries removed.

Table 20. **Validation statistics for PERG reference data and AI-extracted outbreak annotations across two pathogens.** PERG entries contained invalid field values due to manual data entry inconsistencies, while AI-extracted values showed 0% invalid entries due to structured schema enforcement during extraction.

Pathogen	PERG Total	PERG Invalid	Invalid (%)	AI-Extracted Total
Lassa	30	0	0.00	62
Zika	159	15	9.43	240

For data extraction for outbreaks, we defined our key fields as

$$\mathcal{F} = \{\texttt{outbreak\_start\_day}, \texttt{outbreak\_start\_month}, \texttt{outbreak\_start\_year}, \texttt{outbreak\_end\_day}, \texttt{outbreak\_end\_month}, \texttt{outbreak\_end\_year}, \texttt{cases\_confirmed}, \texttt{deaths}, \texttt{outbreak\_country}, \texttt{outbreak\_location}, \texttt{detection\_mode}, \texttt{pre\_outbreak\_status}\}.$$

Weights w_k were determined by the discriminative power of each field k for identifying unique outbreak events. `outbreak_country`, `outbreak_start_year`, `cases_confirmed`, and `deaths` received weights of 1.0, while supporting temporal fields (`outbreak_start_month`, `outbreak_end_year`) received weights of 0.6–0.8, and contextual fields (`outbreak_location`, `mode_of_detection`) received weights of 0.5–0.7.

To provide interpretable summaries of extraction performance, we grouped the 17 outbreak fields into four categories based on their epidemiological function:

- *Temporal Features* (7 fields): outbreak start/end dates (year, month, day) and duration;
- *Geographic and Spatial Features* (2 fields): outbreak country and specific location;
- *Case Burden* (5 fields): confirmed, suspected, asymptomatic, and severe case counts, plus deaths;
- *Epidemiological Context and Metadata* (3 fields): mode of detection, pre-outbreak status, and asymptomatic transmission description.

G.3. Human Expert Validation

For human expert validation, we recruited six epidemiologists to complete a series of form submissions to grade AgentSLR-generated data extractions. Each epidemiologist was onboarded with the expectation to spend up to 10 hours on the validation process over 1 to 2 weeks as their availability permitted. We did not assign experts randomly across parameters, models, and outbreaks—instead, we considered expertise with specific pathogens and familiarity with specific SLR workflows when making assignments. Our assignments resulted in three epidemiologists completing validation solely for parameters, two solely for models, and the final sixth epidemiologist completing validation across all three data modalities.

For each data type in parameters, models, and outbreaks, and for each pathogen in Lassa, Ebola, SARS, and Zika, we sample screened articles randomly without replacement until we generate subsamples guaranteed to exceed the time commitment from each expert. The experts are then instructed to proceed through their assigned extractions in order. Despite normalising for counts across different pathogens, different articles may have varying numbers of extractions, and these extractions may take varying amounts of time to grade. Thus, our experimental setup does not guarantee parity across pathogens or across data types.

Experts are onboarded with a private GitHub repository that contains the Markdown extractions from our OCR model (Section 2), along with Markdown documents rendering the structured data extractions in a readable format. Submissions are collected through Google Forms. Each form proceeds through groups of questions in the same order. Each question contains an optional free-text field for providing context, which we use to collect and synthesise qualitative impressions of the pipeline as well as specific error patterns.

The groups of questions in each form cover the following:

1. The expert records the article identifier, pathogen identifier, and the pathogen.
2. The expert assesses whether the Markdown document has any significant issues that would affect data extraction.
3. Before looking at the AgentSLR-extracted data, the expert determines whether there is any relevant data in the article to extract.
4. The expert rates their particular extraction for overall relevance.
5. The expert answers a series of yes-or-no questions to validate the accuracy of each extracted field.
6. The expert grades the overall pipeline competence using a Likert scale rating between 1 and 7. We provide these particular descriptions to calibrate the Likert scale:
   - “1" means “the system gets nothing right; I couldn’t use it to speed up my process at all."
   - “4" means “the system identifies some things but struggles with edge cases; I could use it with moderate supervision / secondary screening."
   - “7" means “the system is perfectly capable of doing all parameter extraction for me."
7. The expert provides a self-reported estimate of the time they took to complete the survey.

H. Pipeline Statistics: Data Processed & Time

H.1. Runtime Statistics

Article counts across SLR stages

To contextualise runtime estimates for both human and automated pipelines, we first summarise the approximate number of articles processed at each stage of the systematic literature review (SLR). These counts are intended to reflect annotator workload rather than final inclusion totals, and correspond to successive filtering stages commonly used in SLR workflows. In particular, counts decrease substantially between title and abstract screening, full-text screening, and data extraction as relevance criteria are progressively applied.

Table 21. **Estimated number of articles reviewed by human annotators at PERG across successive stages of each systematic literature review (SLR).** Counts for *Title and Abstract Screening* correspond to records remaining after deduplication and exclusion of entries with missing or empty abstract metadata. *Full-text Screening* includes articles flagged as potentially relevant during abstract screening and advanced for full-text review. *Data Extraction* represents articles deemed suitable for extracting structured, task-relevant evidence. All values are estimates intended to reflect annotator workload at each phase rather than finalised inclusion totals.

Pathogen	Title & Abstract Screening	Full-text Screening	Data Extraction
Ebola	11,605	1,674	522
Lassa	2,131	512	193
SARS	12,280	878	289
Zika	10,510	1,343	574
Average	9,132	1,102	395

Runtime estimation methodology

Using the average article counts from Table 21, we estimate total processing time for both the PERG human SLR workflow and the AgentSLR automated pipeline. Per-article time estimates for PERG were obtained through consultation with a Research Associate at PERG who routinely contributes to SLR projects. AgentSLR runtimes were measured directly from pipeline execution logs. All per-article times are converted to hours and multiplied by the average number of articles processed at each stage.

Table 22. **Comparison of average human time investment (PERG) versus automated processing time (AgentSLR) across systematic literature review stages.** The table reports average articles processed across the four pathogens (Ebola, Lassa, SARS, Zika), average per-article time (in seconds), and total processing time (in hours), highlighting efficiency gains from automation .AgentSLR timings are computed using `gpt-oss-120b` as the underlying model. PDF-to-Markdown conversion is applied to all 9,132 retrieved articles to preserve the option of direct full-text screening across the complete corpus.

Stage	Articles (Avg.)	AgentSLR (s/article)	PERG (s/article)	$\overline{\text{AgentSLR}}$ (Hours)	$\overline{\text{PERG}}$ (Hours)
Article Retrieval	9,132	0.63	0	1.6	0.00
Title & Abstract Screening	9,132	0.63	45	1.6	114.2
PDF-to-Markdown Conversion	9,132	1.1	0	2.8	0.00
Full-text Screening	1,102	2.0	240	0.62	73.5
Data Extraction	395	122.1	1,800	13.4	197.5
Total	–	–	–	**20.0**	**385.1**

PERG runtime calculations

Title and abstract screening at PERG is estimated at 30 to 60 seconds per article. Assuming an average of 45 seconds (0.0125 hours) per article and 9,132 articles screened on average, the estimated time is $9{,}132 \times 0.0125 = 114.15$ hours.

Full-text screening is estimated at 2 to 6 minutes per article. Assuming an average of 4 minutes ($0.066\overline{6}$ hours) per article and 1,102 articles screened on average, the estimated time is $1{,}102 \times 0.066\overline{6} = 73.47$ hours.

Data extraction is estimated at a median of 30 minutes (0.5 hours) per article. With 395 articles processed on average, the estimated time is $395 \times 0.5 = 197.50$ hours.

AGENTSLR RUNTIME CALCULATIONS

Article retrieval in AgentSLR requires 0.63 seconds per article. With 9,132 articles retrieved on average, the estimated time is 1.6 hours.

Title and abstract screening requires 0.63 seconds per article. With 9,132 articles screened on average, the estimated time is 1.6 hours.

PDF-to-Markdown conversion requires 1.1 seconds per article. PDF-to-Markdown conversion is applied to all 9,132 retrieved articles to preserve the option of direct full-text screening across the complete corpus, giving an estimated time of 2.8 hours. This estimate reflects parallel execution with 14 concurrent requests, yielding an average processing time of 0.05 seconds per page and 1.1 seconds per document. Under sequential execution, the measured processing time increases substantially to an average of 0.95 seconds per page and 16.47 seconds per document.

Full-text screening requires 2.0 seconds per article. With 1,102 articles screened on average, the estimated time is 0.62 hours.

Data extraction requires 122.1 seconds per article, comprising outbreak identification, model extraction, and parameter extraction. With 395 articles processed on average, the estimated time is 13.4 hours.

H.2. Token Usage and Operational Cost of AgentSLR

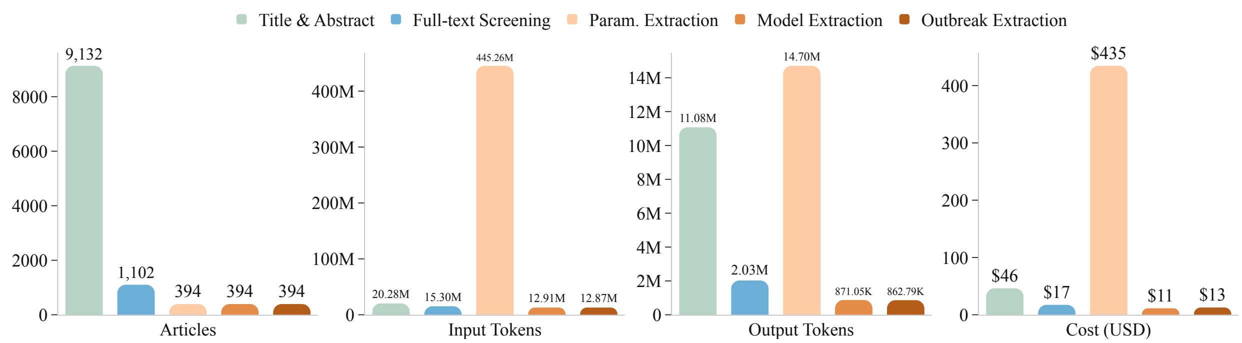


Figure 7. **Articles processed, scaled token and costs by pipeline stage across models.** We report the average number of articles reaching each stage, and the corresponding total input tokens, output tokens, and USD cost per stage averaged across models. Token totals are computed by multiplying per article token usage (Table 23) by the average article counts per stage (Table 21). Parameter extraction dominates overall compute, with substantially higher input and output token totals than other stages. Title and abstract screening processes the largest volume of articles but contributes comparatively less to total cost.

Using the average article counts reported in Table 21, we estimate total token usage and USD cost across pipeline stages by combining per-article token statistics with model-specific pricing. Figure 7 summarises the resulting distribution of articles processed, aggregate input tokens, output tokens, and total cost by stage, averaged across models. Per-article input and output token usage by stage and model is reported in Table 23. Total stage costs are computed by multiplying mean per-article token usage by the average number of articles reaching each stage, and then applying published per-million-token pricing for both input and output tokens. All prices used in these calculations are retrieved directly from the primary API pricing documentation of each model provider at the time of evaluation. Under this pricing regime, parameter extraction dominates overall compute and cost due to substantially higher input and output token volumes, while title and abstract screening processes the largest number of articles but contributes comparatively little to total cost. All reported costs reflect managed API usage; alternative cost estimates could be derived for deployments hosted on dedicated GPU nodes, where pricing would depend on hardware configuration, utilisation, and amortisation assumptions rather than per-token billing.

[11]**GPT-OSS-120B:** https://openrouter.ai/openai/gpt-oss-120b.
GPT-5.2: https://developers.openai.com/api/docs/pricing/.
DeepSeek-V3.2: https://api-docs.deepseek.com/quick_start/pricing.
Kimi-K2.5: https://platform.moonshot.ai/docs/pricing/chat.

Table 23. **Per article token usage and estimated cost by stage and model.** We report mean input tokens, output tokens, and USD cost for processing a **single article** at each stage. Green marks the minimum and Red marks the maximum within each row and subcolumn.

Stage	GPT-OSS-120B (High)			GPT-5.2 (High)			DeepSeek-V3.2			Kimi-K2.5			GLM-4.7		
	Input Tok.	Output Tok.	Cost (USD)	Input Tok.	Output Tok.	Cost (USD)	Input Tok.	Output Tok.	Cost (USD)	Input Tok.	Output Tok.	Cost (USD)	Input Tok.	Output Tok.	Cost (USD)
Title & Abstract Screening	2.3K	1.2K	< 0.01	2.2K	0.6K	0.01	2.2K	0.6K	< 0.01	2.3K	2.0K	< 0.01	2.2K	1.6K	< 0.01
Article Screening (AI Conditioned)	16.9K	1.1K	< 0.01	13.1K	1.4K	0.04	12.9K	0.8K	< 0.01	13.1K	2.7K	0.01	13.4K	3.1K	0.01
Parameter Extraction	510.2K	19.8K	0.02	961.1K	91.1K	2.95	523.2K	3.0K	0.14	605.5K	40.0K	0.48	3050.4K	32.7K	1.90
Model Extraction	35.9K	1.9K	< 0.01	31.5K	2.1K	0.08	37.2K	3.1K	0.01	29.3K	2.2K	0.02	29.7K	1.7K	0.02
Outbreak Extraction	49.9K	2.6K	< 0.01	32.4K	3.5K	0.10	26.1K	0.2K	< 0.01	29.5K	2.7K	0.02	25.4K	2.0K	0.01
Overall	615.3K	26.7K	0.02	1040.4K	98.7K	3.20	601.6K	7.8K	0.17	679.8K	49.6K	0.55	3121.1K	41.1K	1.96

GLM-4.7: https://docs.z.ai/guides/overview/pricing.

I. Extended Results

This section reports disaggregated results across pathogens and stages for AgentSLR (`gpt-oss-120b`), presented in Section 4.2.

I.1. Article Screening

Title and Abstract Screening

Table 24 summarises title-and-abstract screening performance across seven pathogens, showing moderate overall recall (0.72) alongside high precision (0.79), for an overall F_1 of 0.74. This pattern suggests the abstract-stage screening is tuned toward specificity, prioritising the rejection of irrelevant studies at the cost of more false negatives. Performance is broadly consistent across pathogens, with the strongest balance for MERS (F_1 of 0.78) and the weakest for Marburg (F_1 of 0.69).

Table 24. **Precision, recall, and F_1 for title-and-abstract screening across seven pathogens with AgentSLR (`gpt-oss-120b`).** Metrics summarise how well the abstract-stage classifier retained studies judged relevant under PERG screening criteria, reported for each pathogen and overall. Overall performance (precision 0.79, recall 0.72; F_1 0.74) reflects a specificity-oriented triage step that prioritises avoiding false inclusions, with lower recall indicating that some relevant studies may require recovery at the full-text stage. P = precision; R = recall; F_1 = F1-Score.

Pathogen	P	R	F_1
Marburg	0.80	0.64	0.69
Ebola	0.74	0.75	0.75
Lassa	0.82	0.72	0.75
SARS	0.78	0.76	0.77
Zika	0.73	0.77	0.75
MERS	0.83	0.74	0.78
Nipah	0.84	0.66	0.70
Overall	0.79	0.72	0.74

Full-text Screening

Table 25 compares three full-text screening strategies and highlights a clear precision–recall trade-off. Human abstract → AI full-text achieves the strongest overall performance (precision 0.83, recall 0.92), while the fully automated two-stage pipeline (AI abstract → AI full-text) shows lower recall (0.81), consistent with error propagation from abstract gating. Direct AI full-text screening improves recall (0.89) but reduces precision (0.68), reflecting a recall-maximising approach when abstracts are treated as an information bottleneck.

Table 25. **Full-text screening performance on AgentSLR (`gpt-oss-120b`) under three operational strategies for identifying relevant articles.** Metrics compare a two-stage AI pipeline (AI abstract→AI full-text), a mixed workflow (human abstract→AI full-text), and direct AI full-text screening, reported for each pathogen and overall. Results show the trade-off between recall preservation and precision control: human abstract gating yields the highest overall F_1 score (0.87), while direct AI full-text maximises recall (0.89) at the cost of precision (0.68), consistent with abstracts acting as an information bottleneck. P = precision; R = recall; F_1 = F1-Score.

Pathogen	**AI Screen (Abstract) → AI Screen (Full-text)**			**Human Screen (Abstract) → AI Screen (Full-text)**			**AI Screen (Direct Full-text)**		
	P	R	F_1	P	R	F_1	P	R	F_1
Marburg	0.75	0.76	0.75	0.77	0.83	0.80	0.64	0.82	0.69
Ebola	0.73	0.84	0.77	0.86	0.97	0.91	0.67	0.93	0.72
Lassa	0.79	0.78	0.78	0.83	0.94	0.88	0.71	0.91	0.77
SARS	0.71	0.85	0.76	0.80	0.95	0.86	0.64	0.91	0.68
Zika	0.66	0.79	0.69	0.81	0.91	0.85	0.64	0.85	0.67
MERS	0.76	0.83	0.79	0.83	0.96	0.88	0.69	0.95	0.76
Nipah	0.87	0.84	0.85	0.89	0.90	0.90	0.74	0.88	0.79
Overall	0.75	0.81	0.77	0.83	0.92	0.87	0.68	0.89	0.73

I.2. Data Extraction

In this section, we provide complete disaggregated results for our data extraction evaluations comparing AgentSLR extractions against our ground-truth-labelled datasets. The ground-truth-labelled datasets are provided open-source from the Pathogen Epidemiology Review Group, available through the R `epireview` package or on GitHub at https://github.com/mrc-ide/epireview/tree/main/inst/extdata.

As of March 2026, owing to PERG's continual progress through SLRs on nine priority pathogens, ground-truth extraction data is available in a standardised format for four pathogens: Lassa, Ebola, SARS, and Zika. For each pathogen, we evaluate classification measures for each of the *Flagging*, *Count*, and *Extraction* metrics defined formally in Appendix G.2.

PARAMETERS

Table 26 presents the results for parameter extraction Flagging and Count metrics. These results are used to produce the aggregate data presented in the main body text (Table 2 in Section 4). For flagging relevant parameters, AgentSLR performs consistently across all pathogens with high recall (0.92 average), though precision is lower and more variable (0.51 average). The results suggest that while AgentSLR is able to identify nearly all relevant extractions, this coverage comes at the cost of many false positive flags that may propagate errors to later sub-stages. In terms of overall parameter extraction counts, the performance flips in favour of precision (0.83) now at the expense of lower recall (0.47). The discrepancy with parameter flagging performance is understandable, as AgentSLR is capable of disregarding flagged parameters when provided with the option for structured extraction via tool calls. Our data suggests that when AgentSLR produces a final extraction, this extraction is likely correct, however the system often fails to produce all required extractions according to ground-truth data. An article may have multiple extractions of the same parameter class, and in these cases AgentSLR can underestimate the number of extractions required.

Table 26. **Flagging and Count classification metrics for parameter extraction with AgentSLR (`gpt-oss-120b`).** P = precision; R = recall; F_1 = F1-Score.

Metric	**Lassa**			**Ebola**			**SARS**			**Zika**		
	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Flagging	0.56	0.98	0.71	0.60	0.92	0.72	0.50	0.81	0.62	0.40	0.96	0.57
Count	1.00	0.35	0.51	0.79	0.47	0.59	0.80	0.61	0.69	0.72	0.47	0.57

Table 27. **Field-level precision, recall, and F_1 for Extraction on parameters with AgentSLR (`gpt-oss-120b`).** *Group* corresponds to the sub-stage of parameter extraction where the field is collected. The final row shows averages across all fields. P = precision; R = recall; F_1 = F1-Score.

Group	**Field**	**Lassa**			**Ebola**			**SARS**			**Zika**		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Value	value	0.22	0.22	0.22	0.20	0.20	0.20	0.23	0.23	0.23	0.14	0.14	0.14
	unit	0.50	0.43	0.46	0.62	0.35	0.44	0.69	0.61	0.65	0.65	0.43	0.52
	method	1.00	0.89	0.94	0.48	0.78	0.59	0.76	0.83	0.79	0.86	0.80	0.83
	Average	0.57	0.51	0.54	0.44	0.44	0.41	0.56	0.56	0.56	0.55	0.46	0.50
Uncertainty	value type	0.38	0.43	0.40	0.30	0.33	0.32	0.35	0.57	0.43	0.12	0.22	0.16
	statistical approach	–	–	–	–	–	–	–	–	–	0.44	0.66	0.53
	single type uncertainty	1.00	1.00	1.00	0.98	0.94	0.96	0.81	0.95	0.88	0.97	0.99	0.98
	paired uncertainty	0.25	0.40	0.31	0.59	0.72	0.65	0.39	0.88	0.54	0.46	0.90	0.61
	Average	0.54	0.61	0.57	0.62	0.67	0.64	0.52	0.80	0.62	0.50	0.69	0.57
Population	population sex	0.86	0.67	0.75	0.62	0.79	0.69	0.59	0.75	0.66	0.59	0.70	0.64
	population group	0.14	0.11	0.12	0.24	0.33	0.28	0.23	0.25	0.24	0.54	0.54	0.54
	population sample type	0.86	0.67	0.75	0.32	0.41	0.36	0.58	0.63	0.60	0.37	0.36	0.37
	Average	0.62	0.48	0.54	0.40	0.51	0.44	0.47	0.54	0.50	0.50	0.54	0.52
Overall		0.58	0.53	0.55	0.49	0.54	0.50	0.51	0.63	0.56	0.51	0.57	0.53

For Extraction, complete field-level results are presented in Table 27. The aggregate results, presented in the main text, show parameter extractions to have moderate quality across all pathogens, with little variation among them. Analysing the results at the field-level reveals patterns in AgentSLR's handling of different data modalities as well as different types of epidemiological context. The system performs worst on value fields, with population fields also showing relatively weak performance, compared to other groups. We suspect the difficulty with population context arises from the large numbers of valid options for many fields (notably population group and population sample type), with many of these options having precise interpretations in epidemiological literature. Without fine-tuning, `gpt-oss-120b` may struggle to apply these interpretations in a complex tool-calling environment. On the other hand, classification is near perfect for single type uncertainty, and generally strong for method (with the exception of Ebola). We also note that fields with unrestricted domains, like value, are much harder to classify correctly. Seeing the much improved results in our expert validation experiment (Section 4.3), we suspect at least some of this difficulty to stem from our exact-match criteria being overly punitive of equivalent numbers in different formats.

Transmission Models

Table 28 presents the complete results for Flagging, Count, and Extraction evaluations of transmission models across the four priority pathogens. Screening performance was strong across all pathogens for article flagging, with recall ranging from 0.86 to 0.99 and precision from 0.86 to 0.96, indicating reliable identification of modelling studies. Model count extraction achieved consistently high recall (0.97–1.00) but notably lower precision (0.48–0.60), suggesting a systematic tendency to overestimate the number of models reported per article rather than failing to identify them.

Field-level extraction showed a clear gradient in task difficulty. Core structural characteristics were extracted with high accuracy: model type classification and the theoretical versus data-fitted distinction achieved balanced precision and recall between 0.62 and 0.89 across pathogens, while single-value fields such as stochastic versus deterministic modelling and code availability frequently exceeded 0.75 and reached perfect scores for some pathogens. In contrast, more complex or multi-value fields exhibited substantially lower performance. Transmission route extraction was particularly challenging for Ebola and SARS, while assumptions and interventions showed modest precision and recall across all pathogens. Overall, across screening and extraction tasks, precision ranged from 0.61 to 0.70 and recall from 0.75 to 0.81, indicating that the system reliably captures core model characteristics, with remaining limitations concentrated in the extraction of nuanced descriptive details.

Table 28. **Precision, recall, and F_1 metrics for transmission model screening and extraction across four pathogens with AgentSLR (`gpt-oss-120b`).** *Screening* includes article flagging and model count accuracy. *Extraction* evaluates field-level accuracy for matched model pairs, covering core structural characteristics (model type, stochastic vs deterministic, theoretical vs data-fitted, code availability) and more complex multi-value fields (transmission routes, assumptions, interventions). Strong performance is observed for core model characteristics, while extraction of assumptions, interventions, and transmission routes remains more challenging. P = precision; R = recall; F_1 = F1-Score.

	Lassa			**Ebola**			**SARS**			**Zika**		
	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
Flagging												
Article Flagging	0.95	0.99	0.97	0.92	0.92	0.92	0.86	0.86	0.86	0.87	0.89	0.88
Counts												
Model Count	0.60	1.00	0.75	0.50	1.00	0.67	0.49	0.97	0.65	0.48	0.98	0.65
Extraction												
Model Type	0.89	0.89	0.89	0.89	0.89	0.89	0.77	0.77	0.77	0.88	0.88	0.88
Compartmental Type	0.00	0.00	0.00	—	—	—	0.80	0.80	0.80	0.83	0.83	0.83
Stochastic vs Deterministic	1.00	1.00	1.00	0.75	0.85	0.80	0.76	0.78	0.77	0.82	0.79	0.81
Theoretical vs Data-Fitted	0.78	0.78	0.78	0.88	0.88	0.88	0.62	0.62	0.62	0.81	0.81	0.81
Code Available	1.00	0.89	0.94	0.85	0.84	0.84	1.00	1.00	1.00	0.82	0.76	0.79
Transmission Routes	1.00	0.94	0.97	0.13	0.15	0.14	0.26	0.32	0.29	0.68	0.74	0.71
Assumptions	0.29	0.46	0.36	0.27	0.46	0.34	0.21	0.39	0.28	0.31	0.52	0.39
Interventions	0.54	0.64	0.58	0.48	0.69	0.56	0.46	0.79	0.58	0.32	0.69	0.43
Overall	0.70	0.78	0.73	0.62	0.77	0.67	0.61	0.75	0.66	0.66	0.81	0.71

Table 29. **Precision, recall, and F_1 for outbreak screening and extraction with AgentSLR (`gpt-oss-120b`) across major feature categories, evaluated against expert-curated PERG database.** *Screening* measured article flagging (identifying papers containing outbreaks) and outbreak count accuracy (extracting the correct number of outbreaks per paper). *Extraction* evaluated field-level accuracy for matched outbreak pairs across four epidemiological categories: temporal features (start/end dates), geographic features (country, specific location), case burden (confirmed cases, deaths), and epidemiological context (detection mode, pre-outbreak status, ongoing status, asymptomatic transmission). Overall metrics represent the average across all extraction fields. P = precision; R = recall; F_1 = F1-Score.

Metric	Field	Lassa			Zika		
		P	R	$F1$	P	R	$F1$
Flagging	Article Flagging	0.69	0.82	0.75	0.58	0.71	0.64
Counts	Outbreak Counts	0.83	1.00	0.91	0.49	0.45	0.47
Extraction	Temporal Features	0.83	0.74	0.78	0.85	0.82	0.83
	Geographic and Spatial Features	0.75	0.78	0.76	0.75	0.75	0.75
	Case Burden	0.82	0.75	0.79	0.93	0.93	0.93
	Epidemiological Context and Metadata	0.93	0.70	0.80	0.84	0.67	0.75
Overall		0.85	0.73	0.79	0.84	0.78	0.81

OUTBREAKS

Similar to transmission models, applying the evaluation framework described in Appendix G, we analysed outbreak extraction performance across two priority pathogens, Lassa and Zika (Table 29). Screening performance differed between pathogens and screening subtasks. For Lassa, article flagging achieved moderate precision (0.69) and strong recall (0.82), indicating that most outbreak-containing papers were identified, although a non-trivial fraction of flagged papers were false positives. For Zika, article flagging was more balanced (precision 0.58, recall 0.71), suggesting improved sensitivity relative to precision, but still leaving missed outbreak descriptions and over-inclusion of non-outbreak papers. Outbreak counting

Table 30. **Detailed (expanded) precision, recall, and F_1 for outbreak feature extraction by category for AgentSLR (`gpt-oss-120b`).** Each row shows field-level performance within the four major epidemiological categories. Temporal and case burden features showed consistently high performance, while location-specific fields and epidemiological context features showed greater variability. Overall metrics represent the average across all 13 extraction fields. P = precision; R = recall; F_1 = F1-Score.

	Lassa			Zika		
	P	R	$F1$	P	R	$F1$
Temporal Features						
Start Year	0.89	0.80	0.84	0.90	0.82	0.86
Start Month	0.78	0.78	0.78	0.80	0.80	0.80
Start Day	0.86	0.67	0.75	0.95	0.95	0.95
End Month	0.78	0.78	0.78	0.65	0.65	0.65
End Day	0.86	0.67	0.75	0.95	0.86	0.90
Average	0.83	0.74	0.78	0.85	0.82	0.83
Geographic and Spatial Features						
Outbreak Country	1.00	1.00	1.00	1.00	1.00	1.00
Location	0.50	0.56	0.53	0.50	0.50	0.50
Average	0.75	0.78	0.77	0.75	0.75	0.75
Case Burden						
Confirmed Cases	0.75	0.60	0.67	0.86	0.86	0.86
Deaths	0.90	0.90	0.90	1.00	1.00	1.00
Average	0.83	0.75	0.79	0.93	0.93	0.93
Epidemiological Context and Metadata						
Mode of Detection	0.71	0.50	0.59	0.50	0.41	0.45
Pre-outbreak Status	1.00	0.30	0.46	1.00	0.41	0.58
Ongoing Status	1.00	1.00	1.00	0.86	0.86	0.86
Asymptomatic Transmission	1.00	1.00	1.00	1.00	1.00	1.00
Average	0.93	0.70	0.76	0.84	0.67	0.72
Overall	0.85	0.73	0.79	0.84	0.78	0.81

remained strong for Lassa (precision 0.83, recall 1.00), while Zika outbreak counting was substantially lower (precision 0.49, recall 0.45), consistent with continued difficulty in reliably enumerating outbreak events in the Zika corpus.

Field-level extraction performance, grouped by epidemiological feature categories, revealed consistent strengths and persistent weaknesses (Table 30). Temporal features remained robust across both pathogens (Lassa: precision 0.83, recall 0.74; Zika: precision 0.85, recall 0.82), with high precision for start year (0.89) and start day (0.86) in Lassa, and strong start day accuracy in Zika (precision 0.95, recall 0.95). Case burden metrics showed overall extraction (Lassa: precision 0.83, recall 0.75; Zika: precision 0.93, recall 0.93), with high accuracy for deaths (Lassa: 0.90/0.90; Zika: 1.00/1.00).

Geographic extraction continued to show a split between coarse and fine granularity. Outbreak country identification was perfect for both pathogens (1.00 precision and recall), while specific location extraction was notably weaker (Lassa: precision 0.50, recall 0.56; Zika: precision 0.50, recall 0.50), consistent with variability in how places are described in scientific text. Epidemiological context fields showed the greatest variability: for Lassa, mode of detection was moderate (0.71 precision, 0.50 recall) and pre-outbreak status exhibited high precision but low recall (1.00/0.30), indicating frequent omission of this attribute. Ongoing status was extracted perfectly for Lassa (1.00/1.00) but showed moderate performance for Zika (0.86/0.86). For Zika, mode of detection (0.50/0.41) and pre-outbreak status (1.00/0.41) remained challenging, although asymptomatic transmission was extracted perfectly for both pathogens (1.00/1.00). The overall extraction performance averaged 0.85 precision and 0.73 recall for Lassa, and 0.84 precision and 0.78 recall for Zika, suggesting reliable field-level accuracy with remaining gaps concentrated in context-dependent and location-specific attributes.

J. Model Ablation Results

This section reports full pathogen-level and metric-level results for the five model ablations described in Section 5.

J.1. Article Screening

Title & Abstract Screening

At title and abstract screening (Table 31), the spread in overall F_1 from 0.62 (`DeepSeek-V3.2`) to 0.77 (`Kimi-K2.5`) is driven almost entirely by recall rather than precision: `DeepSeek-V3.2` and GPT-5.2 are the two most precise models (0.83 and 0.82 respectively) yet rank last and fourth on F_1, with recalls of 0.59 and 0.61 against `Kimi-K2.5`'s 0.75. `Kimi-K2.5` is the best-performing model for all seven pathogens. Nipah is the worst-performing pathogen for three models and sits at or below 0.72 for all five, consistent with it being one of the smallest and most heterogeneous corpora in the PERG dataset; the same pathogen has the narrowest precision-recall gap across models, suggesting the difficulty is intrinsic to the articles rather than a model-specific calibration issue.

Table 31. **Title and abstract screening metrics with model ablations.** Green and Red denote the best- and worst-performing pathogens for each model (in terms of F_1 score). **Bold** indicates the best-performing model for each pathogen, and Underline indicates the second-best. P = precision; R = recall; F_1 = F1-Score.

Pathogen	`gpt-oss-120b`			GPT-5.2			`DeepSeek-V3.2`			`Kimi-K2.5`			`GLM-4.7`		
	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
Marburg	0.80	0.64	0.69	0.97	0.58	0.62	0.97	0.55	0.58	0.79	0.65	**0.69**	0.88	0.61	0.66
Ebola	0.74	0.75	0.75	0.76	0.64	0.68	0.80	0.61	0.64	0.79	0.79	**0.79**	0.88	0.72	0.77
Lassa	0.82	0.72	0.75	0.78	0.63	0.66	0.84	0.60	0.63	0.84	0.77	**0.80**	0.88	0.68	0.73
SARS	0.78	0.76	0.77	0.77	0.62	0.65	0.82	0.62	0.66	0.80	0.78	**0.79**	0.89	0.73	0.78
Zika	0.73	0.77	0.75	0.70	0.62	0.64	0.73	0.63	0.66	0.78	0.79	**0.79**	0.76	0.69	0.72
MERS	0.83	0.74	0.78	0.87	0.62	0.67	0.86	0.60	0.65	0.86	0.78	**0.81**	0.89	0.67	0.73
Nipah	0.84	0.66	0.70	0.92	0.58	0.59	0.81	0.54	0.53	0.85	0.68	**0.72**	0.90	0.61	0.65
Overall	0.79	0.72	0.74	0.82	0.61	0.65	0.83	0.59	0.62	0.82	0.75	**0.77**	0.87	0.67	0.72

Full-text Screening

The ranking reorders substantially at full-text screening (Table 32), where `gpt-oss-120b` leads (F_1 0.77) and the two highest-precision abstract-stage models fall furthest. `DeepSeek-V3.2`'s precision drops from 0.83 to 0.64 and its recall from 0.59 to 0.56, making it the only model that loses ground on both measures simultaneously, with its Marburg result (F_1 0.42, precision 0.37) being the single weakest pathogen-level score in either screening table. `gpt-oss-120b`'s advantage at this stage comes from recall: it achieves 0.81 overall against the next-best `Kimi-K2.5` at 0.73, and is one of only two models – alongside `GLM-4.7` – for which recall increases from abstract to full-text screening. The Nipah-to-Zika contrast also reverses: Nipah is the best-performing pathogen for `gpt-oss-120b` and `Kimi-K2.5` at full-text (F_1 0.85 and 0.82), whereas it was the worst for three models at the abstract stage, suggesting that the richer context of full texts resolves ambiguity that titles and abstracts leave open for this pathogen.

Table 32. **Full-text screening metrics with model ablations.** Green and Red denote the best- and worst-performing pathogens for each model (in terms of F_1 score). **Bold** indicates the best-performing model for each pathogen, and Underline indicates the second-best. P = precision; R = recall; F_1 = F1-Score.

Pathogen	`gpt-oss-120b`			GPT-5.2			`DeepSeek-V3.2`			`Kimi-K2.5`			`GLM-4.7`		
	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
Marburg	0.75	0.76	0.75	0.76	0.59	0.59	0.37	0.49	0.42	0.66	0.66	0.66	0.86	0.72	**0.76**
Ebola	0.73	0.84	**0.77**	0.61	0.60	0.60	0.68	0.59	0.55	0.72	0.74	0.71	0.75	0.75	0.75
Lassa	0.79	0.78	**0.78**	0.66	0.63	0.63	0.63	0.54	0.47	0.74	0.75	0.74	0.77	0.73	0.73
SARS	0.71	0.85	**0.76**	0.60	0.58	0.58	0.73	0.61	0.59	0.67	0.69	0.66	0.73	0.72	0.72
Zika	0.66	0.79	**0.69**	0.50	0.50	0.50	0.61	0.55	0.52	0.63	0.64	0.61	0.60	0.59	0.59
MERS	0.76	0.83	**0.79**	0.66	0.61	0.61	0.74	0.60	0.58	0.76	0.79	0.77	0.76	0.68	0.69
Nipah	0.87	0.84	**0.85**	0.80	0.63	0.63	0.73	0.56	0.53	0.83	0.81	0.82	0.72	0.61	0.61
Overall	0.75	0.81	**0.77**	0.66	0.59	0.59	0.64	0.56	0.52	0.72	0.73	0.71	0.74	0.69	0.69

J.2. Data Extraction

PARAMETERS

Parameter extraction results are disaggregated by pathogen and extraction type in Table 33. `Kimi-K2.5` achieves the highest overall average F_1 (0.63), marginally ahead of `GLM-4.7` (0.63), and lower performance seen for `gpt-oss-120b` (0.59), GPT-5.2 (0.58) and `DeepSeek-V3.2` (0.56). Performance is most variable in the Counts sub-task: `gpt-oss-120b` attains strong precision (0.83) but low recall (0.47), reproducing the asymmetric pattern observed for the primary model in Appendix I, while GPT-5.2 shows the inverse (recall 0.83, precision 0.36). At the field-level Extraction sub-task, GPT-5.2 achieves the highest overall F_1 (0.59), followed by `Kimi-K2.5` (0.56), and the five models are broadly comparable, consistent with the interpretation that cross-model differences in average F_1 are driven by flagging and counting behaviour rather than the quality of individual field extractions. Zika is the weakest pathogen across most models and sub-tasks, while SARS is frequently the best-performing for `gpt-oss-120b`.

Table 33. **Parameter extraction metrics with model ablations. Average** denotes means across sub-tasks; **Overall** denotes means across pathogens. Green and Red denote the best- and worst-performing pathogens for each model (in terms of F_1 score). **Bold** indicates the best-performing model for each pathogen, and Underline indicates the second-best. P = precision; R = recall; F_1 = F1-Score.

Pathogen	Type	gpt-oss-120b			GPT-5.2			DeepSeek-V3.2			Kimi-K2.5			GLM-4.7		
		P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
Ebola	Flagging	0.60	0.92	0.72	0.58	0.93	0.71	0.49	0.91	0.64	0.67	0.90	0.77	0.72	0.82	**0.77**
	Counts	0.79	0.47	0.59	0.46	0.80	0.58	0.57	0.59	0.58	0.52	0.73	0.61	0.59	0.65	**0.62**
	Extraction	0.48	0.54	0.50	0.58	0.57	**0.57**	0.54	0.47	0.49	0.55	0.57	0.55	0.51	0.56	0.52
	Average	0.62	0.64	0.60	0.54	0.77	0.62	0.54	0.65	0.57	0.58	0.73	**0.64**	0.61	0.68	0.64
Lassa	Flagging	0.56	0.98	0.71	0.58	1.00	0.73	0.54	0.91	0.68	0.70	0.94	0.81	0.77	0.87	**0.82**
	Counts	1.00	0.35	0.51	0.30	0.85	0.45	0.46	0.57	0.51	0.59	0.81	**0.69**	0.74	0.59	0.66
	Extraction	0.58	0.54	0.55	0.66	0.63	**0.63**	0.55	0.46	0.47	0.58	0.60	0.57	0.58	0.56	0.56
	Average	0.71	0.62	0.59	0.51	0.83	0.61	0.52	0.65	0.55	0.62	0.79	**0.69**	0.70	0.67	0.68
SARS	Flagging	0.50	0.81	**0.62**	0.47	0.83	0.60	0.39	0.78	0.52	0.56	0.69	0.62	0.58	0.67	0.62
	Counts	0.80	0.61	**0.69**	0.37	0.88	0.52	0.60	0.60	0.60	0.45	0.72	0.56	0.51	0.59	0.55
	Extraction	0.51	0.63	0.56	0.56	0.63	**0.58**	0.58	0.54	0.55	0.53	0.65	0.57	0.51	0.62	0.55
	Average	0.61	0.69	**0.62**	0.47	0.78	0.57	0.52	0.64	0.56	0.51	0.69	0.58	0.53	0.63	0.57
Zika	Flagging	0.40	0.96	0.57	0.43	0.95	0.59	0.41	0.94	0.57	0.56	0.86	**0.68**	0.62	0.75	0.68
	Counts	0.72	0.47	0.57	0.31	0.80	0.45	0.50	0.60	0.55	0.55	0.73	0.63	0.59	0.67	**0.63**
	Extraction	0.52	0.57	0.53	0.56	0.59	**0.56**	0.55	0.50	0.51	0.54	0.59	0.55	0.52	0.58	0.54
	Average	0.55	0.67	0.56	0.43	0.78	0.53	0.49	0.68	0.54	0.55	0.73	**0.62**	0.58	0.66	0.61
Overall	Flagging	0.51	0.92	0.66	0.51	0.93	0.66	0.46	0.88	0.60	0.62	0.85	0.72	0.67	0.78	**0.72**
	Counts	0.83	0.47	0.59	0.36	0.83	0.50	0.53	0.59	0.56	0.53	0.75	**0.62**	0.61	0.62	0.61
	Extraction	0.52	0.57	0.54	0.59	0.61	**0.59**	0.56	0.49	0.50	0.55	0.60	0.56	0.53	0.58	0.54
	Average	0.62	0.65	0.59	0.49	0.79	0.58	0.52	0.66	0.56	0.57	0.73	**0.63**	0.60	0.66	0.63

TRANSMISSION MODELS

Transmission model extraction results are presented in Table 34. `GLM-4.7` achieves the highest overall average F_1 (0.85), with strong performance across all three sub-tasks: Flagging (0.93), Counts (0.93), and Extraction (0.68). `DeepSeek-V3.2` ranks second overall (0.81), driven by notably high Counts performance (0.92), whilst `gpt-oss-120b` ranks last (0.75), held back by comparatively low Counts precision (0.52 overall). Lassa is the best-performing pathogen for all five models, with `GLM-4.7` achieving an overall average F_1 of 0.91 for that pathogen alone, including perfect Flagging and Counts scores. SARS is consistently the most challenging pathogen: Flagging F_1 ranges from 0.82 (`DeepSeek-V3.2`) to 0.87 (`Kimi-K2.5`), and Extraction F_1 from 0.59 to 0.66. These patterns are consistent with the field-level difficulties in transmission route and assumption extraction identified in Appendix I.

Table 34. **Model extraction metrics with model ablations.** Green and Red denote the best- and worst-performing pathogens for each model (in terms of F_1 score). **Average** denotes means across sub-tasks; **Overall** denotes means across pathogens. **Bold** indicates the best-performing model for each pathogen, and Underline indicates the second-best. P = precision; R = recall; F_1 = F1-Score.

Pathogen	Type	gpt-oss-120b			GPT-5.2			DeepSeek-V3.2			Kimi-K2.5			GLM-4.7		
		P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
Ebola	Flagging	0.92	0.92	0.92	0.89	0.90	0.89	0.87	0.86	0.86	0.93	0.93	0.93	0.95	0.94	**0.95**
	Counts	0.50	1.00	0.67	0.56	1.00	0.71	0.81	0.99	0.89	0.63	0.99	0.77	0.88	0.99	**0.93**
	Extraction	0.59	0.72	0.64	0.62	0.74	**0.66**	0.57	0.65	0.61	0.60	0.71	0.64	0.62	0.71	0.66
	Average	0.67	0.88	0.74	0.69	0.88	0.76	0.75	0.83	0.79	0.72	0.88	0.78	0.81	0.88	**0.84**
Lassa	Flagging	0.95	0.99	0.97	0.95	0.99	0.97	0.96	0.85	0.89	0.95	0.99	0.97	1.00	1.00	**1.00**
	Counts	0.60	1.00	0.75	0.60	1.00	0.75	1.00	1.00	**1.00**	0.75	1.00	0.86	1.00	1.00	**1.00**
	Extraction	0.68	0.73	0.70	0.68	0.79	0.71	0.79	0.78	**0.78**	0.70	0.78	0.73	0.73	0.77	0.74
	Average	0.74	0.91	0.81	0.74	0.92	0.81	0.92	0.88	0.89	0.80	0.92	0.85	0.91	0.92	**0.91**
SARS	Flagging	0.86	0.86	0.86	0.86	0.86	0.86	0.83	0.81	0.82	0.87	0.87	**0.87**	0.85	0.84	0.84
	Counts	0.49	0.97	0.65	0.49	1.00	0.66	0.70	1.00	0.82	0.67	1.00	0.81	0.76	1.00	**0.86**
	Extraction	0.60	0.71	0.64	0.61	0.73	0.64	0.55	0.64	0.59	0.63	0.74	**0.66**	0.59	0.68	0.62
	Average	0.65	0.85	0.72	0.65	0.86	0.72	0.70	0.82	0.74	0.72	0.87	**0.78**	0.73	0.84	0.77
Zika	Flagging	0.87	0.89	0.88	0.89	0.91	0.90	0.90	0.89	0.90	0.90	0.92	0.91	0.93	0.93	**0.93**
	Counts	0.48	0.98	0.65	0.61	1.00	0.76	0.97	0.97	**0.97**	0.72	0.97	0.83	0.88	0.97	0.93
	Extraction	0.66	0.78	0.70	0.67	0.78	0.69	0.59	0.64	0.61	0.67	0.77	0.70	0.69	0.76	**0.71**
	Average	0.67	0.88	0.74	0.72	0.90	0.78	0.82	0.84	0.83	0.76	0.89	0.81	0.83	0.89	**0.85**
Overall	Flagging	0.90	0.91	0.91	0.90	0.91	0.90	0.89	0.85	0.87	0.91	0.93	0.92	0.93	0.93	**0.93**
	Counts	0.52	0.99	0.68	0.56	1.00	0.72	0.87	0.99	0.92	0.69	0.99	0.81	0.88	0.99	**0.93**
	Extraction	0.63	0.74	0.67	0.64	0.76	0.67	0.63	0.68	0.65	0.65	0.75	**0.68**	0.66	0.73	0.68
	Average	0.68	0.88	0.75	0.70	0.89	0.77	0.80	0.84	0.81	0.75	0.89	0.81	0.82	0.88	**0.85**

OUTBREAKS

Outbreak extraction results, evaluated across Lassa and Zika, are shown in Table 35. GPT-5.2 achieves the highest overall average F_1 (0.77), followed closely by `Kimi-K2.5` (0.76), `DeepSeek-V3.2` (0.73), `GLM-4.7` (0.72), and `gpt-oss-120b` (0.70). Results diverge sharply between pathogens. Lassa Counts are strong across all models, ranging from F_1 0.77 (GPT-5.2) to 0.95 (`Kimi-K2.5`), and field-level Extraction is uniformly high (0.76 to 0.83). Zika Counts performance falls substantially for `gpt-oss-120b` (F_1 0.47) and `GLM-4.7` (0.52), whilst GPT-5.2 remains comparatively strong (0.83). A notable divergence in Flagging is also observed: `DeepSeek-V3.2` performs weakest on Lassa Flagging (F_1 0.62) whilst achieving the second-best result on Zika (0.68). These per-pathogen contrasts are consistent with the field-level analysis of suspected cases and epidemiological context fields reported in Appendix I.

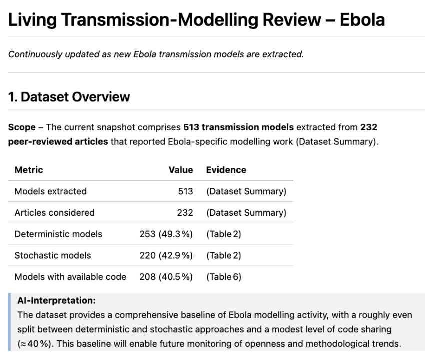
Table 35. **Outbreak extraction metrics with model ablations. Average** denotes means across sub-tasks; **Overall** denotes means across pathogens. Green and Red denote the best- and worst-performing pathogens for each model (in terms of F_1 score). **Bold** indicates the best-performing model for each pathogen, and Underline indicates the second-best. P = precision; R = recall; F_1 = F1-Score.

Pathogen	Type	gpt-oss-120b			GPT-5.2			DeepSeek-V3.2			Kimi-K2.5			GLM-4.7		
		P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$	P	R	$F1$
Lassa	Flagging	0.69	0.82	0.70	0.72	0.84	**0.74**	0.61	0.65	0.62	0.67	0.77	0.69	0.65	0.68	0.66
	Counts	0.83	1.00	0.91	0.62	1.00	0.77	0.83	1.00	0.91	0.90	1.00	**0.95**	1.00	0.86	0.92
	Extraction	0.85	0.73	0.77	0.84	0.79	0.81	0.75	0.78	0.76	0.84	0.83	**0.83**	0.83	0.76	0.78
	Average	0.79	0.85	0.80	0.73	0.88	0.78	0.73	0.81	0.77	0.80	0.87	**0.82**	0.83	0.76	0.79
Zika	Flagging	0.58	0.71	0.53	0.59	0.75	0.57	0.65	0.82	0.68	0.61	0.80	0.59	0.67	0.87	**0.70**
	Counts	0.49	0.45	0.47	0.76	0.92	**0.83**	0.68	0.61	0.64	0.72	0.88	0.79	0.84	0.38	0.52
	Extraction	0.84	0.78	0.80	0.88	0.87	**0.87**	0.69	0.81	0.73	0.71	0.78	0.73	0.74	0.78	0.76
	Average	0.64	0.65	0.60	0.75	0.84	**0.76**	0.68	0.74	0.69	0.68	0.82	0.70	0.75	0.68	0.66
Overall	Flagging	0.63	0.76	0.61	0.66	0.79	0.66	0.63	0.73	0.65	0.64	0.78	0.64	0.66	0.77	**0.68**
	Counts	0.66	0.72	0.69	0.69	0.96	0.80	0.76	0.80	0.78	0.81	0.94	**0.87**	0.92	0.62	0.72
	Extraction	0.85	0.76	0.79	0.86	0.83	**0.84**	0.72	0.79	0.75	0.78	0.81	0.78	0.78	0.77	0.77
	Average	0.71	0.75	0.70	0.74	0.86	**0.77**	0.70	0.78	0.73	0.74	0.84	0.76	0.79	0.72	0.72

K. Living Systematic Reviews with AgentSLR for 9 Priority Pathogens

Utilising AgentSLR on the data extracted corpus from previous stages, we generated living reviews for nine WHO priority pathogens: Marburg virus, Ebola virus, Lassa virus, SARS-CoV-1, Zika virus, MERS-CoV, Nipah virus, Rift Valley fever (RVF) virus, and Crimean–Congo haemorrhagic fever (CCHF) virus. Each review comprises two complementary documents (a transmission-modelling review synthesising extracted model characteristics, and an outbreak surveillance review aggregating historical outbreak data) alongside structured datasets and visualisations. While four of these pathogens (Ebola, Lassa, SARS, Zika) have been validated against PERG's expert annotations as described in Section 4.3, the remaining five represent preliminary syntheses for pathogens where PERG's systematic review process has not yet commenced or is in early stages.

Figure 8 presents excerpts from the Ebola living reviews, illustrating the structure and content of AgentSLR's outputs for a validated pathogen. The transmission-modelling review (Figure 8a) provides a quantitative overview of the 513 extracted models, including distributions across model architectures, stochasticity classifications, and code availability. The outbreak surveillance review (Figure 8b) synthesises 1,104 outbreak records spanning nearly six decades, with temporal coverage, geographic distribution, and detection methodology patterns presented through evidence-based descriptions paired with interpretive commentary blocks.


Living Transmission-Modelling Review – Ebola

Continuously updated as new Ebola transmission models are extracted.

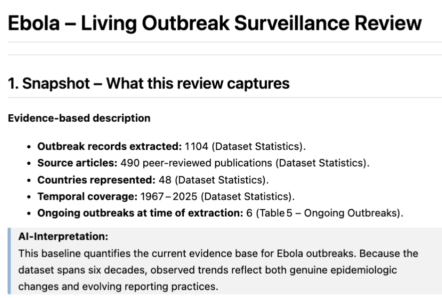
1. Dataset Overview

Scope – The current snapshot comprises **513 transmission models** extracted from **232 peer-reviewed articles** that reported Ebola-specific modelling work (Dataset Summary).

Metric	Value	Evidence
Models extracted	513	(Dataset Summary)
Articles considered	232	(Dataset Summary)
Deterministic models	253 (49.3 %)	(Table 2)
Stochastic models	220 (42.9 %)	(Table 2)
Models with available code	208 (40.5 %)	(Table 6)

AI-Interpretation:
The dataset provides a comprehensive baseline of Ebola modelling activity, with a roughly even split between deterministic and stochastic approaches and a modest level of code sharing (≈ 40 %). This baseline will enable future monitoring of openness and methodological trends.

(a) Transmission-Modelling Review excerpt showing dataset scope, model architecture distribution, and reproducibility indicators for 513 Ebola models extracted from 232 articles.


Ebola – Living Outbreak Surveillance Review

1. Snapshot – What this review captures

Evidence-based description

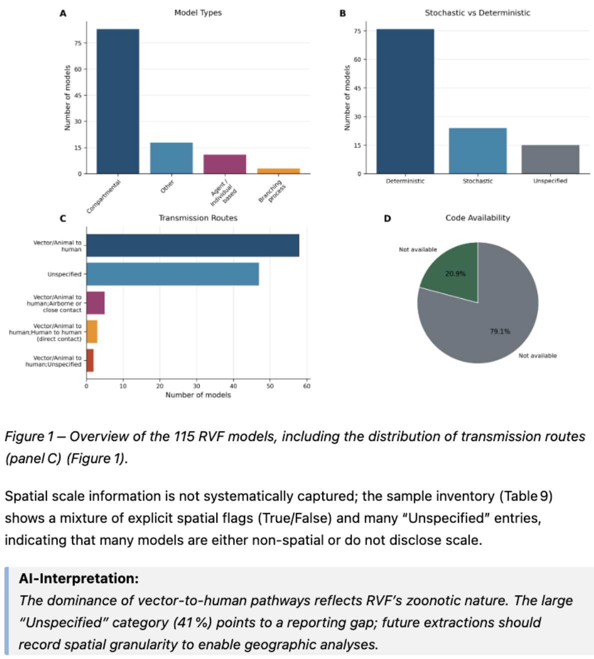
- **Outbreak records extracted:** 1104 (Dataset Statistics).
- **Source articles:** 490 peer-reviewed publications (Dataset Statistics).
- **Countries represented:** 48 (Dataset Statistics).
- **Temporal coverage:** 1967 – 2025 (Dataset Statistics).
- **Ongoing outbreaks at time of extraction:** 6 (Table 5 – Ongoing Outbreaks).

AI-Interpretation:
This baseline quantifies the current evidence base for Ebola outbreaks. Because the dataset spans six decades, observed trends reflect both genuine epidemiologic changes and evolving reporting practices.

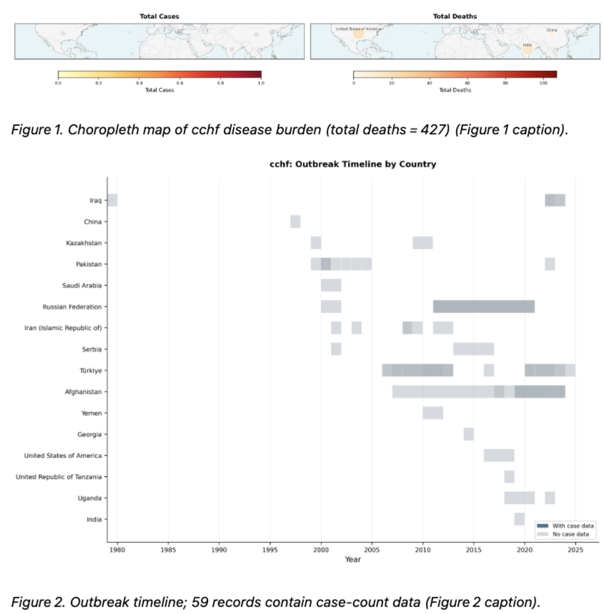
(b) Outbreak Surveillance Review excerpt presenting snapshot statistics for 1,104 outbreak records from 490 publications, covering temporal span 1967–2025 and 48 countries.

Figure 8. **Ebola living reviews generated by AgentSLR.** Both reviews follow a structured format: evidence-based descriptions citing supporting figures and tables, followed by interpretation blocks explicitly labelled as AI-generated synthesis. Ebola represents one of four pathogens validated against PERG expert annotations (Section 4.3).

For emerging or understudied pathogens, rapid synthesis of available evidence can inform outbreak preparedness even when comprehensive expert review remains infeasible. Figure 9 presents excerpts from RVF and CCHF reviews, two pathogens for which PERG has not yet initiated systematic screening. The RVF transmission-modelling review (Figure 9a) characterises 115 models extracted from the retrieved literature, revealing a predominance of compartmental architectures and vector-to-human transmission pathways consistent with RVF's arboviral ecology. The CCHF outbreak surveillance review (Figure 9b) maps 59 outbreak records with quantitative case data, identifying geographic clusters and temporal patterns across affected regions. While these syntheses lack the validation rigour applied to Ebola, Lassa, SARS, and Zika, they demonstrate AgentSLR's capacity to generate preliminary evidence summaries for resource allocation and hypothesis generation in under 48 hours of wall-clock time.



(a) RVF transmission-modelling review showing distribution of 115 extracted models across architecture types, stochasticity, transmission routes, and code availability.



(b) CCHF outbreak surveillance review presenting geographic burden (choropleth maps) and temporal distribution of 59 outbreaks with case-count data.

Figure 9. **Preliminary living reviews for pathogens without completed PERG validation.** RVF and CCHF represent pathogens for which PERG has not yet commenced systematic screening. These AgentSLR-generated reviews provide initial evidence synthesis for outbreak preparedness planning, though they lack the expert validation applied to the four evaluated pathogens (Ebola, Lassa, SARS, Zika).

Table 36 summarises the standardised artefact structure maintained across all pathogen reviews. Both transmission-modelling and outbreak surveillance reports follow consistent schemas: model reviews characterise architecture distributions, stochasticity classifications, transmission pathways, and reproducibility indicators, whilst outbreak reviews present temporal timelines, geographic burden maps, detection methodology breakdowns, and case-count summaries. This structural consistency enables direct cross-pathogen comparison and ensures that future updates (as literature accumulates or as PERG completes validation for additional pathogens) maintain compatibility with existing syntheses.

Table 36. **Key artefacts in AgentSLR living reviews.** Each pathogen generates two review types with consistent visualisation and evidence table structures. The text-based LLM uses manifests, with summary statistics of the figures to write its interpretation.

Type	Transmission-Modelling Review	Outbreak Surveillance Review
Figures	Model architecture distribution (compartmental, branching process, agent-based); Stochasticity classification; Transmission route breakdown; Code availability	Geographic burden (choropleth maps for cases and deaths); Outbreak timeline by country; Detection mode distribution
Tables	Model type counts and proportions; Deterministic vs stochastic breakdown; Transmission routes with sample sizes; Modelling assumptions; Intervention categories; Spatial scale indicators; Code availability and language	Outbreak source categories; Detection methodology breakdown; Ongoing outbreaks at extraction; Case burden stratified by confirmation status; Sex disaggregation where reported

L. Extended Expert Validation Results

Six epidemiology researchers contributed to our validation survey. We collected 62 submissions for parameters, 50 for models, and 31 for outbreaks. Table 37 reports all metrics collected from the survey. The main text reports the aggregate statistics (the 'Overall' rows for each data type) in the first two columns columns of Figure 4.

Table 37. **Extended expert validation results.** Results are reported as expert-rated flagging precision and expert-rated extraction accuracy. Within each section, rows are ordered from overall scores to subgroup scores and then field-level scores.

Item	Score
Parameters	
Overall — Flagging precision	0.66
Overall — Extraction accuracy	0.77
Precision by class	
Attack rate	0.25
Growth rate	1.00
Human delay	0.62
Reproduction number	1.00
Seroprevalence	0.50
Severity	0.57
Accuracy by group	
Value	0.89
Uncertainty	0.76
Population	0.59
Aggregation	0.83
Value fields	
Value	0.81
Unit	0.96
Type	0.88
Bounds	0.79
Value type	0.90
Statistical approach	0.97
Uncertainty fields	
Single-type uncertainty	0.88
Paired uncertainty	0.84
Distribution type	0.57
Population fields	
Sample type	0.74
Population group	0.49
Sample size	0.66
Sex	0.50
Age range	0.58
Countries	0.82
Locations	0.71
Method moment value	0.23
Aggregation fields	
Aggregation	0.83
Models	
Overall — Flagging precision	0.40
Overall — Extraction accuracy	0.83
Field accuracy	
Model type	0.89
Compartmental type	0.89
Stochastic or deterministic	0.70
Theoretical model	0.84
Outbreaks	
Overall — Flagging precision	0.61
Overall — Extraction accuracy	0.80

Continued on next page

Table 37 continued from previous page

Item	Score
Accuracy by group	
Temporal	0.62
Geographical	0.87
Case burden	0.85
Epidemiological	0.85
Temporal fields	
Start year	0.84
Start month	0.70
Start day	0.62
End year	0.50
End month	0.60
End day	0.56
Duration in months	0.50
Geographical fields	
Country	0.95
Location	0.80
Case burden fields	
Confirmed cases	0.88
Suspected cases	0.64
Asymptomatic cases	1.00
Severe cases	1.00
Deaths	0.71
Epidemiological fields	
Mode of detection	0.82
Pre-outbreak status	0.82
Asymptomatic transmission described	0.89

For flagging (sub-task) precision, models and outbreaks are reported only at the overall level (as in the main text). For parameters, precision is averaged over flagging decisions made for each parameter class. The random subsample of articles assigned gave six relevant parameter classes: attack rate, growth rate, human delay, reproduction number, seroprevalence, and severity. The remaining two parameter classes, mutation rate and relative contribution, were absent from the sample. Since there is a flagging decision made for each parameter class on each article, each parameter class-level precision is calculated over the same sample size ($N = 62$).

For extraction accuracy, the aggregate statistics are normalised over groups of similar fields. For example, outbreaks have clusters of fields related to temporal features (start date, end date, and duration), geographical features (country and location), case burden (case counts and fatalities) and epidemiological factors (mode of detection, status pre-outbreak, and asymptomatic transmission). We normalise at the group level to treat each aspect of the extraction as equally important, in order to avoid overemphasising groups with larger numbers of metadata fields. We omit group-level normalisation for models owing to the smaller number of validated fields.

The disaggregated statistics reveal findings that are masked in the average statistics. For example, among parameter classes, AgentSLR performs worst on flagging attack rate (experts reported several instances where the system confused attack rate with seroprevalence information). At the field level, AgentSLR struggles the most with understanding parameter population context and with the temporal outbreak features (group accuracy 0.62). Parameter population fields are multiple-choice selections with many options (see Table 12), and these designations often have specific interpretations in epidemiology. For example, "persons under investigation" is a population group of patients exhibiting clinical and epidemiological risk factors, a definition that an LLM may struggle to apply consistently in different article contexts.

M. The PERG Review Pipeline (Human Reference Workflow)

The *Pathogen Epidemiology Review Group (PERG)* is an expert-led effort (started in 2019) whose goal is to maintain a definitive, curated source of epidemiological parameters for pathogens prioritised for epidemic preparedness. In practice, PERG delivers this through systematic literature reviews and meta-analyses targeting the WHO priority pathogens, with the explicit aim of supporting outbreak response and modelling when time is short and parameter choices matter.

The scope is defined by the WHO priority pathogens framing: diseases that "pose the greatest public health risk due to their epidemic potential and/or whether there is no or insufficient countermeasures." Examples highlighted in PERG onboarding include CCHF virus, Ebola virus, Marburg virus, Lassa virus, Middle East respiratory syndrome coronavirus (MERS-CoV), Severe Acute Respiratory Syndrome coronavirus 1 (SARS-CoV-1), Nipah virus, Rift Valley fever, and Zika virus.

PERG's workflow is end-to-end: it starts from a protocolised literature search, then moves through screening (title & abstract, then full text), structured extraction into REDCap (including quality-assessment fields guided by the PERG wiki), meta-analysis, and finally the write-up of a review that can be used by modellers and public health teams.

Step 1: Paper search (protocol-driven, pathogen-specific)

PERG begins from a registered systematic review protocol (PROSPERO ID: CRD42023393345), and uses a standardised query template that is then tailored to each pathogen. The core idea is to search broadly across the epidemiological concepts that tend to matter during outbreak response: transmission and epidemiology terms, transmission modelling (with explicit exclusion of imaging-related "model" matches), severity outcomes (e.g. CFR), key delays (e.g. incubation period, serial interval, generation time), transmission heterogeneity and superspreading/overdispersion, transmissibility measures (e.g. growth rate and reproduction numbers), serology/serosurveys, evolutionary signals (mutation/substitution/evolution), outbreak/cluster terminology and risk factors. The query is written with wildcards to capture term variants, and then adjusted where needed to avoid cross-contamination with neighbouring literatures (for example, excluding SARS-CoV-2 when the target is SARS-CoV-1).

Step 2: Title and abstract screening (broad triage against explicit criteria)

The first screening pass is based on titles and abstracts. The emphasis here is not on perfect specificity, but on ensuring the pool remains wide enough to avoid missing relevant evidence that is only clearly described later in the paper. PERG's **inclusion criteria** are simple but concrete: studies must be English-language, peer-reviewed original research (systematic reviews and meta-analyses are flagged rather than treated as primary extraction targets), and must involve human data. A paper is kept if it contains *any one* of several types of useful information, including: quantitative descriptions of a human outbreak (size, year, location, duration, spatial scale), a mathematical or statistical model of transmission, estimates of key transmission or timing quantities (e.g., R, R_0, R_t, growth rate, generation time, serial interval, incubation or latent period, other delays), severity metrics (CFR, attack rate), evolutionary rates, overdispersion/superspreading, risk factors (together with the measure), seroprevalence, relative contributions of human-to-human vs zoonotic transmission, and, where relevant, vector-related quantities such as mosquito delays or mosquito reproduction numbers.

PERG's **exclusion criteria** are equally explicit: non-English items; posters, conference proceedings, correspondence, and abstract-only records; in-vitro-only studies; solely animal studies (unless the paper provides clearly relevant transmission quantities); and small case studies with fewer than 10 cases.

Step 3: Full-text review (confirm "extractability")

Articles passing abstract screening move to full-text review. PERG applies the same conceptual criteria, but with a different mindset: reviewers scan the entire paper to confirm that there is *something extractable*, i.e. not just that the topic is on-target. Importantly, PERG explicitly runs both title and abstract screening and this stage with **two reviewers**, reflecting the goal of consistency and defensible inclusion decisions when judgement calls are required.

Step 4: Parameter extraction (read, highlight, enter structured fields)

Once a paper is included, PERG's extraction process is deliberately hands-on. Reviewers (i) check which papers they have been assigned, (ii) download and read the PDF, highlighting everything they may want to extract as they go, and then (iii) enter the extracted information into a REDCap web database (PERG maintains pathogen-specific REDCap projects).

PERG structures extraction into four broad blocks:

- *Article metadata.* Basic bibliographic information such as title, DOI, journal, and related identifiers are recorded.
- *What the paper contains:* outbreaks, models, parameters. PERG extracts (i) outbreak descriptions where present, (ii) mathematical models of transmission (these are not limited to SIR-type models; they can be theoretical and not necessarily fitted to data), and (iii) epidemiological parameter estimates. Parameter families include genomic/evolutionary quantities (mutation/evolution rates), reproduction numbers (R_0, R_t, and human-only or vector-related variants where relevant), human delays (serial interval, incubation period, time-to-death, etc.), severity (CFR/IFR), seroprevalence (e.g. IgG/IgM markers), risk factors (with attention to whether effects are statistically significant and adjusted), relative contributions (human-to-human vs animal-to-human), attack rates (including secondary attack rates), and overdispersion (e.g. the negative binomial k parameter).
- *Associated context for interpretation.* PERG captures the contextual details that make parameter estimates comparable (or not): sex (male/female/both/unspecified), sample size, setting (general population vs hospital), subgroup (children, pregnant, etc.), age ranges, country and more specific location, study start/end dates, and whether the study was conducted before/mid/after an outbreak.
- *Structured outbreak fields (when applicable).* In addition to "is there an outbreak?", PERG-style extraction treats outbreaks as structured entities. In our draft's PERG-aligned outbreak guidance, outbreak characteristics include temporal bounds (start/end day/month/year; whether ongoing), geographic scope (country plus sub-location), outbreak source, mode of detection, case definition method, case counts by confirmation status (confirmed/probable/suspected/unspecified), asymptomatic and severe cases when reported, deaths, and (when available) demographic breakdown such as sex-disaggregated counts. A key principle is that these values are extracted *as stated in the paper*, without calculating missing quantities or inferring unreported fields.

Across all extraction types, PERG points reviewers to the PERG wiki for "how to extract this specific thing" guidance, so that extraction decisions remain consistent across pathogens and across reviewers.

Step 5: Meta-analysis

After extraction and quality assessment, PERG moves into synthesis and reporting. PERG maintains shared tooling for priority pathogens, including codebases that step through cleaning the extracted database, transforming quantities into a common format where needed, performing meta-analysis, and producing plots and summary tables. These outputs feed directly into the final PERG systematic review and meta-analysis write-up.

Step 6: Write-up

The final stage is to turn the extracted REDCap database and the meta-analysis outputs into a PERG review that can be used in practice. In PERG, meta-analysis is implemented through shared, pathogen-focused tooling (the `priority-pathogens` and `epireview` codebases), which steps through cleaning and transforming the extracted data, running the statistical synthesis, and producing the figures and summary tables. These tables and plots then provide the backbone of the manuscript: the review documents what evidence was found for each parameter family (and in what contexts), presents the quantitative summaries produced by the meta-analysis, and translates them into a curated resource for outbreak modelling and public health decision-making. In PERG's framing, this write-up is not just a paper draft: it is the mechanism by which extracted parameters become a stable, citable reference for the WHO priority pathogens, with the longer-term aim of supporting an evolving "live" resource as evidence accumulates.

N. AgentSLR Annotation Tool (Beta)

This section documents the AgentSLR annotation and validation interface, a beta-stage prototype designed to facilitate systematic literature reviews (SLRs) through the integration of LLM-assisted information extraction and expert-led verification.

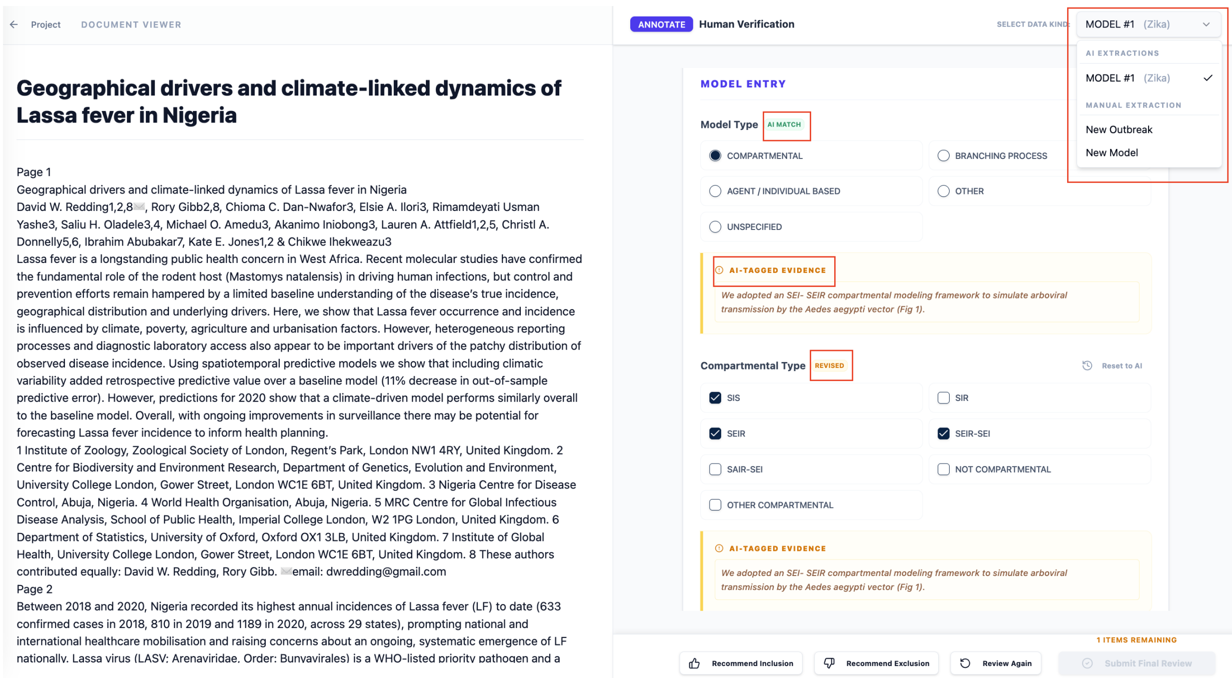


Figure 10. **The AgentSLR extraction and validation tool interface.** The dual-pane view presents the source document (**left**) alongside structured extraction fields (**right**). AI-predicted entries are pre-filled and accompanied by highlighted, AI-tagged evidence excerpts from the manuscript. Reviewers can accept, revise, or reject individual fields, with validation status indicators (e.g. "AI Match" or "Revised") reflecting whether human intervention was required.

N.1. System Architecture and Core Functionality

The AgentSLR annotation tool provides an interactive environment for technical validation of automated information extractions. The system utilises data gathered by LLMs equipped with structured tool-calling to identify and parse epidemiological parameters, transmission models, and outbreak characteristics. To ensure transparency and auditability, the tool implements a provenance layer that maps every extracted value to specific textual excerpts (AI-tagged evidence) within the source article.

N.2. User Interface Design

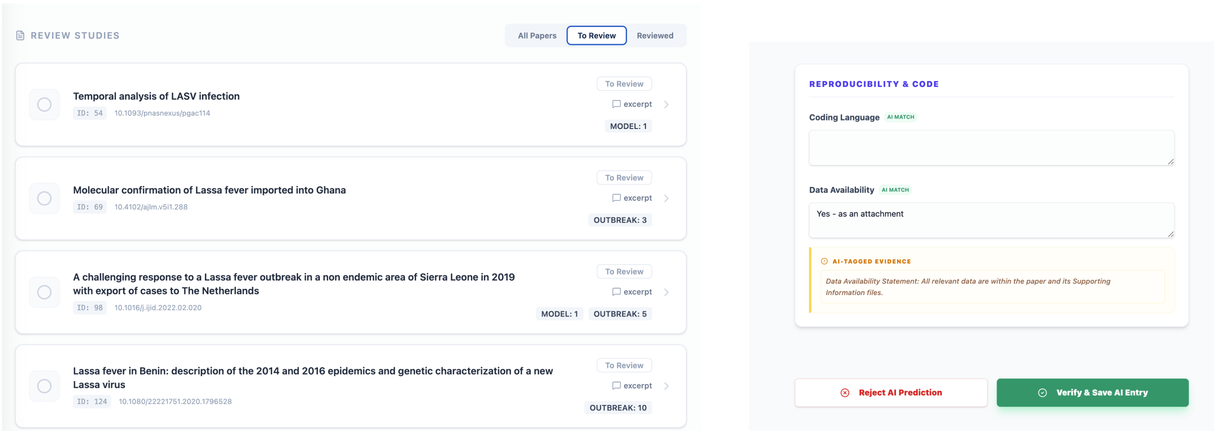
The interface is optimised for high-throughput expert review via a dual-panel architecture:

- *Document Viewer (Left Panel):* Provides the original article text or rendered PDF, ensuring reviewers can verify the context of any extracted data point without context switching.
- *Verification Interface (Right Panel):* Displays a form-based view of pre-filled fields generated by the AgentSLR pipeline. The extraction schema is dynamic, adapting based on the identified content type, such as compartmental model variables or spatio-temporal outbreak data.

N.3. Human-in-the-Loop Validation

The framework enforces a human-in-the-loop (HITL) protocol where automated extractions are audited by subject matter experts before being finalised for evidence synthesis. Within the interface, reviewers perform the following actions:

- *Verify:* Confirm the accuracy of the AI-extracted value and its linked evidence.



(a) Tool Management Dashboard

(b) Verification and Submission

Figure 11. **Review management and submission tracking interfaces.** (a) The study review list displays papers awaiting expert validation, along with associated model and outbreak counts and direct links to extracted excerpts. (b) The verification view presents finalised AI-assisted extractions, enabling reviewers to explicitly reject predictions or verify and save corrected entries, which are then recorded for downstream quality control and system evaluation.

- *Modify:* Correct extraction errors or refine data granularity; modified entries are flagged as "Revised" to facilitate system error analysis.
- *Reject:* Entirely dismiss false positive extractions that do not meet inclusion criteria.

N.4. Current Status and Field Testing

The tool is currently in a beta development phase, with pilot testing conducted by epidemiologists focusing on WHO priority pathogens. While the current pilot utilises epidemiology-specific schemas, the architecture is designed to be domain-agnostic and can be adapted through schema reconfiguration and expert consultation.

Planned field testing is aligned with the Pathogen Epidemiology Review Group (PERG) workflow for remaining priority pathogens. This testing will utilise standardised extraction schemas for parameter, model, and outbreak data, and specifically target systematic reviews for CCHF virus and Rift Valley fever virus.

N.5. Transparency and Reproducibility

By maintaining a persistent link between the structured database and the source text, AgentSLR ensures that synthesised reports can be fully disaggregated. This audit trail is critical for scientific reproducibility, allowing researchers to trace every reported parameter, model and outbreak back to its exact location in the primary literature.