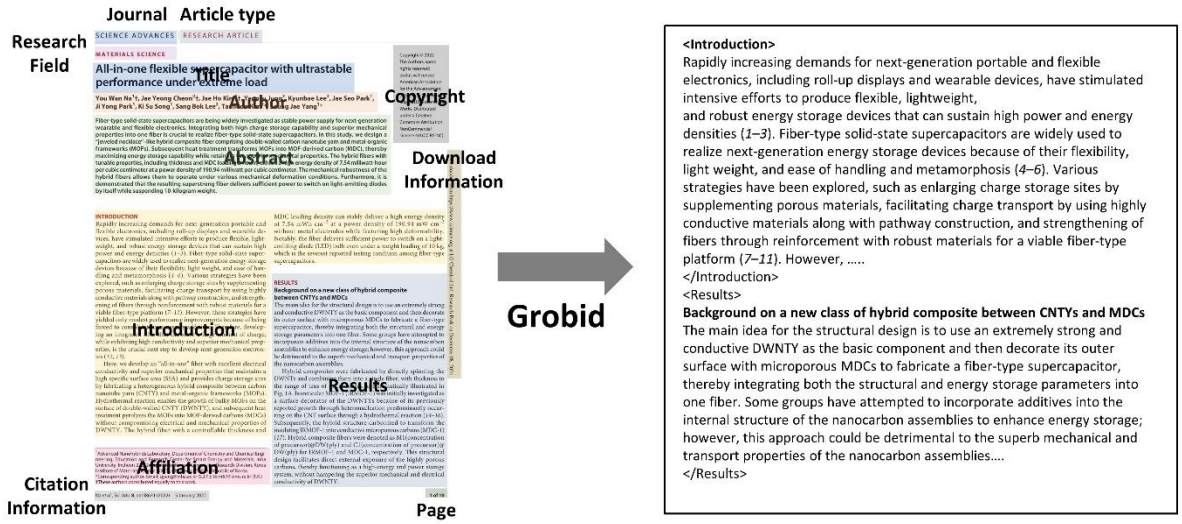
Supporting Information

for

Advanced Scientific Information Mining Using LLM-Driven Approaches in Layered Cathode Materials for Sodium-ion Batteries

Youwan Na[1], Jeffrey J. Kim[1], Chanhyoung Park, Jaewon Hwang, Changgi Kim, Hokyung Lee, and Jehoon Lee

Technology Planning Department, LG Chem Ltd., 30 Magokjungang 10-ro, Gangseo-gu, Seoul, 07796, Republic of Korea



Supporting Information 1. Document preprocessing methodology using GROBID (GeneRation Of BIbliographic Data) system. The preprocessing step employs GROBID for converting unstructured scientific PDF documents into structured XML/TEI formats. GROBID excels at identifying and extracting multiple document components: document metadata (titles, author information, affiliations), structural elements (headers, abstracts, sections, subsections), visual components (figures, tables), and bibliographic information (citations, references). The system implements a comprehensive labeling mechanism for fine-grained document structure analysis, incorporating advanced features such as language identification, sentence segmentation, and adaptive handling of various academic writing styles. Notably implemented at major scientific repositories including ResearchGate and CERN, GROBID enables reliable transformation of unstructured scientific documents into machine-readable formats suitable for subsequent analysis.

Electrode

<Instruction>
From the given context, extract and organize the cathode electrode composition information used in electrochemical analysis according to the following format:Active Material(Content/ratio (include units: wt%)), Binder Information(Type of binder, Binder content (include units: wt%)), Conductive Agent Information(Type of conductive agent, Conductive agent content (include units: wt%)). If there is no specified answer, answer "Not specified"

Electrolyte

<Instruction>
From the given context, extract and organize the electrolyte information used in electrochemical analysis according to the following format) Solvent Information(List all solvents used as a abbreviation, Mixing ratio of solvents (if applicable)), Salt Information(Chemical formula or name of the salt, Salt concentration (include units: M, mol/L, wt%, etc.)), Additive Information(Chemical formula or name of additives, Additive content (include units: vol%, wt%, etc.)). If there is no specified answer, answer "Not specified"
</Instruction>

Material

<Instruction>
From the provided context, list ONLY the layered cathode materials that were directly synthesized in this study (not referenced from other research).

For each composition:

1. Write the complete chemical formula with proper notation:
- Use correct elemental symbols and stoichiometry
- Include proper subscripts
- Express any phase information (e.g., P2-, O3-) if specified
- Express as a target composition. Not from ICP

2. For fractional values in compositions:
- Convert to decimal form
- Round to 2 significant figures

3. If multiple phases exist for the same composition:
- Group them using square brackets [ ]
- Include phase designation for each
Example: [P2-Na0.67MnO2, O3-Na0.67MnO2]
</Instruction>

Chain of Thought

<CoT>
Follow these steps:
1. First, carefully read and understand the given context
2. Break down the question into key components
3. Think through the reasoning process step by step
4. Provide your final answer based on your reasoning

Reasoning:
1. [First step of your thought process]
2. [Next step]
3. [Final step]
</CoT>

Supporting Information 2**.** Engineered prompts and Chain-of-Thought (CoT) methodology for extracting scientific data. The prompt engineering was specifically designed to extract three key categories of information from sodium-ion battery research literature: (1) electrode preparation conditions, (2) electrolyte specifications used in electrochemical evaluations, and (3) compositional details of layered metal oxide cathode materials. The prompts were structured to facilitate systematic extraction through Chain-of-Thought reasoning, enabling the language model to logically process and identify relevant technical information from complex scientific texts. This approach ensures comprehensive and accurate extraction of materials science parameters critical for sodium-ion battery research.

Supporting Information 3. Evaluation Methods

3.1 Confusion Matrix

The confusion matrix serves as a fundamental tool for evaluating model classification performance through multiple dimensions. In binary classification scenarios, it enables quantitative analysis of model prediction accuracy for both positive and negative cases. This matrix facilitates the identification of misclassification patterns and potential prediction biases toward specific classes. Notably, it provides objective performance evaluation even in cases of data imbalance, making it instrumental in determining model improvement strategies.

Key Components:

True Positive (TP): Correct identification of positive cases

True Negative (TN): Correct identification of negative cases

False Positive (FP): Type I error - negative cases incorrectly classified as positive

False Negative (FN): Type II error - positive cases incorrectly classified as negative

Performance Metrics:

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP})$$

$$\text{Recall} = \text{TP} / (\text{TP} + \text{FN})$$

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})$$

$$\text{F1 Score} = 2 \times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

3.2 Economic Efficiency

Economic efficiency assessment incorporates four quantitative metrics: the average number of tokens used for input processing, the average token count for output generation, the processing duration required per question in seconds, and the operational cost measured in USD per 100 questions processed.

3.3 Reliability

System reliability is evaluated through two key measures: the consistency score derived from the average of 5 repeated trials, and the self-confidence metric calculated from the mean confidence scores.

3.4 RAGAS Framework

The RAGAS (Retrieval Augmented Generation Assessment) framework provides an automated, reference-free approach for evaluating language model outputs, with particular emphasis on hallucination detection. The framework evaluates three critical aspects:

Key Metrics:

Semantic Similarity: The metric evaluates the semantic alignment between generated and reference content using a cross-encoder model, producing a normalized score between 0 and 1, where higher scores represent stronger semantic correspondence between outputs.

Faithfulness: The metric quantifies factual consistency by decomposing responses into discrete claims and validating them against the source material, yielding a score between 0 and 1 that represents the proportion of verifiable claims.

The implementation follows standardized RAGAS protocols to ensure systematic and objective evaluation of response quality and reliability.

Metrics	Detail	Chunk size							
		100	300	500	1000	2000	3000	5000	Full_context
Confusion Metrics	Precision	0.8673	0.8991	0.8984	0.9289	0.9335	0.9231	0.9307	0.9089
	Recall	0.4067	0.7610	0.7880	0.8716	0.9111	0.9029	0.9104	0.8682
	F1-score	0.5537	0.8243	0.8395	0.8994	0.9221	0.9129	0.9204	0.8880
	Accuracy	0.3829	0.7011	0.7235	0.8171	0.8555	0.8397	0.8526	0.7986
Economicefficiency	Token usage(input, avg)	590	797	974	1346	2085	2852	4603	7805
	Token usage(output, avg)	81	102	100	104	107	109	109	145
	Processing time(s / 1 question)	0.8582	0.7756	0.8535	0.7864	0.7473	0.8653	1.1230	0.7610
	Cost($/100 Questions)	0.2283	0.3012	0.3440	0.4403	0.6287	0.8223	1.2601	2.0958
Reliability	Consistency(5 times, avg)	0.9750	0.9917	0.9854	0.9646	0.9896	0.9729	0.9875	0.8667
	Self-confidence(avg)	0.7376	0.8885	0.9001	0.9112	0.9241	0.9258	0.9252	0.8598
RAGAS	SemanticSimilarity	0.9155	0.9428	0.9427	0.9502	0.9529	0.9527	0.9518	0.949
	Faithfullness	0.7808	0.7912	0.8577	0.8578	0.8717	0.8822	0.9125	0.8727

Supporting Information 4. GPT-4o Response Evaluation Across Different Context Lengths

Metrics	Detail	Chunk size							
		100	300	500	1000	2000	3000	5000	Full_context
Confusion Metrics	Precision	0.7647	0.8389	0.7995	0.8514	0.8766	0.8553	0.8545	0.9067
	Recall	0.3723	0.7569	0.7656	0.8535	0.8900	0.8532	0.8372	0.8706
	F1-score	0.5008	0.7958	0.7822	0.8525	0.8832	0.8542	0.8458	0.8883
	Accuracy	0.3340	0.6608	0.6423	0.7429	0.7909	0.7456	0.7327	0.7991
Economicefficiency	Token usage(input, avg)	590	797	974	1353	2085	2852	4603	7805
	Token usage(output, avg)	102	129	125	119	118	116	114	145
	Processing time(s / 1 question)	0.6542	0.8012	0.8810	0.9744	0.8524	0.8447	0.8010	0.8069
	Cost($/100 Questions)	0.0150	0.0197	0.0221	0.0275	0.0384	0.0498	0.0759	0.1258
Reliability	Consistency(5 times, avg)	0.8896	0.9042	0.9000	0.9063	0.9500	0.9354	0.9167	0.8854
	Self-confidence(avg)	0.4979	0.7595	0.8019	0.8464	0.8794	0.8730	0.8738	0.8497
RAGAS	SemanticSimilarity	0.9181	0.941	0.9405	0.9478	0.9496	0.9479	0.9461	0.9479
	Faithfullness	0.7128	0.7838	0.8199	0.8327	0.8554	0.8322	0.865	0.863

Supporting Information 5. GPT-4o-mini Response Evaluation Across Different Context Lengths

Metrics	Detail	Chunk size							
		100	300	500	1000	2000	3000	5000	Full_context
Confusion Metrics	Precision	0.7676	0.7958	0.8116	0.8508	0.8635	0.8329	0.8453	0.9122
	Recall	0.3413	0.6848	0.7198	0.7979	0.8333	0.8089	0.8270	0.8532
	F1-score	0.4725	0.7361	0.7629	0.8235	0.8482	0.8207	0.8361	0.8817
	Accuracy	0.3094	0.5824	0.6167	0.7000	0.7363	0.6959	0.7183	0.7885
Economicefficiency	Token usage(input, avg)	590	791	959	1322	2081	2845	4587	7805
	Token usage(output, avg)	77	112	102	101	106	108	110	145
	Processing time(s / 1 question)	0.3792	0.5172	0.4415	0.4225	0.5154	0.5004	0.5529	0.7994
	Cost($/100 Questions)	0.0135	0.0186	0.0205	0.0259	0.0376	0.0491	0.0754	0.1258
Reliability	Consistency(5 times, avg)	0.8458	0.8125	0.8092	0.8245	0.8427	0.8438	0.8451	0.8958
	Self-confidence(avg)	0.9121	0.9223	0.9171	0.9178	0.9410	0.9477	0.9476	0.8503
RAGAS	SemanticSimilarity	0.9042	0.9305	0.9325	0.9455	0.9476	0.9443	0.9434	0.9479
	Faithfullness	0.6793	0.7035	0.7888	0.8098	0.8115	0.8235	0.8551	0.8734

Supporting Information 6. GPT-3.5-turbo Response Evaluation Across Different Context Lengths

Metrics	Detail	RAG Method(Chunk size = 2,000, model = gpt-4o)				
		Naïve-RAG	HyDE	RAG-Fusion	ToC	Self-RAG
Confusion Metrics	Precision	0.9335	0.9409	0.9370	0.9481	0.9441
	Recall	0.9111	0.9044	0.9061	0.9264	0.9126
	F1-score	0.9221	0.9223	0.9213	0.9371	0.9281
	Accuracy	0.8555	0.8557	0.8541	0.8816	0.8659
Economicefficiency	Token usage(input, avg)	2085	2419	2459	3831	2361
	Token usage(output, avg)	107	245	153	297	135
	Processing time(s / 1 question)	0.7473	2.3102	1.8000	2.7831	3.44
	Cost($/100 Questions)	0.6287	0.8500	0.7677	1.2556	0.7253
Reliability	Consistency(5 times, avg)	0.9896	0.9556	0.9253	0.9474	0.9049
	Self-confidence(avg)	0.9241	0.9209	0.9206	0.9219	0.9252
RAGAS	SemanticSimilarity	0.9529	0.9530	0.9499	0.9543	0.9514
	Faithfullness	0.8717	0.7859	0.8165	0.9005	0.7795

Supporting Information 7. Performance Analysis of Advanced RAG Techniques for Scientific Information Extraction from Research Literature