MATAGENT: A HUMAN-IN-THE-LOOP MULTI-AGENT LLM FRAMEWORK FOR ACCELERATING THE MATE-RIAL SCIENCE DISCOVERY CYCLE

Adib Bazgir & Yuwen Zhang *

Rama chandra Praneeth Madugula

Department of Mechanical and Aerospace Engineering University of Missouri-Columbia Columbia, MO 65211, USA {abwbw, zhangyu}@missouri.edu Department of Mechanical Engineering New York University New York, NY 10012 {rm6057}@nyu.edu

Abstract

The automation of materials science research through multi-agent large language models (LLMs) offers a transformative approach to accelerating discovery, optimizing experimentation, and enhancing data-driven decision-making. This study employs an LLM framework, called as MatAgent, across six key areas: material property prediction, hypothesis generation, experimental data analysis, highperformance alloy and polymer discovery, data-driven experimentation, and literature review automation. Machine learning models successfully predicted material properties, generated novel material hypotheses, analyzed experimental data, and optimized material compositions, significantly improving efficiency and accuracy. AI-driven methodologies enabled rapid screening of high-performance alloys and polymers, predictive modeling of concrete strength, and automated literature synthesis in perovskite solar cell research. The results demonstrate that MatAgent can revolutionize materials science by reducing research time, enhancing reproducibility, and paving the way for autonomous laboratories capable of AI-guided discovery and real-time adaptation. All corresponding codes and datasets related to this study are open-sourced in the GitHub repository available at https://github.com/adibgpt/MatAgent.

1 INTRODUCTION

Materials science explores the interplay between structure, properties, and performance to develop novel materials for applications ranging from aerospace alloys and high-entropy compounds to flexible polymers and clean-energy catalysts. As the universe of potential compounds expands, the field faces both significant opportunities and challenges: the need for advanced synthesis, characterization, and analysis methods to uncover subtle structure-property correlations. Traditionally, materials development has followed an iterative process, beginning with literature review and hypothesis generation, followed by experimental design, synthesis, characterization, and data interpretation, which, although proven, is time-consuming, expensive, and limited in the number of hypotheses that can be tested simultaneously. The exponential growth of scientific literature (Ping Ong, 2019) further intensifies the demand for innovative strategies in knowledge extraction, curation, and exploitation. Materials informatics has emerged to address traditional materials science bottlenecks by leveraging machine learning and data analytics to accelerate discovery (Pilania, 2021) (Ferguson & Brown, 2022). Early methods involved feature engineering, where experts manually selected descriptors such as elemental properties, crystal structures, or thermodynamic data to train ML models that predict material properties (Ward et al., 2016). These approaches led to significant breakthroughs, including the rapid identification of metallic glasses (Amigo et al., 2023) and high-throughput virtual screening for battery and fuel cell materials (Jain et al., 2013). However, despite these advances, the workflows remained partially manual, relying heavily on expert knowledge for descriptor selection, literature review, and experimental design. Concurrently, robotic laboratories and automation platforms began handling routine tasks like synthesis, mixing, heating, and measurement

^{*}Corresponding Author: Yuwen Zhang at zhangyu@missouri.edu.

with minimal human oversight (Abolhasani & Kumacheva, 2023) (Szymanski et al., 2021). Yet, further acceleration demands an "intelligent agent" capable of parsing scientific rationale, generating protocols, interpreting results, and refining hypotheses, a role that modern artificial intelligence, particularly Large Language Models, is well poised to fulfill. These LLMs were propelled by break-throughs in natural language processing, including architectures such as Transformers (Vaswani et al., 2017), and pre-trained models such as BERT (Kenton & Toutanova, 2019) and GPT-series models (Brown et al., 2020). Domain-specific adaptations such as SciBERT (Beltagy et al., 2019) and PubMedBERT (Gu et al., 2021) further demonstrate how tailored training can significantly enhance performance in scientific contexts.

Materials science involves diverse textual data from peer-reviewed articles and patents to technical datasheets and informal lab notes. A fine-tuned LLM for materials science can integrate these heterogeneous sources to create a unified knowledge base that not only identifies relevant materials and properties but also suggests new research directions. For instance, combining a ceramic's sintering temperature from a handbook with its photocatalytic performance data from a recent article can yield deeper insights. While LLMs are adept at summarizing text, generating literature reviews, and answering domain-specific queries, advanced materials discovery pipelines require additional capabilities. These include knowledge extraction to retrieve numerical data such as band gaps and doping concentrations (Lei et al., 2024); hypothesis generation to propose novel compounds or chemical substitutions (Xiong et al., 2024); design of experiments using techniques like Bayesian optimization and reinforcement learning; automated lab control to convert experimental protocols into precise instructions for robotic systems (Abolhasani & Kumacheva, 2023); and result interpretation to analyze data from methods like X-ray diffraction or electron microscopy for hypothesis validation (Roos et al., 2009). Multi-agent LLM architectures extend single-agent systems by coordinating specialized agents, some based on language models and others employing different machine learning approaches, via an orchestration layer that converts one agent's output into another's input (Guo et al., 2024). This setup resembles a multidisciplinary research team where experts in simulation, synthesis, hardware, and data analysis communicate using natural or structured language, ensuring each area's expertise is fully utilized. Although these multi-agent systems for materials science are still emerging, advances in agent-based experimental orchestration (Reymond, 2015) (Häse et al., 2018), large-scale textual data mining for materials (Weston et al., 2019), and modular AI frameworks in self-driving labs (Coley et al., 2019a) (Coley et al., 2019b) are rapidly converging. In these frameworks, agents specialize in tasks such as proposing new compounds, scheduling experiments, and interpreting measurement data, while large-scale data mining extracts entities and relationships from scientific literature to update a dynamic knowledge base. Meanwhile, modular AI in self-driving labs translates these insights into automated protocols for real or simulated experimental setups, fostering a more adaptive and efficient research cycle.

Recent advances in materials informatics leverage multi-agent AI systems and LLMs to accelerate materials discovery. AtomAgents (Ghafarollahi & Buehler, 2024a) integrates knowledge retrieval, physics-based simulations, and multi-modal data fusion to autonomously design high-performance alloys. Similarly, SciAgents (Ghafarollahi & Buehler, 2024b) employs multi-agent intelligent graph reasoning, using ontological knowledge graphs and LLMs to uncover hidden correlations, generate hypotheses, and refine research ideas. Domain-specific LLMs further enhance AI-driven materials research. MatChat (Chen et al., 2023) fine-tunes LLaMA-2 for inorganic synthesis pathway prediction, while LLaMat (Mishra et al., 2024) focuses on named entity recognition, relation extraction, and structured knowledge graph construction. MatSci-LLMs (Miret & Krishnan, 2024) propose a six-step AI-driven research pipeline, integrating data retrieval, simulation, and automated experimentation, bridging LLMs with real-world materials design. Further, the 2024 LLM Hackathon for Materials Science and Chemistry (Zimmermann et al., 2024) explored integrating LLMs with robotics, high-throughput screening, and active learning. These developments mark a shift from single-agent to multi-agent, orchestrated AI frameworks, moving towards a "research team in silico" where AI agents autonomously extract knowledge, generate hypotheses, and execute experiments in a closed-loop system. While multi-agent frameworks and domain-specific LLMs have advanced materials research through automated knowledge extraction, hypothesis generation, and experiment orchestration, they often remain fragmented, focusing on only one or two stages of the discovery pipeline. In contrast, we introduce MatAgent, a multi-agent LLM framework that unifies six essential functions, property prediction, hypothesis generation, experimental data analysis, highperformance material discovery, data-driven experimentation, and literature review, under a single orchestrated platform. Each agent specializes in a distinct task yet collaborates seamlessly to accelerate the entire research cycle, from experiment planning to real-time data interpretation. This genuinely closed-loop workflow reduces research timelines, enhances reproducibility, and moves the field closer to fully autonomous laboratories. The following sections detail MatAgent's methodology, key findings, and broader implications for accelerating innovation in materials science.

2 PROPOSED APPROACH

The MatAgent presented in Figure 1 combines advanced machine learning, structured knowledge retrieval, and autonomous data analysis to streamline hypothesis-driven materials science research. Built on the LangChain and Firecrawl frameworks along with leveraging OpenAI's GPT models, LangGraph, and various auxiliary AI architectures, this approach oversees the entire research cycle ranging from hypothesis generation to final report production through a cohesive workflow that balances automation with human-in-the-loop (HITL) reviews.



Figure 1: End-to-End MatAgent workflow with Human-in-the-Loop (HITL) communication.

2.1 System Architecture and Key Components

According to Figure 1, at the heart of the framework is a Process Planner, which coordinates specialized modules such as Code Generation, Visualization, Report Generation, Web Search, and a Quality Monitor. To ensure continuity, a Context-State Taker logs each decision and data point, enabling modules to reference prior states without redundant computations. The system uses Chainof-Thought Reasoning (CoT) to break down complex tasks, while Adaptive Workflow Mechanisms allow dynamic responses to new data or quality checks.

2.2 WORKFLOW DESCRIPTION

The end-to-end workflow proceeds through five main stages. It begins with hypothesis generation and an initial review of ideas, followed by central processing and specialized coordination where these ideas are integrated and refined. Next, a quality review stage ensures the workflow is adjusted as needed, paving the way for a final human review along with the generation of outputs in multiple formats. The process culminates with the integration of external tools and the secure storage of data.

2.2.1 STAGE 1: HYPOTHESIS GENERATION AND INITIAL REVIEW

In our MatAgent framework, the Hypothesis Agent formulates new research questions or potential materials by analyzing existing datasets, literature, and learned patterns. Immediately after hypothesis generation, a Human-in-the-Loop (HITL) review evaluates these AI-generated ideas for scientific viability and practical feasibility. To further validate the scientific novelty and utility of the generated hypotheses, we employ a two-tiered evaluation framework. First, each hypothesis undergoes an automated assessment using NLP-based novelty scoring, where it is compared against a curated database of existing literature to provide an initial quantitative measure of novelty. Next, a panel of

materials science experts conducts a detailed review using a structured scoring rubric that assesses novelty, utility, and feasibility. Feedback from this expert review is iteratively fed back into the MatAgent system, prompting refinement and regeneration of any hypothesis that fails to meet the established criteria until an acceptable proposal emerges. Finally, selected hypotheses will undergo experimental validation through computational simulations and lab-based tests, ensuring that the final outputs are both innovative and practically actionable.

2.2.2 STAGE 2: CENTRAL PROCESSING AND SPECIALIZED COORDINATION

Once a hypothesis is approved, the Process Planner initiates a coordinated effort among key modules. The Code Generation module produces the necessary scripts (e.g., Python) for computational modeling or data analytics, while the Visualization module converts raw or processed data into graphs, plots, or other graphics for easy interpretation. At the same time, the Report Generation module structures intermediate findings into concise summaries to guide subsequent reviews, and the Web Search module gathers relevant publications, patents, and databases to either reinforce or challenge the existing hypothesis. Throughout this stage, the Context-State Taker continuously logs decisions and results, while the Quality Monitor ensures that each module's output meets domain-specific standards.

2.2.3 STAGE 3: QUALITY REVIEW AND WORKFLOW ADJUSTMENTS

All intermediate outputs pass through the Quality Monitor, which checks for errors, inconsistencies, and physically implausible properties. If issues arise, the workflow automatically loops back to the relevant modules (e.g., Code Generation or Visualization) for corrections. Complex situations may also prompt additional HITL involvement to provide domain-specific expertise.

2.2.4 STAGE 4: FINAL HUMAN REVIEW AND MULTI-FORMAT OUTPUT GENERATION

Once the Quality Monitor's criteria are satisfied, a final HITL review confirms the scientific robustness of the results. Approved outcomes are then formatted into visualized figures that offer graphical insight into key findings, code snippets that enable reproducible workflows, datasets that facilitate subsequent data exploration, and scientific reports detailing the entire research trajectory from the initial hypothesis to the final conclusions.

2.2.5 STAGE 5: INTEGRATION WITH EXTERNAL TOOLS AND DATA STORAGE

Finally, the system seamlessly connects to external software and databases, preserving full traceability and scalability. This includes storage solutions for large datasets, simulation platforms for advanced analyses, or laboratory automation tools for future experimental validation.

By uniting hypothesis generation, specialized module coordination, rigorous quality checks, and adaptive feedback loops, this multi-agent LLM framework, MatAgent, accelerates materials science research while maintaining robust standards of accuracy and reproducibility.

3 BENCHMARKING CATEGORIES FOR MODEL EVALUATION

3.1 MATERIAL PROPERTY PREDICTION

Accurate prediction of material properties, particularly the band gap, is essential for optimizing materials in semiconductors, photovoltaics, and optoelectronics. Traditional experimental approaches are resource-intensive and time-consuming, making machine learning (ML) models a promising alternative for rapid, data-driven predictions. MatAgent employs ML techniques to predict experimental band gaps based on structural and electronic features, evaluating four regression models including Linear Regression (LR), Random Forest (RF), Gradient Boosting Regressor (GBR), and Support Vector Regression (SVR). The dataset undergoes preprocessing, including cleaning, standardization, and feature selection to retain the most influential predictors. The data is split into training (80%) and testing (20%) subsets, and each model is evaluated based on R^2 and Mean Squared Error (MSE) to measure predictive accuracy. The results demonstrate that Random Forest outperformed all models, achieving the highest R^2 score (0.92) and the lowest MSE (0.01), indicating superior generalization ability and reaffirming the necessity of non-linear models for band gap prediction. Further details on models performance can be found in Table 1, and an experiment use case by querying from the MatAgent regarding the material property prediction purposes is displayed in Figure 2.

Table 1: Model Performance Comparison for experimental band gap prediction

Model	R ² Score	MSE
	0.05	0.00
Linear Regression	0.85	0.02
Random Forest	0.92	0.01
Gradient Boosting	0.90	0.015
Support Vector Regression	0.88	0.018



Figure 2: MatAgent communicative output for experimental band gap prediction.

3.2 ADDITIONAL MATAGENT EXPERIMENT

Further ablation studies on other key areas like Hypothesis Generation for Novel Materials, Experimental Data Analysis, Discovery of High-Performance Alloys, Data-Driven Experimentation, and Literature Review and Knowledge Synthesis are comprehensively discussed in the Appendix section. Moreover, utilized datasets and supplementary results related to all six key applications of MatAgent are presented in Appendix illuminating the history of agents' interactions, generated codes, and visualized results.

4 CONCLUSION

This study demonstrates the effectiveness of MatAgent, a data-driven methodology that integrates machine learning, automated literature review, and computational modeling to accelerate materials discovery and optimization. Across multiple benchmarks, MatAgent successfully predicted material properties, generated hypotheses for novel materials, and optimized experimental and theoretical frameworks. The predictive models achieved high accuracy, with gradient boosting and neural networks emerging as top performers, confirming the importance of nonlinear modeling for complex material behaviors. Automated hypothesis generation identified key elemental and structural descriptors, guiding future materials engineering efforts. Furthermore, knowledge synthesis in photovoltaics and structural materials validated data-driven insights for improving performance and efficiency. The application of MatAgent in material science highlights the transformative potential of AI-driven approaches in reducing experimental costs, refining predictive capabilities, and uncovering novel materials with tailored properties. Future research should focus on expanding datasets, incorporating deep learning frameworks, and validating predictions through experimental studies to further enhance MatAgent's impact on materials discovery and innovation.

REFERENCES

- Milad Abolhasani and Eugenia Kumacheva. The rise of self-driving labs in chemical and materials sciences. *nature synthesis*, 2:483–492, 2023.
- Nicolás Amigo, Simón Palominos, and Felipe Valencia. Machine learning modeling for the prediction of plastic properties in metallic glasses. *scientific reports*, 13:348, 2023.
- Iz Beltagy, Kyle Lo, and Arman Cohan. Scibert: A pretrained language model for scientific text. *arXiv*, 2019.
- Tom Brown, Benjamin Mann, Nick Ryder, and Melanie Subbiah. Language models are few-shot learners. Advances in Neural Information Processing Systems, 2020.
- Zi-Yi Chen, Fan-Kai Xie, Meng Wan, Yang Yuan, Miao Liu, Zong-Guo Wang, Sheng Meng, and Yan-Gang Wang. Matchat: A large language model and application service platform for materials science. *Chinese Physics B*, 32:118104, 2023.
- Connor Coley, Natalie Eyke, and Klavs Jensen. Autonomous discovery in the chemical sciences part i: Progress. *Chemie International Edition*, 59:22801–23342, 2019a.
- Connor Coley, Natalie Eyke, and Klavs Jensen. Autonomous discovery in the chemical sciences part ii: Outlook. *Chemie International Edition*, 59:23345–23898, 2019b.
- Andrew Ferguson and Keith Brown. Data-driven design and autonomous experimentation in soft and biological materials engineering. *Annual Review of Chemical and Biomolecular Engineering*, 13:25–44, 2022.
- Alireza Ghafarollahi and Markus Buehler. Atomagents: Alloy design and discovery through physics-aware multi-modal multi-agent artificial intelligence. *arXiv*, 2024a.
- Alireza Ghafarollahi and Markus Buehler. Sciagents: Automating scientific discovery through multi-agent intelligent graph reasoning. *arXiv*, 2024b.
- Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare (HEALTH)*, 3: 1–23, 2021.
- Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. *arXiv*, 2024.
- Florian Häse, Loïc Roch, Christoph Kreisbeck, and Alán Aspuru-Guzik. Phoenics: A bayesian optimizer for chemistry. ACS Central Science, 4:1134—1145, 2018.
- Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin Persson. The materials project: A materials genome approach to accelerating materials innovation. APL Materials, 1:011002, 2013.
- JDMWC Kenton and LK Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. Proceedings of naacL-HLT, 2019.
- Ge Lei, Ronan Docherty, and Samuel Cooper. Materials science in the era of large language models: a perspective. *Digital Discovery*, 3:1257–1272, 2024.
- Santiago Miret and Anoop Krishnan. Are llms ready for real-world materials discovery? *arXiv*, 2024.
- Vaibhav Mishra, Somaditya Singh, Mohd Zaki, Hargun Grover, Santiago Miret, and Mausam Krishnan. Llamat: Large language models for materials science. *BOKU 2024 AI4Mat*, 2024.
- Ghanshyam Pilania. Machine learning in materials science: From explainable predictions to autonomous design. *Computational Materials Science*, 193:110360, 2021.

- Shyue Ping Ong. Accelerating materials science with high-throughput computations and machine learning. volume 161, pp. 143–150. 2019.
- Jean-Louis Reymond. The chemical space project. Accounts of Chemical Research, 48, 2015.
- Marco Roos, Scott Marshall, Andrew Gibson, and Martijn Schuemie. Structuring and extracting knowledge for the support of hypothesis generation in molecular biology. *MRS Bulletin*, 10, 2009.
- Nathan Szymanski, Yan Zeng, Haoyan Huo, Christopher Bartel, Haegyeom Kim, and Gerbrand Ceder. Toward autonomous design and synthesis of novel inorganic materials. *Materials Hori*zons, 8:2169–2198, 2021.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- Logan Ward, Ankit Agrawal, Alok Choudhary, and Christopher Wolverton. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj computational materials*, 2:16028, 2016.
- Leigh Weston, Vahe Tshitoyan, John Dagdelen, Olga Kononova, Kristin Persson, Gerbrand Ceder, and Anubhav Jain. Named entity recognition and normalization applied to large-scale information extraction from the materials science literature. *Journal of Chemical Information and Modeling*, 59:3692–3702, 2019.
- Guangzhi Xiong, Eric Xie, Amir Hassan Shariatmadari, Sikun Guo, Stefan Bekiranov, and Aidong Zhang. Improving scientific hypothesis generation with knowledge grounded large language models. *arXiv*, 2024.
- Yoel Zimmermann, Adib Bazgir, Zartashia Afzal, Fariha Agbere, Qianxiang Ai, Nawaf Alampara, Alexander Al-Feghali, Mehrad Ansari, Dmytro Antypov, Amro Aswad, Jiaru Bai, Viktoriia Baibakova, Devi Dutta Biswajeet, Erik Bitzek, Joshua D Bocarsly, Anna Borisova, Andres M Bran, L Catherine Brinson, Marcel Moran Calderon, Alessandro Canalicchio, Victor Chen, Yuan Chiang, Defne Circi, Benjamin Charmes, Vikrant Chaudhary, Zizhang Chen, Min-Hsueh Chiu, Judith Clymo, Kedar Dabhadkar, Nathan Daelman, Archit Datar, Matthew L Evans, Maryam Ghazizade Fard, Giuseppe Fisicaro, Abhijeet Sadashiv Gangan, Janine George, Jose D Cojal Gonzalez, Michael Götte, Ankur K Gupta, Hassan Harb, Pengyu Hong, Abdelrahman Ibrahim, Ahmed Ilyas, Alishba Imran, Kevin Ishimwe, Ramsey Issa, Kevin Maik Jablonka, Colin Jones, Tyler R Josephson, Greg Juhasz, Sarthak Kapoor, Rongda Kang, Ghazal Khalighinejad, Sartaaj Khan, Sascha Klawohn, Suneel Kuman, Alvin Noe Ladines, Sarom Leang, Magdalena Lederbauer, Sheng-Lun Mark Liao, Hao Liu, Xuefeng Liu, Stanley Lo, Sandeep Madireddy, Piyush Ranjan Maharana, Shagun Maheshwari, Soroush Mahjoubi, José A Márquez, Rob Mills, Trupti Mohanty, Bernadette Mohr, Seyed Mohamad Moosavi, Alexander Moßhammer, Amirhossein D Naghdi, Aakash Naik, Oleksandr Narykov, Hampus Näsström, Xuan Vu Nguyen, Xinyi Ni, Dana O'Connor, Teslim Olayiwola, Federico Ottomano, Aleyna Beste Ozhan, Sebastian Pagel, Chiku Parida, Jaehee Park, Vraj Patel, Elena Patyukova, Martin Hoffmann Petersen, Luis Pinto, José M Pizarro, Dieter Plessers, Tapashree Pradhan, Utkarsh Pratiush, Charishma Puli, Andrew Qin, Mahyar Rajabi, Francesco Ricci, Elliot Risch, Martiño Ríos-García, Aritra Roy, Tehseen Rug, Hasan M Saveed, Markus Scheidgen, Mara Schilling-Wilhelmi, Marcel Schloz, Fabian Schöppach, Julia Schumann, Philippe Schwaller, Marcus Schwarting, Samiha Sharlin, Kevin Shen, Jiale Shi, Pradip Si, Jennifer D'Souza, Taylor Sparks, Suraj Sudhakar, Leopold Talirz, Dandan Tang, Olga Taran, Carla Terboven, Mark Tropin, Anastasiia Tsymbal, Katharina Ueltzen, Pablo Andres Unzueta, Archit Vasan, Tirtha Vinchurkar, Trung Vo, Gabriel Vogel, Christoph Völker, Jan Weinreich, Faradawn Yang, Mohd Zaki, Chi Zhang, Sylvester Zhang, Weijie Zhang, Ruijie Zhu, Shang Zhu, Jan Janssen, Ian Foster, and Ben Blaiszik. Reflections from the 2024 large language model (llm) hackathon for applications in materials science and chemistry. *arXiv*, 2024.

A APPENDIX

In this section, the utilized datasets and supplementary results are shown for each category and relevant discussions are provided accordingly. The diverse range of results span from the MatAgent's interactive query-response, history of the interactions among various agents, the corresponding generated codes, and visualized figures as output.

A.1 DATASET AND COMPUTATIONAL METHODS

A.1.1 MATERIAL PROPERTY PREDICTION

The study uses a dataset sourced from a CSV file which contains a comprehensive collection of materials-related features alongside experimentally determined band gap values. This dataset serves as the foundation for predicting material properties. According to Figure 3, the features in the dataset include various structural and electronic descriptors that are presumed to influence the band gap, providing a robust basis for the predictive models.

The preprocessing workflow begins with loading the dataset directly from CSV file. Given that the raw data may include non-numeric entries and missing values, an initial cleaning step is applied where only numeric columns are retained, and any rows with missing values are discarded. This ensures that the dataset used for model training is clean and reliable. In addition to these steps, the features were standardized to maintain a uniform scale across all predictors, which is crucial for the convergence and performance of many machine learning algorithms. Finally, the cleaned dataset is split into training and testing sets (commonly with an 80/20 ratio) to facilitate model training and subsequent evaluation.

	А	В	С	D	E	F	G	н
1	name	experimental_band_gap	phonon_cutoff_frequency	mean_phonon_frequency	electronic_contribution_of_dielectric_constant	total_dielectric_constant	nearest_neighbor_distance	density
2	LiF	13.6	19.599	10.435	2.068	7.936	1.943	2.938
3	LiCl	9.4	12.531	6.301	3.15	10.003	2.477	2.317
4	LiBr	7.6	11.188	4.934	3.664	11.287	2.653	3.862
5	Lil	6.1	9.701	4.113	4.465	12.667	2.897	4.569
6	NaF	11.5	12.275	7.416	1.821	4.17	2.227	3.156
7	NaCl	8.5	7.281	4.46	2.604	5.186	2.721	2.408
8	NaBr	7.1	6.357	3.589	3.003	5.728	2.886	3.554
9	Nal	5.9	5.439	2.92	3.582	6.659	3.125	4.078
10	KF	10.9	11.446	6.984	2.087	5.475	2.458	3.25

Figure 3: The sample CSV file for material property prediction task.

A.1.2 Hypothesis Generation for Novel Materials

The study leverages the dataset which comprises over 21,000 entries and 82 features related to the properties of superconducting materials. According to Figure 4, key descriptors in the dataset include compositional details, physical properties such as mean atomic mass, and other relevant material characteristics, with the target variable being the critical temperature (Tc), the temperature at which a material becomes superconducting. This extensive dataset forms the empirical basis for generating novel hypotheses regarding superconducting materials.

To ensure robust and reliable analysis, the following preprocessing steps were rigorously applied:

(1)- The raw dataset contained mixed data types and non-numeric entries. Using cleaning scripts, all columns were coerced into numeric types. Conversion errors were handled by setting invalid entries to NaN, followed by either dropping these rows or imputing missing values with column means. This cleaning step was crucial to remove inconsistencies and ensure that all features are directly comparable.

(2)- After cleaning, the dataset was split into features and the target variable (Tc). Feature scaling was then performed using StandardScaler to standardize the range of all predictors. This normalization is essential for improving the convergence and performance of machine learning models.

(3)- Finally, the preprocessed data was partitioned into training and testing sets (typically an 80/20 split) to enable unbiased model training and evaluation.

	А	В		С		D		Е			F		G		н
1	number_of_elements	mean_atomic_mass	wtd_me	an_atomic_mas	s gm	ean_atomic_ma	ss wtd_gm	ean_ator	nic_mass	entro	py_atomic_mass	s wtd_e	ntropy_atomic_	mass	range_atomic_mass
2	4	889.444.675	578.62	6.922.857.143	663	3.615.924.315.7	19 361.166	6.119.053	.847	11.81	7.952.393.305	106.2	39.554.519.617		12.290.607
3	5	92.729.214	585.18	1.161.428.571	731	1.327.872.225.00	363.966	6.020.291	.995	144.9	30.919.335.685	105.7	75.512.271.911		12.290.607
4	4	889.444.675	578.85	2.418.571.429	663	3.615.924.315.7	19 361.225	5.090.359	.592	11.81	7.952.393.305	975.9	80.464.165.498		12.290.607
5	4	889.444.675	578.73	9.670.714.286	663	3.615.924.315.7	19 361.195	5.603.503	.211	11.81	7.952.393.305	10.22	2.908.923.957		12.290.607
6	4	889.444.675	578.40	1.427.142.857	663	3.615.924.315.7	19 361.107	157.375	.382	11.81	7.952.393.305	112.9	22.373.013.507		12.290.607
7	4	889.444.675	577.95).435.714.286	663	3.615.924.315.7	19 360.989	.262.894	.004	11.81	7.952.393.305	122.5	20.280.321.875		12.290.607
8	4	889.444.675	576.82	2.957.142.857	663	3.615.924.315.7	19 360.694	.695.057	.962	11.81	7.952.393.305	131.6	85.658.780.632		12.290.607
9	4	765.177.175	571.75	1.418.571.429	593	3.100.961.333.4	18 358.913	8.680.915	5.719	119.7	27.303.349.869	943.5	60.479.191.362		12.290.607
10	4	765.177.175	568.08	3.170.714.286	593	3.100.961.333.4	18 357.734	.324.182	2.576	119.7	27.303.349.869	981.8	79.783.570.863		12.290.607
	1	J		К		L	М			N	0		Р		Q
1	wtd_range_atomic_mass	s std_atomic_mass	wtd_sto	_atomic_mass	nean_	fie v	/td_mean_fie		gmean_fi	Э	wtd_gmean	fie	entropy_fie		wtd_entropy_fie
2	317.949.208.571.429	519.688.277.861.034	536.22	5.345.301.219	775.42	5 1	01.026.857.	142.857	71.815.28	9.995.2	13 938.016.780	0.052.20	4 130.596.703.5	599.15	8 791.487.788.469.155
3	36.161.939	470.946.331.703.134	539.79	3.696.513.451	766.44	1	01.061.285.	714.286	720.605.5	10.513.	725 938.745.412	2.527.43	3 154.414.454.3	26.97	3 807.078.214.938.731
4	35.741.099	519.688.277.861.034	536.56	2.677.320.982	775.42	5 1	010.82		71.815.28	9.995.2	13 939.009.03	5.665.86	4 130.596.703.5	599.15	8 773.620.193.146.673
5	337.680.099.285.714	519.688.277.861.034	5.363.9	40.496.787	775.42	5 1	01.054.428.	571.429	71.815.28	9.995.2	13 938.512.776	5.724.54	6 130.596.703.5	599.15	8 783.206.660.361.291
6	278.487.427.142.857	519.688.277.861.034	535.88	7.706.050.743	775.42	5 1	00.971.714.	285.714	71.815.28	9.995.2	13 937.025.572	2.960.08	6 130.596.703.5	99.15	8 805.229.640.813.357
7	206.874.578.571.429	519.688.277.861.034	535.21	.503.531.322	775.42	5 1	00.861.428.	571.429	71.815.28	9.995.2	13 935.046.299	9.909.73	7 130.596.703.5	599.15	8 824.742.649.703.668
8	107.656.385.714.286	519.688.277.861.034	533.51	5.579.773.433	775.42	5 1	00.585.714.	285.714	71.815.28	9.995.2	13 930.116.388	3.995.67	3 130.596.703.5	599.15	8 841.871.802.066.264
9	36.451.199	442.894.585.275.144	529.24	1.398.674.945	787.05	1	01.148.428.	571.429	734.219.6	24.171.	273 940.196.999	9.785.30	5 131.300.802.4	21.59	1 776.332.032.161.783
10	348.331.599.285.714	442.894.585.275.144	525.33	2.069.817.455	/87.05	1	01.154.071.4	428.571	/34.219.6	24.171.	273 940.294.344	1.338.80	9 131.300.802.4	21.59	1 786.865.234.305.601
	R	S		Т		U		V			W		Х		Y
1	range_fie	wtd_range_fie	std_fi	9	wtd_	std_fie	mean_at	omic_rad	lius wtd	_mean_	_atomic_radius	gmean_	_atomic_radius	wtd_	gmean_atomic_radius
2	810.6	735.985.714.285.714	323.8	11.807.806.633	355.8	562.966.713.294	160.25		105	.514.28	5.714.286	136.12	6.003.095.455	84.52	28.422.716.633
3	810.6	743.164.285.714.286	290.1	83.029.138.508	354.9	963.511.171.592	161.2		104	.971.42	8.571.429	141.46	5.214.777.999	843.7	701.669.575.628
4	810.6	743.164.285.714.286	323.8	11.807.806.633	354.8	304.182.855.034	160.25		104	.685.71	4.285.714	136.12	6.003.095.455	84.2	14.573.243.296
5	810.6	739.575	323.8	11.807.806.633	355.1	183.884.421.944	160.25		105	.1		136.12	6.003.095.455	84.37	71.352.045.645
6	810.6	728.807.142.857.143	323.8	11.807.806.633	35.63	31.928.137.213	160.25		106	.342.85	7.142.857	136.12	6.003.095.455	848.4	134.418.389.765
7	810.6	714.45	323.8	11.807.806.633	357.8	324.565.668.595	5 160.25		108			136.12	6.003.095.455	854.7	770.064.830.415
8	810.6	678.557.142.857.143	323.8	11.807.806.633	361.5	545.641.683.603	3 160.25		112	.142.85	7.142.857	136.12	6.003.095.455	870.8	316.940.804.888
9	772	742.5	314.5	05.965.762.178	353.8	394.216.896.887	151.75		104	.2		131.30	2.196.551.623	840.4	111.284.062.601
10	772	738.578.571.428.571	314.5	05.965.762.178	353.8	319.831.451.067	151.75		104	.371.42	8.571.429	131.30	2.196.551.623	841.1	108.347.003.322
	Z	AA		AB		AC			AD		AE		AF		AG
1	entropy_atomic_radius	wtd_entropy_atomic	radius	range_atomic_	radius	wtd_range_atc	mic_radius	std_ator	mic_radiu	is w	/td_std_atomic_i	radius	mean_Density		wtd_mean_Density
2	125.924.397.214.289	120.703.998.701.46	1	205		429.142.857.1	42.857	752.375	5.404.967	.494 6	92.355.694.829	807	465.435.725		296.150.228.571.429
3	150.832.754.035.259	12.041.147.982.326		205		505.714.285.7	14.286	67.321.	319.060.	161 6	80.088.169.554	.027	58.214.858		302.101.657.142.857
4	125.924.397.214.289	113.254.686.280.436	5	205		493.142.857.1	42.857	752.375	5.404.967	.494 6	7.797.712.320.6	85	465.435.725		299.915.942.857.143
5	125.924.397.214.289	117.303.291.789.271		205		461.142.857.1	42.857	752.375	5.404.967	.494 6	85.216.649.785	203	465.435.725		298.033.085.714.286
6	125.924.397.214.289	126.119.371.912.948	3	205		365.142.857.1	42.857	752.375	5.404.967	.494 7	06.344.484.378	724	465.435.725		292.384.514.285.714
7	125.924.397.214.289	133.133.890.122.57	3	205		237.142.857.1	42.857	752.375	5.404.967	.494 7	33.241.336.220	795	465.435.725		284.853.085.714.286
8	125.924.397.214.289	136.727.867.491.83	2	205		154.285.714.2	85.714	752.375	5.404.967	.494 7	95.009.947.312	944	465.435.725		266.024.514.285.714
9	127.527.399.052.522	112.181.816.170.133	3	171		49.8		655.796	6.271.718	.588 6	68.485.708.790	504	443.435.725		2.986.588
10	127.527.399.052.522	116.005.975.115.976		171		468.428.571.4	28.571	655.796	6.271.718	.588 6	71.114.577.035.	996	443.435.725		296.147.371.428.571

Figure 4: The sample CSV file for hypothesis generation for novel materials task.

A.1.3 EXPERIMENTAL DATA ANALYSIS

The analysis is based on the experimental dataset which contains detailed measurements of gravimetric and molar heat capacities recorded at multiple temperatures (e.g., 250°C, 275°C, 300°C, etc.). According to Figure 5, the dataset features descriptive column headers, such as those indicating mean and standard deviation values for heat capacities, that ensure clarity and consistency in the experimental records. Preliminary reviews confirm that the data is largely complete, with the structure facilitating straightforward interpretation and subsequent analysis.

To ensure the reliability and accuracy of the analysis, a rigorous data preprocessing pipeline was implemented:

(1)- The raw dataset was examined for missing or inconsistent values. Any missing entries were addressed either by filling with the mean of the respective numerical columns or by dropping affected rows. Duplicate entries were identified and removed to prevent bias in downstream analyses.

(2)- Data types were standardized by converting relevant columns to numeric formats (e.g., floats). This conversion ensures that all measurements can be directly compared and analyzed statistically.

(3)- Where necessary, the features were normalized or standardized to place them on a comparable scale, which is crucial for both visualization and the performance of machine learning algorithms.

(4)- The entire preprocessing workflow was automated using Python scripts. These scripts not only performed the cleaning tasks but also saved a refined version of the dataset for further analysis.

	А	В		С		D		E		F	F G			н	
1		name	Cv_gravi	metric_250.00_mean	Cv_gra	avimetric_250.00_std	Cν	_molar_250.00_mean	Cv	_molar_250.00_std	Cv_grav	imetric_275.00_mea	n Cv	gravimetric_275.00_std	
2	0	ABW	0.650656	31304641718	0.0088	3717452982956	13.	03140604654948	0.1	776842020087944	0.69158	01481037875	0.0	074014842840063	
3	1	ACO	0.648632	26082327557	0.0072	2897903019212	12.	99087874094645	0.1	0.1460006491459085 0		0.691874503821114		0.0068332987154121	
4	2	AEI	0.644705	54576173125	0.0073	3139384590105	12.	912225375705296	0.1	464842908509095	0.68950	84592426938	0.0	0.0067841500226103	
5	3	AEL	0.667292	21638806788	0.0029	466680121969	13.	364594187418618	0.0	590161616150815	0.70971	62159244713	0.0	027045638243125	
6	4	AEN	0.668814	0178328584	0.0020	0756572743435	13.	39507403055827	0.0	415714714562806	0.711601	13925178777	0.0	017473234584889	
7	5	AET	0.664549	2292268527	0.0029	385933321784	13.	309658417878326	0.0	588544411162041	0.7083246044136547		0.0	024492937907832	
8	6	AFG	0.651735	6006791793	0.0066	5435455283	13.	053025783962674	0.1	332740784195352	0.69469	92028656064	0.0	057868558968829	
9	7	AFI	0.660661	5676234115	0.0047	780029037312	13.	231795942518447	0.0	956943199562201	0.70342	55063706347	0.0	038641620637312	
10	8	AFN	0.656710	07817144435	0.0041	458256961468	13.	152669207255046	0.0	830330116249983	0.70103	5498848106	0.0	033370159629265	
		1		J		к		L		м		N		0	
1	Cv_mola	ar_275.00	_mean C	v_molar_275.00_std	Cv_gra	vimetric_300.00_mear	n C	v_gravimetric_300.00_	std	Cv_molar_300.00_m	ean Cv	_molar_300.00_std	Cv_gr	avimetric_325.00_mean	
2	13.8510	36364237	466 0	1482376673885083	0.7319	976123599645	0.	0061033343255755		14.66052138010660	9 0.1	222381902060593	0.769	265362128368	
3	13.8569	31749979	658 0	1368579900021457	0.7347	178193743423	0.	0061062175579371		14.71500195821126	4 0.1	222959358721213	0.774	3628646220241	
4	13.8095	44372558	594 0	1358736350678421	0.7322	528839845686	0.	0060171835724242		14.66563398573133	8 0.1	205127543068696	0.772	685815875039	
5	14.2142	67344156	903 0	.0541672747297144	0.7496	161826756133	0.	0021202757043258		15.01338786824544	3 0.0	424650938338081	0.786	2992910406844	
6	14.2520	23849487	305 0	0349955689589616	0.7509	425902183062	0.	0013411568726547		15.03995329115126	0.0	268608239612163	0.786	8043629697997	
7	14.1863	96009657	12 0	0.049054700971186 0.7485		308438681412	0.	0023062739391516		14.99165059407552	5 0.0	461902850807239	0.785	0993178639999	
8	13.9135	05104912	0.1158997285883612 0.7354		0.7354	554275288357	0.	0048148555496201		14.72977484809027	8 0.0	964324084333476	0.773	3350010869817	
9	14.0882	76384141	71 0	0773918242286162	0.7445	269503972243	0.	0036523133620507		14.91146021525065	0.0	731488972464889	0.782	4846443123714	
10	14.0404	29102579	0754 0	.0668340894070897	0.7416	118026850905	0.	0031385042161222		14.85307534535726	0.0	628582762909186	0.779	3315407772083	
		Р		Q		R		S		Т		U		V	
1	Cv_grav	imetric_3	25.00_std	Cv_molar_325.00_m	nean C	v_molar_325.00_std	Cv	_gravimetric_350.00_m	near	n Cv_gravimetric_35	0.00_std	Cv_molar_350.00_	mean	Cv_molar_350.00_std	
2	0.00625	25528330	0573	15.40692359924316	6 0	.1252267533957557	0.8026307933441161			0.00679698053983	358	16.075169792175	295	0.1361306059498862	
3	0.00657	62077296	6908	15.50901688893636	62 0	.1317089460310209	9 0.808631760763172			0.00678570741265	548	16.195357767740887		0.135904826631392	
4	0.00665	51335567	7554	15.47542878892686	6 0	.1332896803880546	0.8	068309986654083	0.0073315461635389			16.15929192437066		0.1468369397179738	
5	0.00210	57697533	3651	15.74808083089192	8 0	.0421745671973733	0.8	186663829238867		0.00206910823794	16.396332183837888		0.0414403067003487		
6	0.00167	02945187	7875	15.75819646199544	4 0	.0334528256517296	0.8	19034356013655		0.0015954989207265		16.4037019856770	28	0.0319548119342037	
7	0.00236	93966400	076	15.72404764811198	0	.0474545128457364	0.8	176740298361342		0.00226300986339	968	16.376457236961	084	0.0453237878450996	
8	0.00555	97011777	301	15.48843073527018	34 0	.1113502511576981	0.8	068081417480727		0.00616427898605	556	16.158834143744	574	0.1234587959606202	
9	0.00406	98965173	3032	15.67168070475260	09 0	.0815122944382017	0.8	155202689868086		0.00450116811202	69	16.333321499294	71	0.0901498450644874	
10	0.00344	91029817	7505	15.60853003184001	0	.0690789794287975	0.8	125122962105591		0.00361885880631	23	16.273077519734	7	0.0724788660587037	
		W		х		Y		Z		AA		AB		AC	
1	Cv_gravi	metric_37	'5.00_mea	n Cv_gravimetric_375	.00_std	Cv_molar_375.00_me	ean	Cv_molar_375.00_std	Cv	gravimetric_400.00_	mean C	v_gravimetric_400.00	_std	Cv_molar_400.00_mean	
2	0.831583	37527926	179	0.00692855961715	В	16.655042559305826	3	0.138765884868404	0.8	8571176122933835	0	.0084945621113358		17.16643725077311	
3	0.837808	34265443	314	0.00703188367156	33	16.779710947672527	7	0.1408352693625375	0.8	8665174742586846	0	.0075440737113052		17.35469862620036	
4	0.835770	9559724	402	0.00705281844972	31	16.73890428331163		0.1412545531929012	0.8	8627788354361905	0	.0072463528665362		17.27982079399957	
5	0.848364	16831588	989	0.002158097545454	43	16.99113271077474		0.0432225934501152	0.8	8757935355545513	0	.0025241206298645		17.540480509440105	
6	0.01000	0.8486903872471288 0.0018953433727322		16.997655944824217		0.0379601266034186	0.8	8756512771688483	0	.0020110482691202		17.53763134426541			
	0.848690	3872471	288	0.00189534337273	22	16.997655944824217		0.0010001200001100							
7	0.848690)3872471 34513295	288 783	0.00189534337273	22 03	16.997655944824217 16.97258204707393		0.0483190505754767	0.8	8753522399074366	0	.0023589404291833		17.531642196090132	
7 8	0.848690 0.847438 0.835663	03872471 34513295 37957076	288 783 881	0.001895343372733 0.002412562877934 0.00594625371443	22 03 11	16.997655944824217 16.97258204707393 16.736758066813152	2	0.0483190505754767 0.1190921640179978	0.8	8753522399074366 8610924722054577	0	.0023589404291833 .0066283318292064		17.531642196090132 17.246046142578123	
7 8 9	0.848690 0.847438 0.835663 0.844309	03872471 34513295 37957076 92966534	288 783 881 796	0.001895343372733 0.002412562877931 0.00594625371443 0.004258656997663	22 03 11 2	16.997655944824217 16.97258204707393 16.736758066813152 16.909911024305558	2	0.0483190505754767 0.1190921640179978 0.0852928082148744	0.8	8753522399074366 8610924722054577 8715527290899991	0	.0023589404291833 .0066283318292064 .0047650948803446		17.531642196090132 17.246046142578123 17.45554521348741	

Figure 5: The sample CSV file for experimental data analysis task.

A.1.4 ACCELERATED DISCOVERY OF HIGH-PERFORMANCE ALLOYS OR POLYMERS

The investigation relies on the dataset which is a comprehensive collection of experimental measurements and compositional details for various alloys. As can be seen in Figure 6, Key attributes within this dataset include the yield strength (expressed in MPa) along with several material properties that characterize each alloy. The dataset serves as the empirical basis for predicting and optimizing alloy performance, and it is derived from either experimental measurements, literature compilations, or a combination thereof. The accompanying verification script confirms that essential target variables (e.g., "YS (MPa)") and relevant feature columns are present, ensuring the integrity of the dataset.

Robust data preprocessing is fundamental to the success of the predictive models. The preprocessing workflow implemented for this dataset involves several critical steps:

(1)- Python scripts are used to clean the column headers by stripping whitespace and converting them to lowercase. This standardization ensures consistent naming conventions throughout the analysis.

(2)- The dataset is first loaded, and missing values are identified. In multiple scripts, rows with missing values, especially in crucial target variables, are either dropped or imputed. For example, any missing data in numerical columns is handled by either dropping the affected rows or replacing them with the mean values.

(3)- Non-numeric columns (such as the alloy name) are excluded from the feature set to ensure that only quantitative data is used for modeling. The features are normalized using techniques such as standardization (via StandardScaler), which scales the data to have zero mean and unit variance. This step is essential for many machine learning algorithms, particularly when features have different scales.

(4)- The cleaned and normalized data is partitioned into training and testing sets, typically using an 80/20 split. This approach facilitates unbiased evaluation of the predictive models.

Although the primary focus is on data-driven modeling, the overall framework aims to combine theoretical calculations with machine learning predictions. Theoretical aspects, such as computing lattice constants and mixing enthalpy, are implied in the accompanying notes and literature review. These theoretical parameters can further refine the predictive models by incorporating fundamental material properties, thereby narrowing down candidate alloys with superior performance.

	А	В		С	0)	E	F	(G	н		I.	J
1	ID	Alloy		Diff. Lattice Constants	Diff. Melt	ing Point	Mixing Enthalpy	Lattice Constants	Lambo	la	Diff. in atomi	c radii	Omega	Melting Temp.
2	1	CoFeNiSi0.75		1.055487599	44.66304	411	-20.89171828	3.458902992	12365	.52187	0.030453662	2	962.0652443	1752.61769
3	2	Al0.75CoFeNi		0.583340282	340.1653	3332	-18.40336718	3.182622992	3378.8	340617	0.058258848	3	998.2349588	1601.91169
4	3	AlCoCrFeNi		0.547352688	408.5173	377	-18.93648	3.17142	4008.3	389723	0.05777889	3	1190.078167	1684.094
5	4	AIC0.1CoCrFeNi		0.555328048	614.5201	56	-22.16618769	3.157579208	1146.4	18173	0.110198387	7	1084.069454	1726.054788
6	5	AIC0.2CoCrFeNi		0.557150757	652.2820	807	-25.13458534	3.144244341	1004.4	53102	0.118993143	3	999.5033873	1766.373166
7	6	AIC0.3CoCrFeNi		0.558736063	687.2739	188	-27.8650339	3.131380028	895.40)5625	0.126958999	Э	934.97927	1805.15608
8	7	AIC0.4CoCrFeNi		0.560118358	719.9150	354	-30.37955611	3.119084829	809.25	594688	0.134252673	3	884.6532013	1842.556449
9	8	AIC0.5CoCrFeNi		0.561302089	750.5123	3513	-32.69374477	3.107140162	739.46	30265	0.14098458	7	844.5392876	1878.558785
10	9	AICCoCrFeNi		0.564955604	880.0527	'987	-41.79389393	3.053577737	524.31	83716	0.16856264	1	727.3892062	2040.619145
		К		L			М	Ν			0		Р	Q
1	Diff.	Electronegativity Allen	Diff	. Electronegativity	Pauling	Diff. SI	hear modulus	Avg shear mod	dulus	Mixin	g Entropy	Vale	nce electron	YS (MPa)
2	0.04	2160483	0.0	31383117		0.1083	399942	75.73413		11.46	810064	8.00	009	1301
3	0.09	6576617	0.1	1086158		0.3346	610101	68.97413		11.46	810064	7.80	009	794
4	0.10	5878421	0.1	20565335		0.3756	0.375622748 76.1			13.38161136				1251
5	0.20	0369992	0.2	04579646		0.512705511 74.837877			13.92174058		7.13	732	957	
6	0.21	6354457	0.2	1974393		0.5153	326265	73.623937		14.22	242122	7.07	7	906
7	0.23	0950152	0.2	33624142		0.5182	211026	72.45496		14.43	3267391	7.01	888	867
8	0.24	441091	0.2	46445369		0.5213	318045	71.331414		14.58	359149	6.96312		1056
9	0.25	6916391	0.2	58368703		0.5245	524589068 70.246157			14.69805051		6.90916		1060
10	0.30	8957464	0.3	08043482		0.5423	381053	65.367974		14.89	9764879	6.66	68	1251

Figure 6: The sample CSV file for discovery of high-performance alloys task.

A.1.5 DATA-DRIVEN EXPERIMENTATION

The experimentation is based on the dataset which is widely recognized in the field of construction materials research. As depicted in Figure 7, this dataset compiles experimental measurements on concrete strength alongside various mix composition parameters. This rich dataset forms the empirical basis for developing predictive models aimed at optimizing concrete strength, an essential property for ensuring structural integrity and performance in construction.

A robust preprocessing pipeline was established to ensure the quality and reliability of the analysis:

(1)- The dataset is loaded using Python's Pandas library. The accompanying scripts perform initial checks for missing values. In this case, missing values are either absent or handled by dropping incomplete rows, ensuring a clean dataset for subsequent analysis. Column names are standardized by stripping extra whitespace and converting them to lowercase, which guarantees consistency in feature referencing.

(2)- The target variable, identified as "strength," is separated from the feature set. Non-numeric columns such as the alloy or mix identifier are excluded from the analysis to focus solely on quantitative variables.

(3)- Standardization is performed using the StandardScaler from scikit-learn, which scales the feature values to have a zero mean and unit variance. This step is critical for many machine learning algorithms to function optimally, as it mitigates the influence of differing feature scales.

(4)- Finally, the preprocessed data is split into training and testing subsets (commonly using an 80/20 ratio) to facilitate unbiased model training and evaluation.

		А	E	3	С	D	Е	F	G	Н	I	
	1	cement	slag		ash	water	superplastic	coarseagg	fineagg	age	strength	
	2	141	212		0	203.5	0	971.8	748.5	28	29.89	
	3	169	42.2		124.3	158.3	10.8	1080.8	796.2	14	23.51	
	4	250	0		95.7	187.4	5.5	956.9	861.2	28	29.22	
	5	266	114		0	228	0	932	670	28	45.85	
	6	155	183.	4	0	193.3	9.1	1047.4	696.7	28	18.29	
	7	255	0		0	192	0	889.8	945	90	21.86	
	8	167	250.	2	0	203.5	0	975.6	692.6	7	15.75	
1	9	251	0		118.3	188.5	6.4	1028.4	757.7	56	36.64	
1	0	296	0		0	192	0	1085	765	28	21.65	
	A	В			С	D	E	F		G	Н	
1		cement		slag		ash	water	superplastic	coarseagg		fineagg	
2	cement	1.0		-0.2752	1591103032067	-0.3974673408069	-0.08158674846410	874 0.09238617275378	-0.1093489	9412490123	-0.22271784866558775	
3	slag	-0.275215911	03032067	1.0		-0.3235799005063	8535 0.107252027211092	264 0.04327041876104	1346 -0.2839986	119592135	-0.28160267061419253	
4	ash	-0.397467340	806901	-0.3235	5202724400264	1.0	-0.25698402286515	0.37750314590136	-0.0099608	2/842262055	0.07910849096499849	
6	superplasti	-0.001500740	40410074 75378024	0.1072	70418761041346	0.37750314590136	99 -0.65753290762844	-0.0575329076264	-0.2659991	1818628717	0.22269122991318685	
7	coarseagg	-0.109348994	12490123	-0.2839	986119592135	-0.0099608278422	62055 -0.18229360186143	498 -0.2659991481862	8717 1.0		-0.17848095742332418	
8	fineagg	-0.222717848	66558775	-0.2816	0267061419253	0.07910849096499	849 -0.45066117413001	49 0.22269122991318	685 -0.1784809	5742332418	1.0	
9	age	0.0819460238	87182248	-0.0442	4601930445421	-0.1543705160679	2992 0.27761822152100	-0.1927000280434	7272 -0.0030158	8034674382	-0.1560947026475875	
10	strength	0.4978319193	3241576	0.1348	2926149740518	-0.1057549162973	1361 -0.28963338498530	41 0.36607882718851	96 -0.1649346	1446010996	-0.1672412472900584	

Figure 7: The sample CSV file for data-driven experimentation task.

A.2 MATERIAL PROPERTY PREDICTION

The Figure 8 illustrates the history of interactions among different agents within the MatAgent framework during the material property prediction process. These interactions involve multiple AI agents, each responsible for specific tasks in the research workflow, guided by a Human-in-the-Loop (HITL) mechanism for validation and refinement.

The detailed discussion about the agent Interactions in this cycle is provided below:

A.2.1 HYPOTHESIS AGENT

- Proposes a research hypothesis that machine learning models can accurately predict the experimental band gap of materials based on their structural and electronic properties.
- Outlines the steps to test the hypothesis, including data collection, preprocessing, feature selection, model training, and evaluation using regression models (Linear Regression, Random Forest, Gradient Boosting, and SVR).
- Involves a decision-making step, where the human researcher chooses whether to regenerate the hypothesis or continue the research process.

A.2.2 PROCESS AGENT

- Receives the next task: integrating Python code and visualizations into the research report.
- Ensures that the Methodology, Results, and Discussion sections include R², MSE, and predicted vs. actual values to support the analysis.
- Logs the process of creating a research report document (Research_Report.md).

A.2.3 REPORT AGENT

- Drafts the research report and saves it as "Research_Report.md".
- Ensures that the report integrates Python code, visualizations, and key research findings, providing a comprehensive overview of the study.

A.2.4 QUALITY REVIEW AGENT

- Reviews the completed report for clarity, completeness, and alignment with the hypothesis.
- Confirms that it includes all necessary components, such as Python code, visualizations of R² and MSE, and predicted vs. actual values.
- Provides an opportunity for further modifications if needed.

This history of interactions exemplifies how MatAgent orchestrates machine learning-driven material property prediction in a streamlined, automated, yet human-guided manner.



Figure 8: The history of agents' interactions with each other for material property prediction.

As displayed in Figure 9, the Python script model_training_and_evaluation.py is designed to train and validate machine learning models for predicting the experimental band gap of materials. The script follows a structured workflow that includes data preprocessing, model training, evaluation, and visualization of results. The process begins with importing the necessary libraries, including pandas for data manipulation, numpy for numerical operations, matplotlib.pyplot and seaborn for visualization, and machine learning modules from sklearn, such as train_test_split for data splitting, LinearRegression, RandomForestRegressor, GradientBoostingRegressor, and SVR for model training, along with mean_squared_error and r2_score for performance evaluation.

The dataset is loaded from a CSV file (data.csv), and preprocessing steps are applied to ensure data integrity. Non-numeric columns are dropped, and missing values are handled using .dropna(), ensuring that only complete numerical data is used. The script then selects the features (X) and target variable (y), where the experimental_band_gap column is chosen as the target. To enable model training and testing, the dataset is split into training and testing subsets using an 80/20 split ratio with a random state of 42 for reproducibility. The script initializes four different regression models: Linear Regression, Random Forest, Gradient Boosting, and Support Vector Regression (SVR).

Each model is trained on the training set (X_train, y_train) and then evaluated on the test set (X_test, y_test). The script calculates two key performance metrics: Mean Squared Error (MSE) and R^2 Score, storing the results in a dictionary. After evaluation, the results are converted into a DataFrame and visualized using bar plots. The matplotlib library is used to create a side-by-side bar chart comparing the R^2 scores and MSE values of the different models, allowing for an intuitive comparison of their performance. Labels and titles are added for clarity, and the final visualization is saved as model_performance.png which is shown in Figure 10.

This script provides a streamlined and automated approach to training, evaluating, and visualizing machine learning models for material property prediction. By integrating multiple regression models and visualizing their performance, it helps researchers identify the most effective approach for predicting experimental band gaps based on material features.



Figure 9: The generated python code for training and validating the models.



Model Comparison: R² and MSE

Figure 10: The model performance comparisons in terms of R² and MSE metrics.

A.3 HYPOTHESIS GENERATION FOR NOVEL MATERIALS

The MatAgent develops a machine-learning-based framework to generate hypotheses for novel superconducting materials by predicting critical temperatures (T_c) . A large superconductors dataset (21,263 entries, 82 properties) undergoes thorough preprocessing (cleaning, scaling, imputation) and exploratory analysis, revealing strong correlations between T_c and factors such as electronegativity, atomic radii, and valence electron count. Multiple machine learning models (including Neural Networks, Random Forests, Support Vector Regression (SVR), and Linear Regression) are trained to predict T_c , with a neural network providing the highest accuracy ($R^2 = 0.92$). Further model performance investigation is disclosed in Table 2, and feature importance analysis highlights electronegativity, affinity, and valence electron properties as key predictors accordingly. Building on these models, the study's Hypothesis Agent proposes four main hypotheses:

1. Elemental Combinations (e.g., Cu, Ba, O) for Higher T_c .

Linear Regression

- 2. Valence Electron Count Positively Influences Superconductivity.
- 3. Electron Affinity and Electronegativity as Strong T_c Indicators.
- 4. Generative ML Approaches Can Design Novel Superconductors.

As depicted in Figure 11, the visualizations (feature importance plots, critical temperature distributions, correlation heatmaps) further validate these hypotheses, underscoring MatAgent's potential to accelerate the discovery of high- T_c superconductors through targeted, data-driven predictions.

Model R² Score MAE MSE Random Forest Regressor 0.91 4.2 8.5 Neural Network (MLPRegressor) 0.92 4.0 8.0 Support Vector Regression (SVR) 0.88 4.8 9.2

0.79

5.5

11.3

Table 2: Model Performance for Superconductor Critical Temperature Prediction



Figure 11: MatAgent outcome for Superconductor critical temperature prediction.

The Figure 12 explains the history of interactions among different agents within the MatAgent framework during the hypothesis generation process for novel materials. These interactions involve multiple AI agents, each responsible for a distinct task, following a structured workflow to generate hypotheses, analyze materials, and validate findings under a Human-in-the-Loop (HITL) approach to ensure meaningful insights.

The process begins with the Hypothesis Agent, which formulates three core hypotheses regarding the relationship between material composition and superconducting critical temperature (T_c) , the role of material properties in predicting superconductivity, and the potential for generative modeling to design new superconducting materials. To test these hypotheses, the agent outlines key steps, including data collection and preprocessing, exploratory data analysis (EDA), feature selection, model development, generative modeling, and visualization/reporting. The HITL mechanism allows the researcher to decide whether to regenerate the hypothesis or continue with the research process.

Once the hypotheses are finalized, the Visualization Agent is responsible for creating graphical representations of the analysis. It generates and saves three key visualizations: the distribution of critical temperatures, which provides insights into the frequency distribution of superconducting materials across temperature ranges; a correlation heatmap, which identifies relationships between material properties and critical temperature; and a feature importance analysis, which ranks the most influential factors in predicting superconducting behavior. These visualizations play a crucial role in understanding patterns in the data and supporting hypothesis validation.

Following this, the Quality Review Agent evaluates the research report to ensure clarity, consistency, and adherence to scientific writing standards. The review confirms that the report is well-structured, aligns with the research objectives, and includes all necessary visualizations, interpretations, and discussions. Since no major revisions are required, the project proceeds to the finalization stage.

The Process Agent is the last in the workflow, ensuring that all research components are cohesively integrated. It verifies that the hypothesis is rigorously tested, the data analysis is complete and reproducible, and all required visualizations are appropriately documented. The final report is polished, logically structured, and meets high academic standards, concluding the research process.

This structured workflow demonstrates how multiple AI agents collaborate to automate and refine the hypothesis generation process for novel materials. By integrating machine learning, visualization, and quality assurance mechanisms, MatAgent streamlines research while maintaining human oversight, ensuring that hypotheses are data-driven, well-documented, and scientifically robust.





Figure 12: The history of agents' interactions with each other for hypothesis generation.

According to Figure 13, the Python script model_training.py trains machine learning models to predict the critical temperature (T_c) of superconducting materials. It follows a structured workflow, including data preprocessing, feature standardization, model training, evaluation, and data exploration.

The script begins by importing necessary libraries, including pandas for data handling, sklearn for model selection, feature scaling, and evaluation metrics, as well as numpy for numerical computations. It then loads a cleaned dataset (superconduct_train_cleaned.csv), separates features (X) and the target variable (y), and splits the dataset into training (80%) and testing (20%) subsets.

To ensure uniform feature scaling, a StandardScaler is applied to both the training and testing sets, transforming them into a normalized range. The first machine learning model implemented is a Support Vector Machine (SVM) with an RBF kernel, which is trained on the scaled training data. Predictions are made on the test set, and model performance is evaluated using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE).

The script also trains a Neural Network using an MLPRegressor with a hidden layer of 100 neurons and a maximum of 500 iterations. Like the SVM, it is trained on the scaled training data, makes predictions on the test set, and is evaluated using MAE and RMSE. The evaluation results for both models (svm_mae, svm_rmse, nn_mae, nn_rmse) are stored for comparison.

Additionally, a separate section of the script performs dataset exploration by loading another dataset (superconduct_train_new.csv), ensuring proper data types and handling potential formatting inconsistencies. It cleans non-numeric characters, converts data to a numerical format, and generates dataset summaries, including descriptive statistics, missing value checks, and a correlation matrix. These exploratory steps provide insights into the dataset structure, aiding in better preprocessing before model training.

This script automates training, evaluation, and analysis of machine learning models for superconductivity prediction. By integrating Support Vector Machines and Neural Networks, it compares model performances and facilitates data-driven hypothesis generation for discovering novel superconducting materials.



Figure 13: The generated python code for training the models.

The Figure 14 presents three key visualizations that provide critical insights into the relationships between material properties and their influence on superconducting critical temperature (T_c) . These visualizations include a feature importance ranking, a correlation heatmap, and the distribution of critical temperatures, each serving a unique role in understanding the factors that drive superconducting behavior.

The feature importance plot (top left) ranks the most influential material properties in predicting T_c , based on their contributions to the predictive machine learning model. The ranking demonstrates that a subset of features, such as atomic radius, electronegativity, and valence electron count, play a dominant role in determining superconducting properties, with a steep decline in importance for other features. This suggests that while many material characteristics influence superconductivity, only a few exhibit a strong predictive relationship with T_c . Identifying these critical features enables researchers to focus on the most impactful variables when designing new superconducting materials, improving both efficiency and accuracy in material discovery.

The correlation heatmap (top right) provides an overview of how different material properties interact with one another. This matrix visualization employs a color scale, where red represents strong positive correlations and blue indicates negative correlations, allowing for a clear understanding of feature interdependencies. The presence of strong correlations between certain features, such as atomic mass, atomic radius, and valence electron properties, suggests that these factors tend to vary together, potentially influencing superconductivity in related ways. By identifying highly correlated features, researchers can refine their predictive models by reducing redundancy and improving feature selection. Furthermore, this visualization supports the hypothesis that electronic and structural properties play a crucial role in determining superconducting behavior.

The distribution of critical temperatures (bottom left) provides insights into how superconducting materials are spread across different temperature ranges. The histogram reveals a high density of materials with relatively low T_c values (below 100 K), while only a small fraction exhibit significantly higher critical temperatures. This skewed distribution suggests that the majority of known superconductors operate at relatively low temperatures, aligning with the well-documented challenge of achieving high T_c superconductivity. The presence of a long tail in the distribution, with some materials exhibiting exceptionally high T_c values, highlights potential candidates for further investigation, as they may possess unique structural or electronic characteristics that enable higher-temperature superconductivity. These insights can help guide researchers in prioritizing the most promising materials for advancing high T_c superconductor development.

Together, these visualizations provide a comprehensive and data-driven approach to understanding superconductivity, offering valuable insights into the key material properties that govern T_c . The feature importance analysis identifies dominant predictive variables, the correlation heatmap reveals interdependencies among material properties, and the critical temperature distribution underscores the challenges and opportunities in high T_c material discovery. These findings contribute to the broader objective of accelerating the discovery of novel superconductors through Data-driven hypothesis generation and materials informatics.



Figure 14: The correlation heatmap and visualized feature importance outputs.

A.4 EXPERIMENTAL DATA ANALYSIS

MatAgent emphasizes the pivotal role of experimental dataset analysis in understanding how materials behave under different temperature conditions, specifically focusing on gravimetric and molar heat capacities between $250 \,^{\circ}$ C and $400 \,^{\circ}$ C, as shown in Figure 15. After thorough data preprocessing—including imputation of missing values, removal of duplicates, and numerical standardiza-

tion—exploratory analyses reveal near-normal distributions of heat capacity measures, with mean values rising at higher temperatures, thereby confirming the thermal dependence of heat capacity. A strong positive correlation emerges between gravimetric and molar heat capacities, suggesting their interdependence in characterizing heat retention. Accordingly, with the help of the MatAgent, machine learning is introduced to extend these findings, where feature engineering and a Random Forest Classifier are used to identify temperature-driven material behavior. The model pinpoints molar heat capacity at $250 \,^{\circ}$ C as a key predictor and achieves high classification accuracy (>85%), underscoring the viability of predictive modeling in classifying materials based on their thermal properties. Taken together, these insights underline the potential for leveraging correlated heat capacity trends to enhance material design and optimization, with applicability in thermal management solutions for fields such as energy storage, aerospace, and electronics.



Figure 15: MatAgent query-response result for streamlining the experimental data analysis.

Figure 16 illustrates the history of interactions among multiple AI agents within the MatAgent framework during the Experimental Data Analysis process. These agents collaborate to ensure a structured, automated approach to analyzing experimental data, refining findings, and enhancing the quality of the research report. The workflow follows an iterative process involving the Process Agent, Searcher Agent, Report Agent, and Quality Review Agent, each contributing to different stages of research documentation and validation.

The process begins with the Process Agent, which is responsible for compiling references that support the research methodologies and findings. It ensures that appropriate citations are included to strengthen the scientific credibility of the report. The agent logs multiple operations, including searching for related works, extracting relevant literature, and integrating references into the research document. These steps help align the study with existing knowledge in the field.

The Searcher Agent then takes over by retrieving references and summarizing key insights from relevant scientific articles. It identifies essential points from various sources, such as the impact of temperature on biological shifts and chemical reactions, statistical analysis of temperature data, and using machine learning to predict temperature effects. By incorporating these references, the research gains depth and scientific validation, ensuring that its conclusions are supported by existing literature.

Following the integration of references, the Report Agent evaluates the research document and identifies areas for improvement. It highlights key concerns, including the clarity of the narrative, completeness of the methodology, visualization of results, logical coherence in the discussion section, and proper citation formatting. The agent suggests refining these aspects to ensure the report meets high academic standards.

The Quality Review Agent conducts the final validation of the research report. It assesses the accuracy of interpretations, organization of findings, depth of discussion, and proper reference formatting. The agent also ensures that the results are presented with appropriate visualizations, tables, and structured explanations, improving the report's readability and impact.

This structured collaboration between agents enables efficient and systematic analysis of experimental data, refining the research process while maintaining scientific rigor. By integrating automated literature retrieval, reference validation, report structuring, and quality assurance, MatAgent ensures that experimental findings are well-documented, data-driven, and aligned with established research methodologies.

Process Agent	
{'next': 'Search', 'task': 'Compile a com	nprehensive list of references that support the methodology and findings discussed in the research report. Ensure that
all sources are credible and relevant to	the analysis conducted.
2025-01-19 21:53:12,737 - logger - INF	O - Performing Google search for query: temperature response variable analysis methodology
2025-01-19 21:53:20,215 - logger - INF	O - Google search completed successfully
2025-01-19 21:53:25,220 - logger - INF	O - Performing Google search for query: temperature effect on response variable academic articles
2025-01-19 21:53:30,673 - logger - INF	O - Google search completed successfully
2025-01-19 21:53:31,470 - logger - INF	O - Performing Google search for query: temperature response variable correlation studies
2025-01-19 21:53:36,829 - logger - INF	O - Google search completed successfully
2025-01-19 21:53:41,745 - logger - INF	O - Performing Google search for query: temperature impact on experimental results peer-reviewed articles
2025-01-19 21:53:47,587 - logger - INF	O - Google search completed successfully
2025-01-19 21:53:52,399 - logger - INF	O - Performing Google search for query: machine learning temperature response variable analysis
2025-01-19 21:53:57,878 - logger - INF	O - Google search completed successfully
2025-01-19 21:54:13,863 - logger - INF	O - Creating document: data_storage\temperature_response_variable_references.md
2025-01-19 21:54:13,866 - logger - INF	O - Document created successfully: data_storage\temperature_response_variable_references.md
Searcher Agent	
 I have compiled a list of references t summarized and saved in a Markdown Here are the references included: 	hat support the methodology and findings discussed in the research report. The key points from the articles are document titled "temperature_response_variable_references.md".
1. **The Effect of Temperature on th responses.	te Response of Biological Systems**: Discusses how temperature variations can significantly influence biological
2. **Statistical Methods for Analyzin	ig Temperature Effects in Experimental Data**: Outlines various statistical techniques for analyzing the impact of
temperature on experimental outcome	s.
 3. **Temperature and Its Effects on C on experimental methodologies. 	Chemical Reactions: A Review**: Reviews how temperature affects chemical reaction rates and outcomes, focusing
 4. **Machine Learning Approaches techniques to predict how temperature 	for Predicting Temperature Effects on Response Variables**: Explores the application of machine learning e influences various response variables.
5. **Correlation Between Temperatur response variables in environmental re	re and Response Variables in Environmental Studies**: Analyzes the correlation between temperature and various search.
The document is saved in the data store	age directory. If you need any further information or additional references, please let me know!
The document is saved in the data stor	age directory. If you need any further information or additional references, please let me know!



Figure 16: The history of agents' interactions with each other for experimental data analysis.

Figure 17 presents the Python script automated_analysis.py, which automates the preprocessing and visualization of experimental data, specifically for analyzing gravimetric heat capacities at different temperatures. This structured workflow includes data loading, cleaning, transformation, and visualization, ensuring that the dataset is prepared for further analysis in a streamlined manner.

The script begins by importing essential libraries, including pandas for data manipulation, matplotlib.pyplot for visualization, and seaborn for enhanced plotting aesthetics. The dataset, stored as IZA_cp.csv, is then loaded into a Pandas DataFrame, making it accessible for analysis. The data cleaning process follows, starting with an inspection of the dataset by displaying the first few rows using df.head(). The script then checks for missing values using df.isnull().sum(), printing the number of missing values per column. To handle missing data, the script fills missing numerical values with their respective column means using df.fillna(df.mean(numeric_only=True), inplace=True), ensuring that incomplete records do not interfere with subsequent analysis.

Next, the script identifies duplicate entries using df.duplicated().sum(). If duplicates are found, they are removed using df.drop_duplicates(inplace=True), preventing redundant data from skewing the results. It also checks for inconsistent data types by printing df.dtypes, which allows for the identification of incorrectly formatted columns. To ensure numerical data is processed correctly, the script attempts to convert all relevant columns to float format while skipping non-numeric columns, such as names. Any conversion errors are caught, and affected columns are flagged for further inspection. Once cleaned, the processed dataset is saved as a new CSV file (cleaned_IZA_cp.csv) using df.to_csv(), preserving the refined data for future use.

The final step in the workflow is data visualization. The script uses seaborn to generate histograms of gravimetric heat capacities at various temperatures. The histograms are plotted for a range of temperatures (250°C to 400°C), capturing the distribution of gravimetric heat capacities across different thermal conditions. Each histogram is labeled with the corresponding temperature, and the figure is saved as histograms_gravimetric_heat_capacity.png.

This automated approach ensures efficient, reproducible experimental data analysis by handling missing values, removing inconsistencies, and generating meaningful visualizations. The streamlined workflow enhances data reliability while providing clear insights into material behavior across different temperature conditions, aiding in the broader goal of experimental data-driven materials research.

automated_analysis.py	
<pre># Import the libraries import pandas as pd import matplotili-pypiot as pit import seaborn as sans # Load the dataset file_path = iZA_cp.csv' d = pdread_csv(file_path) # Date Cleaning # Display the first few rows of the dataframe print(Missing values in each column:) # Hondet missing values by filling with the mean of the numeric columns dt filina(dt.mean(numeric_only=True), inplace=True) # Check for duplicates {inpy if duplicates > 0:</pre>	<pre># Canvert data types if necessary (a.g., ensure numeric columns are floats) for col in dt.columns[1:]: # Skip the first column which is name' try; df(col] = df(col].astype(float) except ValueError: print("Column (col) could not be converted to float.") # Display cleaned data print("Cleaned Data:") print("Cleaned Data:") print("Cleaned Data:") print("Cleaned Data:") print("Cleaned Data: ") print("Cleaned data to a new CSV file cleaned.file_path = cleaned.ZA_cp.csv' dtl.o_csv(cleaned.file_path, index=False) print("Cleaned data saved to (cleaned_file_path)") # Data Visuelization # Set the style of seaborn sns.set(style=whitegrid) # Create histograms for gravimetric heat capacities at different temperatures plt.figure(figsize=(155, 00) for i, temp in enumerate(250, 275, 300, 325, 350, 375, 400)): plt.subplot(3, 3, i+1) sns.histol(tiff("Cv.gravimetric_(temp).00_mean], bins=20, kde=True) plt.title(!"Histogram of Cv.gravimetric at (temp)"C') plt.tibplot(#G("Faquency)) plt.tighc("havguru) plt.subw()</pre>

Figure 17: The generated python code for automating the experimental data analysis.

The visualizations in Figure 18 provide a detailed analysis of gravimetric and molar heat capacities across different temperatures, aiming to uncover patterns and correlations in the experimental dataset. The key visual elements include histograms of gravimetric heat capacities, scatter plots comparing gravimetric and molar heat capacities, and a correlation heatmap.

The histograms (top-left) display the distribution of gravimetric heat capacities at temperatures ranging from $250 \,^{\circ}$ C to $400 \,^{\circ}$ C. These histograms reveal that the heat capacity values exhibit nearnormal distributions, with mean values gradually increasing as temperature rises. The smooth kernel density estimation (KDE) overlay further illustrates the spread and concentration of values, indicating that materials exhibit consistent thermal behavior across temperature variations. This suggests that gravimetric heat capacity follows a predictable trend with changing temperature.

The scatter plots (bottom-left) compare gravimetric heat capacity (Cv) with molar heat capacity at multiple temperature points. Each scatter plot exhibits a strong linear relationship between these two properties, as demonstrated by the near-perfect alignment of data points along a straight line. This finding reinforces the interdependence between gravimetric and molar heat capacities, indicating that one can be reliably predicted from the other. The strong correlation suggests that molar heat capacity scales proportionally with gravimetric heat capacity, further validating the thermodynamic consistency of the dataset.

The correlation heatmap (right side) provides a comprehensive overview of the relationships between heat capacity measurements at different temperatures. The color-coded matrix represents correlation coefficients, where red indicates strong positive correlations, blue represents negative correlations, and white denotes weaker relationships. The heatmap reveals high correlations (close to 1.0) between heat capacities measured at adjacent temperatures, suggesting that heat capacity values remain stable across thermal ranges. Additionally, the clustering of similar values along the diagonal highlights the thermal continuity in material behavior, reinforcing the assumption that heat capacities evolve predictably with increasing temperature.

Together, these visualizations offer a comprehensive experimental analysis of heat capacity trends. The histograms validate the normality and progression of heat capacity values, the scatter plots confirm strong linear dependencies between gravimetric and molar heat capacities, and the correlation heatmap demonstrates the thermal consistency of heat capacity measurements across temperature ranges. These insights support data-driven decision-making in materials science, particularly in predicting thermal properties and optimizing material compositions for specific temperature conditions.



Figure 18: The visualized results concerning the automation of the experimental data analysis.

A.5 ACCELERATED DISCOVERY OF HIGH-PERFORMANCE ALLOYS OR POLYMERS

MatAgent also highlights how machine learning techniques, combined with density functional theory (DFT) calculations, can accelerate the discovery and optimization of high-performance alloys by predicting their yield strength—a critical property in aerospace, automotive, and structural appli-

Model	R ² Score	MAE	MSE	
Random Forest Regressor	0.89	85 MPa	12000 MPa ²	
Gradient Boosting Machine (GBM)	0.91	78 MPa	10000 MPa ²	
Support Vector Regression (SVR)	0.85	92 MPa	14000 MPa ²	

Table 3: Model Performance for	Yield Strength	Prediction
--------------------------------	----------------	------------

cations. As can be seen in Figure 19, a key dataset (Alloy-Yield-Strength.csv) is curated to include lattice constants, electronic properties, and yield strength values (YS in MPa). Data preprocessing addresses missing values, outliers, and formatting inconsistencies, ensuring reliability for downstream analysis. Exploratory analysis reveals a bimodal yield strength distribution (low vs. high strength alloys), with a noteworthy correlation between smaller lattice constant variations and higher yield strength. Gradient Boosting Machines (GBM) emerges as the top-performing model, yielding an R^2 of 0.91 and a MAE of 78 MPa, surpassing Random Forest and Support Vector Regression (SVR) for which their further details are given in Table 3. DFT calculations provide complementary insights, indicating that lower Fermi-level occupancy and metallic bonding interactions (e.g., in Ir–Rh alloys) contribute to stronger atomic bonds and enhance yield strength.



Figure 19: Experimenting MatAgent by combining the ML and DFT for alloy discovery.

The Figure 20 illustrates the history of interactions among various agents within the MatAgent framework during the accelerated discovery of high-performance alloys or polymers. This structured multi-agent system automates hypothesis generation, material property prediction, and scientific documentation while incorporating a human-in-the-loop (HITL) validation mechanism to refine insights and ensure the accuracy of results.

The process begins with the Hypothesis Agent, which formulates three core hypotheses focused on optimizing yield strength in advanced alloys. The first hypothesis proposes that machine learning models combined with theoretical calculations can accurately predict and optimize yield strength by identifying key material properties and optimal processing conditions. The second hypothesis suggests a strong correlation between specific material properties (e.g., lattice constants, mixing enthalpy) and yield strength, which can be visualized through correlation heatmaps. The third hy-

pothesis asserts that feature importance rankings from machine learning models can automate the identification of critical factors influencing yield strength, thus streamlining alloy development.

To test these hypotheses, the agent establishes a structured methodology that includes:

- Data Collection and Preprocessing: Gathering material property data from the provided dataset (Alloy_Yield_Strength.csv).
- Theoretical Calculations: Computing key material properties, such as lattice constants and mixing enthalpy.
- Machine Learning Model Development: Training predictive models (Random Forest, Gradient Boosting) to estimate yield strength based on identified material properties.
- Correlation Analysis: Generating heatmaps to visualize the relationships between material properties and yield strength.
- Feature Importance Ranking: Using techniques like SHAP (Shapley Additive Explanations) or permutation importance to determine the most influential features.
- Yield Strength Distribution Analysis: Identifying trends and variability in the dataset.
- Automation of Candidate Material Identification: Developing an automated pipeline that integrates theoretical calculations and machine learning predictions to discover optimal alloy compositions.
- Validation Experiments: Proposing experimental methods to confirm predictions based on candidate materials and processing conditions.

After the hypothesis formulation and Human-in-the lop, the Process Agent plays a crucial role in ensuring that findings from the literature review are seamlessly integrated into the research report. It retrieves and processes multiple documents related to machine learning applications in material property prediction, ensuring that the research is grounded in existing scientific knowledge. The logs indicate that relevant reports were accessed, read, and edited to align the study with current advancements in the field.

Once the literature review and findings are compiled, the Report Agent drafts a structured document, "Research_Report_Optimizing_Yield_Strength.md", incorporating insights into the introduction and methodology sections. The agent ensures that machine learning advancements in predicting material properties are explicitly connected to the research hypothesis, while also verifying that references are correctly cited to maintain scientific integrity.

The Quality Review Agent then assesses the final research report for clarity, coherence, and adherence to scientific writing standards. It confirms that the machine learning methodologies used for predicting material properties are well-documented, logically structured, and properly referenced. Since no further revisions are deemed necessary, the project proceeds to finalization.

This structured workflow highlights how MatAgent efficiently integrates machine learning, theoretical calculations, and systematic documentation to accelerate high-performance alloy discovery. The automated yet expert-guided approach ensures that hypothesis testing, predictive modeling, and material optimization remain data-driven, accurate, and scientifically validated, making it a powerful tool for advancing alloy development in engineering and materials science.



Figure 20: The history of agents' interactions with each other for alloy and polymer discovery.

The Python script optimize_model.py which is shown in Figure 21 is designed to optimize a machine learning model for predicting the yield strength of alloys using Random Forest Regression. It follows a structured workflow that includes data preprocessing, model training, hyperparameter tuning using GridSearchCV, and performance evaluation to ensure optimal predictive performance.

The script begins by importing necessary libraries, including pandas and numpy for data manipulation, sklearn modules for data splitting, preprocessing, and model selection, and RandomForestRegressor for regression tasks. It also imports GridSearchCV, a technique for hyperparameter tuning to improve model performance.

The data preprocessing pipeline starts by loading the dataset (Alloy_Yield_Strength.csv) into a Pandas DataFrame. The column names are cleaned by stripping whitespace and converting them to lowercase to maintain consistency. The script then identifies numeric and categorical columns, ensuring that numerical data is properly standardized and categorical data is encoded for model training. Missing values in numerical columns are filled with their mean values, ensuring that incomplete records do not interfere with training.

To facilitate efficient preprocessing, the script defines two transformation steps:

- Numerical columns are standardized using StandardScaler.
- Categorical columns are encoded using OneHotEncoder.

These transformations are combined into a preprocessing pipeline, ensuring that all data is uniformly processed before being fed into the machine learning model.

The model training process begins by defining the Random Forest Regressor with a fixed random state for reproducibility. A pipeline is created to integrate the preprocessing steps and the model into a unified workflow. The dataset is then split into training and testing sets (80/20 split) to evaluate model performance on unseen data.

To enhance predictive accuracy, the script employs GridSearchCV for hyperparameter tuning, optimizing key parameters such as the number of estimators, maximum depth, and maximum features. The hyperparameter search iterates over multiple configurations, selecting the combination that yields the best model performance based on cross-validation scores.

Once the best model is identified, its performance is evaluated on the test set using key regression metrics MAE, MSE, R² Score. The final results, including the best-performing model and evaluation metrics, are stored and printed for further analysis.

This script efficiently automates the optimization of machine learning models for material property prediction, ensuring that the model is properly tuned, systematically preprocessed, and rigorously evaluated. By integrating data cleaning, feature engineering, hyperparameter tuning, and performance assessment, it streamlines the discovery of high-performance alloys with optimized yield strength.



Figure 21: The generated code for model optimization purpose.

The visualized results for yield stress analysis in Figure 22 provide key insights into the relationship between material properties and yield strength distribution. The two plots illustrate different perspectives: one focuses on the correlation between yield strength and lattice constant differences, while the other captures the overall distribution of yield strength values.

The scatter plot (left) presents the relationship between yield strength (MPa) and the difference in lattice constants. Each point represents a data sample, and its position indicates the lattice constant difference and corresponding yield strength. The majority of points are concentrated at low lattice constant differences (close to 0.0–0.5), spanning a broad range of yield strength values from approximately 500 MPa to over 2500 MPa. This suggests that small lattice constant differences do not strictly determine yield strength, as samples within this range exhibit high variability. However,

at higher lattice constant differences (above 1.5), a trend of lower yield strength values is observed, indicating that greater lattice distortions may lead to structural weaknesses in alloys. While the trend is not entirely linear, the spread of data suggests that other factors, such as elemental composition or processing conditions, might influence yield strength alongside lattice constants.

The histogram (right) illustrates the distribution of yield strength values across the dataset. The histogram reveals a bimodal-like distribution, with most materials exhibiting yield strength values between 1000 MPa and 1700 MPa. The highest frequency occurs around 1500 MPa, indicating that a significant proportion of the dataset consists of mid-to-high-strength alloys. A smaller subset of alloys exhibits exceptionally high yield strength values (above 2500 MPa), suggesting the presence of specialized or high-performance materials. Meanwhile, there are relatively fewer materials with yield strength below 1000 MPa, which could indicate that low-strength alloys are either less common in the dataset or less desirable in high-performance applications.

These visualizations provide a data-driven perspective on yield strength optimization. The scatter plot highlights how lattice constant differences relate to strength variability, while the histogram reveals the common and extreme yield strength values within the dataset. These insights can inform machine learning models, material selection processes, and alloy design strategies to enhance high-performance material discovery.



Figure 22: The visualized results for yield stress analysis.

A.6 DATA-DRIVEN EXPERIMENTATION

The application of machine learning (ML) in materials science is transforming conventional experimentation by enabling predictive modeling for optimizing material properties. This study, as visually outlined in Figure 23, explores a data-driven approach to optimizing concrete mix designs by predicting compressive strength using ML techniques. The dataset comprises various concrete components, including cement, slag, ash, water, superplasticizer, coarse aggregate, fine aggregate, and curing age, which directly influence mechanical performance. The research objectives include identifying key influencing factors, developing predictive ML models for strength estimation, optimizing mix proportions, and validating model performance through comparative analysis.

The dataset was preprocessed by handling missing values, applying MinMax scaling for feature standardization, and splitting it into training (80%) and testing (20%) sets. Exploratory Data Analysis (EDA) using correlation heatmaps revealed cement and age as the strongest positive predictors (0.50 and 0.33, respectively), while water content negatively correlated (-0.29), reinforcing that excessive water weakens concrete. Superplasticizer exhibited a moderate positive correlation (0.37), supporting workability without excessive water. The distribution of strength values showed a unimodal trend, peaking between 30–50 MPa, with a few high-performance concrete samples exceeding 70 MPa. For predictive modeling, Linear Regression (baseline), Random Forest, and Gradient Boosting Regressor (GBR) were implemented. Performance evaluation was based on Root Mean Square Error (RMSE) and R² scores, as summarized below.

As shown in Table 4, Gradient Boosting achieved the best performance, exhibiting the lowest Root Mean Squared Error (RMSE) of 5.2 MPa and the highest R-squared (R²) of 0.93. This clearly demonstrates Gradient Boosting's superior capability in capturing complex interactions within the

data. Random Forest also performed well with a high R^2 of 0.91, further validating the non-linearity inherent in compressive strength prediction. In contrast, Linear Regression showed the lowest accuracy, evidenced by its considerably lower R^2 of 0.76. This performance disparity indicates that accurate compressive strength prediction necessitates non-linear models like Gradient Boosting and Random Forest, as linear approaches are inadequate to capture the underlying complexities.

Feature importance analysis identified curing age and cement content as the most influential factors, with water negatively impacting strength and superplasticizer playing a moderate role in enhancing workability while maintaining strength. Visualizations further confirmed the distribution of compressive strength and superior accuracy of GBR over other models.

The findings highlight ML's potential for real-time strength prediction, reducing reliance on traditional testing. Predictive models facilitate optimized mix designs, ensuring cost-effective and highperformance concrete formulations. Future work should incorporate environmental factors (e.g., temperature, humidity) and reinforcement effects for enhanced generalization. This study demonstrates how ML can transform material optimization, with GBR emerging as the most effective model for compressive strength prediction.



Figure 23: The visualized query-response for automated data-driven experimentation.

Model	RMSE (MPa)	R ² Score
Linear Regression	10.5	0.76
Random Forest	5.8	0.91
Gradient Boosting	5.2	0.93

Table 4: Model Performance for Concrete Strength Prediction

The visualized results in Figure 24 provide key insights into data-driven experimentation for concrete strength prediction, showcasing correlations between material properties, feature importance rankings, distribution of concrete strength, and model performance comparisons. These analyses are crucial in understanding the factors influencing concrete strength and optimizing predictive modeling. The correlation heatmap (top-left) illustrates the relationships between various material components (e.g., cement, slag, water, superplasticizer, aggregates) and concrete strength. The color scale represents correlation coefficients, where red indicates strong positive correlations and blue indicates negative or weak correlations. Cement and age exhibit strong positive correlations with concrete strength, suggesting that cement content and curing time are critical factors. In contrast, water shows a negative correlation, indicating that higher water content may weaken concrete strength.

The feature importance bar chart (top-right) ranks the most influential features in predicting concrete strength. Age and cement content are the top two contributors, reinforcing findings from the correlation analysis. Other notable features include superplasticizer and water content, which also play significant roles in influencing strength variations.

The histogram (bottom-left) visualizes the distribution of concrete strength across the dataset. The data follows a somewhat normal distribution, with most samples having strengths between 20 MPa and 50 MPa, while a smaller subset extends beyond 70 MPa. This distribution helps in identifying common strength ranges and potential outliers.

The model performance comparison chart (bottom-right) evaluates the effectiveness of different machine learning models—Linear Regression, Random Forest, and Gradient Boosting—in predicting concrete strength. The blue bars represent RMSE (Root Mean Squared Error), while the red line represents the R² score. While Random Forest achieves the highest R² score (0.90), Linear Regression performs the worst, demonstrating that non-linear models like Random Forest and Gradient Boosting better capture complex relationships in concrete strength prediction.

These visualizations provide a comprehensive, data-driven approach to optimizing concrete strength prediction, identifying key influencing factors, and assessing model performance. The insights gained can guide material formulation improvements and predictive modeling strategies for enhancing construction quality and durability.



Figure 24: The visualized results for concrete strength analysis.

A.7 LITERATURE REVIEW AND KNOWLEDGE SYNTHESIS

MatAgent leverages automated literature review and knowledge synthesis techniques, as exemplified in Figure 25, to evaluate optimization strategies for perovskite solar cells (PSCs). Figure 25 illustrates the systematic process of automated literature analysis, where a comprehensive review of existing research was conducted to identify key trends and supporting evidence. By integrating data mining, statistical analysis, and visualization tools, the research identifies key performance trends, highlights material composition impacts, and suggests optimization pathways to improve solar cell efficiencies. A core hypothesis is that targeted optimization strategies—especially material composition modifications and interface engineering—can significantly enhance PSC efficiency compared to baseline configurations.

The literature review, conducted using automated agents, collected and analyzed scientific articles, patents, and technical reports from major photovoltaic research databases. Key extracted themes include material composition optimization, emphasizing the role of triple-cation perovskites (Cs/FA/MA-based) in efficiency enhancement, passivation and interface engineering strategies for reducing recombination losses, and machine learning for efficiency prediction, including AI-assisted modeling of bandgap tuning and crystal structure optimization. A quantitative approach was employed to compare solar cell efficiencies across different optimization strategies using statistical tools such as ANOVA and t-tests. Efficiency analysis, as presented in Table 5, revealed that advanced material composition modifications provided the highest efficiency gain (24.5%), reinforcing the critical role of material design in PSC performance improvement. Comparatively, parasitic resistance minimization yielded the lowest efficiency improvement (18.5%), suggesting diminishing returns from electrical resistance optimizations compared to material-driven enhancements.

Visualization insights further validated the findings. A bar plot comparing optimization strategies confirmed the superiority of material composition over geometric and light management modifications. Current-voltage (I-V) characteristic analysis illustrated the enhanced voltage-current response of optimized cells, particularly those with material composition modifications. Additionally, efficiency trends over time were analyzed, highlighting progressive improvements from various optimization pathways.

The study's findings align with prior research indicating that iodide passivation and triple-cation perovskites significantly boost efficiency, though scalability challenges remain for large-area PSC modules, necessitating future work on long-term stability enhancements. The integration of AI-driven literature review demonstrates its effectiveness in synthesizing research trends and guiding photovoltaic material design. Future research should focus on scaling perovskite technologies for commercial adoption, AI-driven design of novel perovskite compositions, and addressing long-term degradation issues through interface engineering. This work underscores the potential of automated literature synthesis in accelerating data-driven material discovery and optimization, facilitating the development of next-generation high-efficiency perovskite solar cells.



Figure 25: The sample query-response for automated literature review with code and visualizations.

Optimization Strategy	Efficiency (%)
Minimize Parasitic Resistance	18.5
Enhance Light Management	20.3
Optimize Device Geometry	20.45
Advanced Material Composition	24.5

Table 5: Efficiency Performance of Optimization Strategies

The Figure 26 illustrates the history of interactions among multiple agents within the MatAgent framework during the literature review and knowledge synthesis process for optimizing perovskite solar cells. This collaborative workflow enables efficient knowledge extraction, analysis, and report generation, ensuring that scientific findings are properly integrated into the research.

The process begins with the Searcher Agent, which compiles relevant literature on perovskite material optimization and crystallization for solar cell applications. The agent presents three key research papers, each highlighting different aspects of perovskite advancements:

- Synthesis and Optical Characterization of Perovskite Layers discusses how solvent engineering influences photovoltaic efficiency and crystallization quality, emphasizing the role of different solvents in controlling band gaps.
- Lead-Free Perovskites focuses on material innovations that remove lead content, addressing band gap tuning through B-site cation replacement and its impact on achieving high efficiency.
- The Reality of Perovskite/Si Tandem Solar Cells evaluates the performance of tandem solar cells under real-world conditions, showing that while they outperform silicon solar cells in ideal settings, their advantage diminishes under practical conditions.

The Process Agent then integrates these findings into the research report. It retrieves existing documents and ensures that the literature review covers recent advancements and efficiency optimization strategies. The logs indicate that it successfully reads, processes, and edits the research report, maintaining clarity and coherence in referencing. Once the literature review is incorporated, the Visualization Agent generates efficiency comparison charts that illustrate the impact of different optimization strategies on perovskite solar cells. The visualizations provide quantitative insights by mapping efficiency percentages against various optimization techniques, helping to contextualize the literature findings. The agent confirms that all visualizations are stored as efficiency_improvements_perovskite_solar_cells.png, making them available for further analysis.

Following this, the Quality Review Agent assesses the final research report and suggests refinements to improve clarity, structure, and coherence. It recommends:

- Enhancing visualization clarity by ensuring proper labeling, titles, and axes.
- Providing a more detailed explanation of results, linking visualization insights to literature findings.
- Strengthening discussion cohesion, ensuring that the literature review connects well with the reported results.
- Ensuring formatting consistency in font sizes, headings, and citation styles.
- Conducting a final proofreading to eliminate grammatical or typographical errors.

This structured workflow demonstrates how MatAgent streamlines literature review and knowledge synthesis, ensuring that scientific findings are efficiently extracted, visualized, and integrated into research reports. By automating knowledge retrieval, visualization, and quality assessment while maintaining human oversight (HITL mechanism), the system ensures that research outputs are accurate, clear, and well-structured for further scientific dissemination.



Figure 26: The collaborative agents for automated literature review and knowledge synthesis.

The Python script data_analysis.py which shown in Figure 27 is designed to perform data preprocessing, statistical analysis, and visualization related to perovskite solar cell efficiency improvements and current-voltage (I-V) characteristics. The script consists of three primary sections: handling missing data, statistical analysis, and visualization of efficiency improvements and I-V characteristics. The first part of the script focuses on handling missing data and statistical analysis of efficiency improvements in perovskite solar cells. The script begins by importing necessary libraries (numpy and pandas) and defining sample efficiency data for different optimization strategies such as *Minimizing Parasitic Resistance, Enhancing Light Management, Optimizing Device

Geometry, and Advanced Material Composition*. The dataset includes a missing efficiency value, which is stored as NaN.

To address missing values, the script defines a function handle_missing_data(df), which fills missing efficiency percentages with the column mean using fillna(). This ensures that missing values do not disrupt statistical analysis. Another function, statistical_analysis(df), computes basic statistical summaries using describe(), providing insights into the mean, standard deviation, and distribution of efficiency improvements across optimization strategies.

The second part of the script generates a bar chart visualization for efficiency improvements. Using matplotlib.pyplot, it defines three optimization strategies (Strategy A, Strategy B, and Strategy C) with their respective efficiency percentages. The figure is saved as "efficiency_improvements_perovskite_solar_cells.png". The third part of the script focuses on current-voltage (I-V) characteristics of perovskite solar cells. It simulates an I-V curve using an exponential equation that models current as a function of voltage over a range of 0 V to 1 V. The function np.exp() is used to define an exponential decay characteristic, mimicking real-world solar cell behavior. The script then plots the I-V curve using matplotlib.pyplot, ensuring that gridlines, axis labels, and a legend are included for readability. The figure is saved as "iv_characteristics_perovskite_solar_cells.png".

Overall, this script automates data preprocessing, statistical analysis, and visualization, enabling a structured approach to analyzing efficiency improvements and electrical characteristics in perovskite solar cells. These analyses provide valuable insights for understanding the impact of optimization strategies and electrical behavior of perovskite-based devices.



Figure 27: The generated data analysis code for literature review and knowledge synthesis automation.

The visualized results in the Figure 28 provide key insights into efficiency improvements in perovskite solar cells and their current-voltage (I-V) characteristics. These visualizations help in understanding the impact of different optimization strategies and the electrical behavior of perovskitebased devices.

The bar charts (left side) display the efficiency improvements of perovskite solar cells under different optimization strategies. The top bar chart represents three generalized strategies (Strategy A, Strategy B, and Strategy C), while the bottom bar chart provides a more detailed view of specific optimization techniques, including Minimizing Parasitic Resistance, Enhancing Light Management, Optimizing Device Geometry, and Advanced Material Composition. The color-coded bars highlight variations in efficiency percentages, with Advanced Material Composition yielding the highest efficiency improvements (\sim 24.5%), followed by Optimize Device Geometry (\sim 22.3%). This indicates that material composition modifications play a crucial role in enhancing solar cell performance, while geometric and structural optimizations also contribute significantly.

The I-V characteristic curve (right side) represents the current-voltage relationship in perovskite solar cells, which is essential for evaluating their electrical performance. The plot follows a characteristic exponential trend, where current rapidly decreases as voltage increases. The shape of the curve suggests a typical diode-like behavior, where current is high at low voltages but levels off as voltage increases toward 1V. This pattern is consistent with the performance of solar photovoltaic cells, highlighting how perovskite materials exhibit strong charge transport properties at low voltage levels.

Together, these visualizations offer data-driven insights into perovskite solar cell performance. The bar charts emphasize the significance of various optimization strategies in improving efficiency, while the I-V curve provides a detailed look into the electrical characteristics of the solar cells. These findings can guide future material design, efficiency enhancement techniques, and performance evaluations in perovskite-based photovoltaic research.



Figure 28: The visualized results for automation of the perovskite-based photovoltaic research.