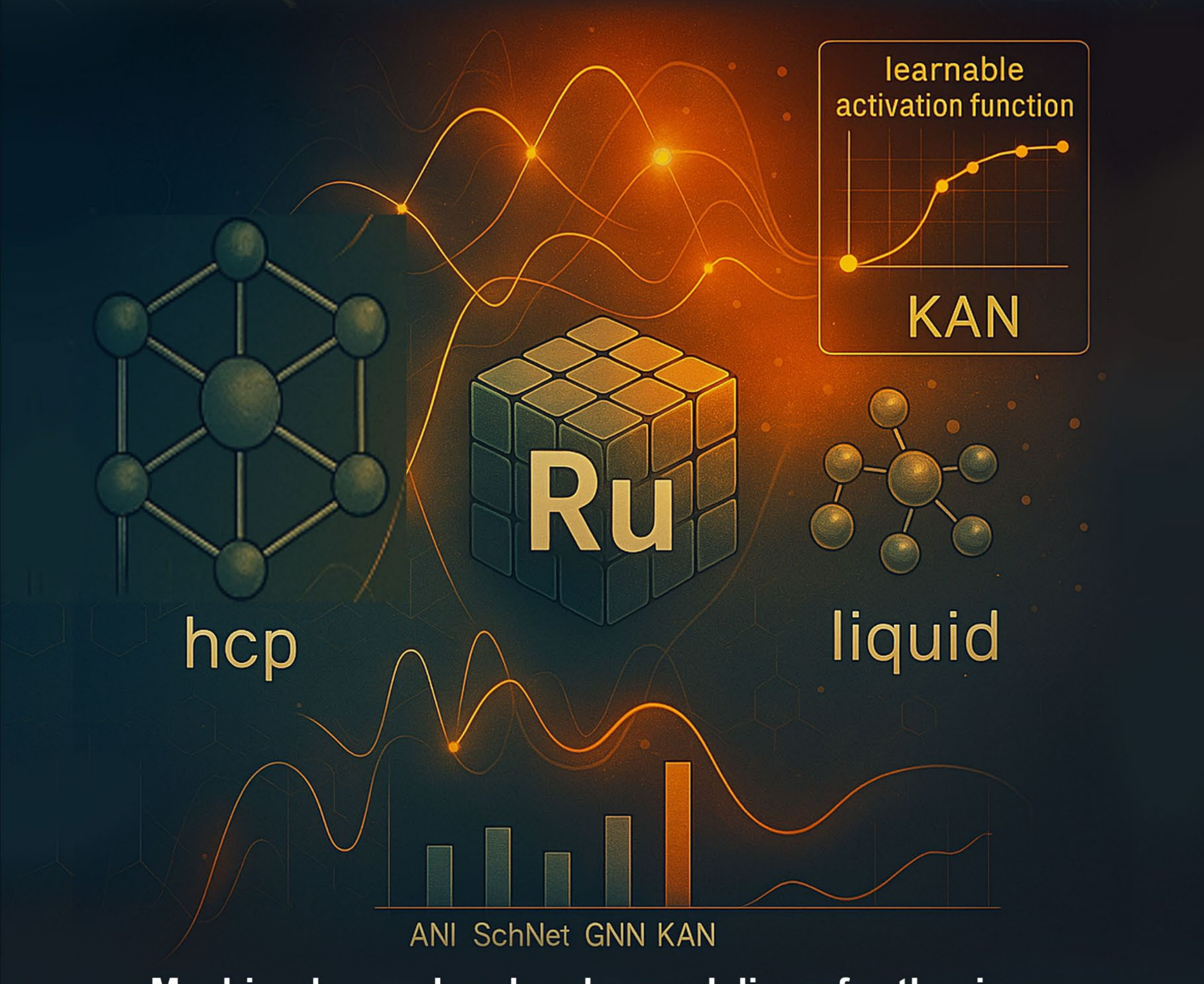
Volume 2 · Issue 1
March 2025

International Journal of AI for Materials and Design



Machine-learned molecular modeling of ruthenium: A Kolmogorov-Arnold Network approach

ISSN 3041-0746 (Print)
ISSN 3029-2573 (Online)

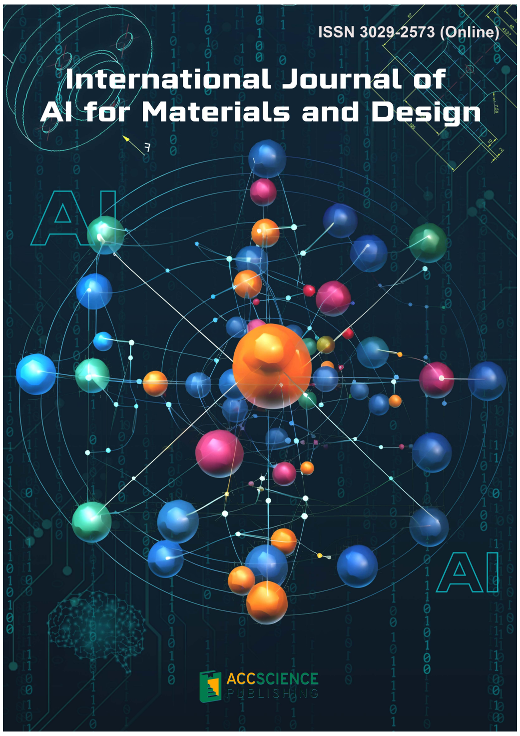
ACCSCIENCE PUBLISHING

ACCSCIENCE
PUBLISHING

International Journal of AI for Materials and Design

Print ISSN: 3041-0746
Online ISSN: 3029-2573

The International Journal of AI for Material and Design (IJAMD) is a scholarly publication dedicated to advancing the intersection of artificial intelligence (AI), materials science and design. This peer-reviewed journal provides a platform for researchers, academics, and industry professionals to disseminate cutting-edge research, innovative methodologies, and practical applications that leverage AI techniques to enhance the understanding, development, and optimization of aspects related to materials and design processes. IJAMD seeks to contribute to the advancement of technology, innovation, and sustainability in materials design, engineering disciplines, product manufacturing and process technology.



About the Publisher

AccScience Publishing is a publishing company based in Singapore. We publish a range of high-quality, open-access, peer-reviewed journals and books from a broad spectrum of disciplines.

Contact Us

Managing Editor
ijamd.office@accscience.sg

AccScience Publishing
8 Burn Road, #15-03 Trivex, Singapore 369977.

Volume 2 • Issue 1 • March 2025
ISSN 3041-0746 (print) ISSN 3029-2573 (online)

INTERNATIONAL JOURNAL OF AI FOR MATERIALS AND DESIGN

Editor-in-Chief

Wai Yee Yeong

Nanyang Technological University, Singapore

ACCSCIENCE PUBLISHING

Access Science Without Barriers

ACCSCIENCE
PUBLISHING

Full issue copyright © 2025 AccScience Publishing
All rights reserved. Without permission in writing from the publisher, this full issue publication in its entirety may not be reproduced or transmitted for commercial purposes in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system. Permissions may be sought from ijamd.office@accscience.sg.

Article copyright © Respective Author(s)
See articles for copyright year. All articles in this full issue publication are open-access. There are no restrictions in the distribution and reproduction of individual articles, provided the original work is properly cited. However, permission to reuse copyrighted materials of an article for commercial purposes is applicable if the article is licensed under Creative Commons Attribution-NonCommercial License. Check the specific license before reusing.

International Journal of AI for Materials and Design

ISSN: 3041-0746 (print)
ISSN: 3029-2573 (online)

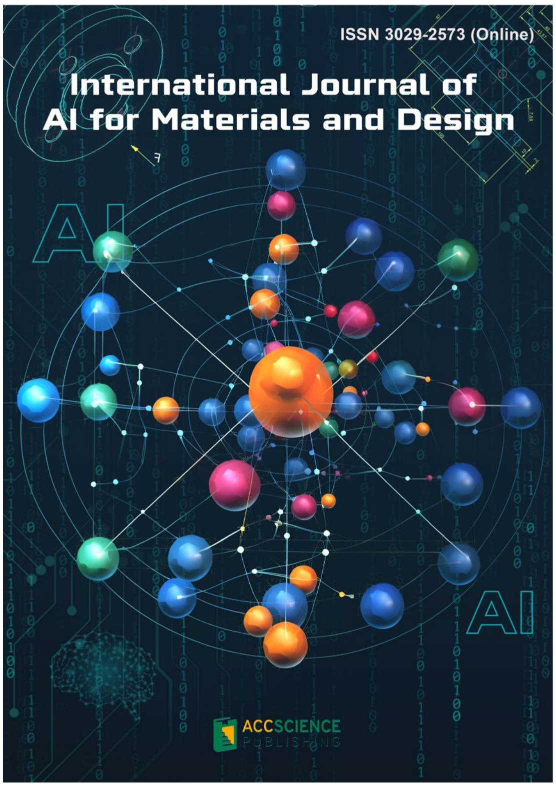
Editorial and Production Credits

Publisher: AccScience Publishing
Managing Editor: Shirley Lu
Production Editor: Sharmila Velapasamy
Article Layout and Typeset: Sinjore Technologies (India)

For all advertising queries, contact ijamd.office@accscience.sg.

Supplementary file

Supplementary files of articles can be obtained at https://accscience.com/journal/IJAMD/2/1.



Disclaimer

AccScience Publishing is not liable to the statements, perspectives, and opinions contained in the publications. The appearance of advertisements in the journal shall not be construed as a warranty, endorsement, or approval of the products or services advertised and/or the safety thereof. AccScience Publishing disclaims responsibility for any injury to persons or property resulting from any ideas or products referred to in the publications or advertisements. AccScience Publishing remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

International Journal of AI for Materials and Design

Editorial Board

Editor-in-Chief
Wai Yee Yeong, *Singapore*

Associate Editor
Shweta Agarwala, *Denmark*

*Editorial Board Members**
Antonio Gloria, *Italy*
Faris M. Al-Oqla, *Jordan*
Mehdi Amiri, *USA*
Ruzena Bajcsy, *USA*
Valentina E. Balas, *Romania*
Michail Beliatis, *Denmark*
Filippo Berto, *Italy*
Dermot Brabazon, *Ireland*
Ilaria Cacciotti, *Italy*
Yanlong Cao, *China*
Danni Chang, *China*
Chongdu Cho, *Korea*
Alfredo Cuzzocrea, *Italy*
Gianni D'Angelo, *Italy*
Frédéric Demoly, *France*
Shi Xue Dou, *Australia*
Zhimin Du, *China*
Mohammad Elahinia, *USA*
Gerd Grau, *Canada*
Qi Gu, *China*
Mohammad Heidari-Rarani, *Iran*
Im Doo Jung, *South Korea*
Seong Su Kim, *South Korea*
Hyunwoong Ko, *USA*
A. Senthil Kumar, *Singapore*
Panagiotis Kyratsis, *Greece*
Pascal LORENZ, *France*
Jay Lee, *USA*
Xiaopeng Li, *Australia*
Weifu Li, *China*
Kaili Lin, *China*
Xu Long, *China*
Jianxi Luo, *Singapore*
Dragan Marinkovic, *Germany*
Abbas Milani, *Canada*
Chilukuri K. Mohan, *USA*
Seung Ki Moon, *Singapore*
Nezih Mrad, *Canada*
Tuhin Mukherjee, *USA*
Roger Narayan, *USA*
Ping Feng Pai, *Taiwan*
Stefanos Papanikolaou, *Poland*
Radu-Emil Precup, *Romania*
Seunghwa Ryu, *South Korea*
Sonal Shreya, *Denmark*
Swee Leong Sing, *Singapore*
Anil Srivastava, *USA*
Tino Stankovic, *Switzerland*
Ephraim Suhir, *USA*
Gyorgy Szekely, *Saudi Arabia*
Jeanne Tan, *China*
Ehsan Toyserkani, *Canada*
Man Pun Wan, *Singapore*
Pan Wang, *Singapore*
Shenghao Wang, *China*
Hao Wang, *China*
Dazhong Wu, *USA*
W. Hong Yeo, *USA*
Jingjie Yeo, *USA*
Zhen Yuan, *China*
Y. Shrike Zhang, *USA*
Xing Zhang, *China*
Pai Zheng, *China*
Junxing Zheng, *China*
Hongxiang Zong, *China*

Assistant Editors
Nehru Devabharathi, *India*
Wei Long Ng, *Singapore*

Youth Editorial Board Members
Murali Mohan Cheepu, *Korea*
Faeez Masurkar, *UAE*
Haoyuan Shi, *USA*
Jinlong Su, *Singapore*
Yutai Su, China
Bijun Tang, *Singapore*
César M. A. Vasques, *Portugal*

Co-Guest Editors
Xiaorui Liu, *USA*
Liming Tan, *China*
Liang Wang, China
Kun Zhou, *Singapore*

Guest Editor
Dong Wang, *China*

*Editorial Board Members as of March 31, 2025

ACCSCIENCE
PUBLISHING

Volume 2 | Issue 1 | March 2025 | Page 1 – 72 |

International Journal of AI for Materials and Design

https://accscience.com/journal/IJAMD

CONTENTS

PERSPECTIVE ARTICLE

ORIGINAL RESEARCH ARTICLES

ACCSCIENCE PUBLISHING

International Journal of AI for Materials and Design

PERSPECTIVE ARTICLE

Utilizing artificial intelligence for National Transportation Safety Board unmanned aerial vehicle accident analysis and categorization

Eugene Pik[1]* **and Joao S. D. Garcia**[2]

[1]Mevocopter Aerospace, Vaughan, Ontario, Canada

[2]DB-School of Graduate Studies, Embry-Riddle Aeronautical University, Daytona Beach, Florida, United States of America

Abstract

The rapid increase in unmanned aerial vehicle (UAV) usage has introduced significant safety challenges, including issues such as system failure, loss of control, transmission failures, and collisions. Analyzing these incidents has been challenging due to the absence of a dedicated category field in the National Transportation Safety Board (NTSB) data. This research tackles this problem by utilizing artificial intelligence (AI) to automate the classification of UAV accident reports collected between 2006 and 2023. Using natural language processing techniques, we categorize NTSB reports to improve the analysis and interpretation of incident data. We also employ advanced data visualization tools to reveal geographic and temporal patterns, offering a detailed view of UAV accident trends. The results indicate that system and component failures unrelated to propulsion systems (system/component failure or malfunction [non-powerplant]) and abnormal contact upon landing (abnormal runway contact) are predicted as the primary categories (37%) of UAV accidents for the period. These insights suggest the potential value of AI-driven categorization and visualization techniques in enhancing UAV safety standards and supporting policy development. Initial results provide promising insight into the use of language models for text classification in aviation safety problems.

Keywords: UAV accident analysis; AI categorization; GPT-4 analysis; Data visualization in safety; NTSB accident data; Accident trend analysis

1. Introduction

The use of unmanned aerial vehicles (UAVs) has seen a dramatic increase in recent years. The commercial UAV fleet in the United States expanded from 42,000 in 2016 to 349,000 in 2023, representing a staggering 731% increase.[1] This surge in UAV usage brings with it some safety concerns, including loss of control, transmission failures, navigation system malfunctions, and collisions with aircraft, buildings, and power lines.[2] In addition, severe weather events, take-off and landing incidents, and rotor failures have also been mentioned as relevant to safety in UAV operations.

With the increase in operations, UAV-related accidents have also escalated, creating the need for improved categorization and understanding of these incidents to support

***Corresponding author:**
Eugene Pik
(eugene.pik@mevocopter.com)

Citation: Pik E, Garcia JSD. Utilizing artificial intelligence for National Transportation Safety Board unmanned aerial vehicle accident analysis and categorization. *Int J AI Mater Design*. 2025;2(1):1-7. doi: 10.36922/ijamd.8544

Received: January 15, 2025

Revised: February 11, 2025

Accepted: February 18, 2025

Published online: February 28, 2025

decision-making and risk management.[3] The textual nature of traditional accident reports can often hamper more automated structured categorization efforts, making it challenging to analyze and interpret the data efficiently. Advanced methods are gaining prominence in aviation research and practical industry application. They can contribute to the need to process large volumes of unstructured data and extract meaningful insights.[4] The National Transportation Safety Board (NTSB) investigates UAV-related accidents and makes detailed reports openly available to support safety analysis. Still, at times, some of these reports may have missing fields, inconsistent taxonomies and data formats, and can require natural language processing (NLP) to extract critical information and support statistical analysis.[5]

Artificial intelligence (AI), specifically NLP and machine learning approaches, can enhance the analysis of UAV accident reports and support the identification of safety improvements. By leveraging OpenAI's GPT-4, this study aims to evaluate the feasibility of automating the categorization of UAV accident reports and the identification of probable causes and patterns in the data. AI-driven categorization coupled with data visualization techniques can provide a deeper understanding of UAV accidents, enabling proactive measures to address safety issues and inform policy decisions. Given proper consideration of accuracy challenges that could be explored in further studies, the approach can also complement the resource-intensive manual categorization tasks or serve as a capability augmentation tool for such activities. This research highlights the potential of AI to revolutionize the way UAV accident data are processed and analyzed, ultimately contributing to improved safety standards and practices in the UAV industry.

2. Literature review

The exponential increase in UAV usage has led to a corresponding rise in safety challenges, necessitating comprehensive studies to mitigate associated risks. Key safety concerns include loss of control, navigation failures, and collisions with other aircraft or infrastructure. These issues often stem from operator error, mechanical failure, and adverse environmental conditions.[2,3] Regulatory and technical challenges further complicate the safe integration of UAVs into national airspace systems. Nguyen *et al.*[6] highlight difficulties in establishing consistent regulatory frameworks and the technical limitations of current UAV systems, which contribute to navigation and control issues. Synthesizing these studies, it becomes evident that addressing UAV safety requires a multifaceted approach involving enhanced risk assessments, robust regulatory measures, and technological advancements.

AI, particularly machine learning and deep learning, has shown significant promise in UAV data analysis. Nguyen *et al.*[6] demonstrated the effectiveness of multitask deep learning in analyzing UAV multisensory data, enhancing crop productivity and safety by accurately predicting equipment malfunctions and environmental conditions. Similarly, the application of large language models (LLM), particularly GPT-4, for categorizing UAV accident reports, as proposed in this study, leverages the technology to systematically classify incident causes. This structured approach offers a more nuanced understanding of UAV-related risks. Other researchers have also employed AI models to parse and interpret large datasets of accident reports, achieving higher accuracy in identifying patterns and probable causes than manual methods.[4,5,7-9] For instance, AI categorization techniques have been applied in manned aviation and automotive industries, demonstrating the versatility and robustness of AI-driven analysis.[10]

Effective data visualization can also be crucial for interpreting complex UAV data and communicating findings. Techniques such as cluster maps and interactive charts are employed to identify geographic and temporal patterns in UAV accidents. These visualization methods facilitate a better understanding of spatial distributions and trends, aiding in the development of targeted safety measures.[11,12] Research underscores the importance of visualization in making complex data accessible and actionable, particularly for policymakers and regulatory bodies aiming to implement data-driven safety interventions.[13,14]

The integration of AI and advanced data visualization techniques offers a powerful framework for enhancing UAV safety research. AI-driven categorization streamlines data analysis while visualization tools make the data more accessible and interpretable. Studies have shown that this integrated approach not only improves the accuracy of UAV data analysis but also supports the development of proactive safety policies. For instance, combining AI categorization with visual analytics has been used to identify critical safety issues and inform regulatory measures in aviation and other high-risk industries.[4] This synergy is essential for addressing the evolving challenges in UAV operations and ensuring robust safety standards. By leveraging the strengths of both AI and data visualization, researchers can provide deeper insights into accident data, enabling more effective and timely interventions to enhance UAV safety.

3. Methods

The UAV accident reports ($n = 34$) were sourced from the NTSB database, which provides detailed records of aviation incidents. These reports, covering accidents from April 2006 to August 2023, were obtained through the NTSB Aviation Investigation Search platform.[15] The dataset includes various fields such as event dates, probable causes, and geographic coordinates, which are crucial for analyzing and categorizing the accidents.

The collected data underwent several preprocessing steps to ensure its quality and usability. Dynamic encoding detection was employed to accurately read files with different encodings, preventing data corruption.[16] Text sanitization was performed using the Python and Unidecode library, which converts text to ASCII, making it uniform and easier to process.[17,18] Python's NumPy and Pandas together with the error handling mechanisms were implemented to manage missing values, inconsistent formats, and other anomalies in the dataset, ensuring the integrity and reliability of the processed data.[19,20]

3.1. AI categorization

OpenAI's GPT-4 application programming interface (API) was utilized to categorize the probable causes of UAV accidents.[21] The categorization process involved feeding the cleaned and sanitized report text data into the GPT-4 model, which then assigned an aviation occurrence category from the CAST-ICAO common taxonomy team (CICTT) to each accident report.[22] This approach leveraged the advanced NLP capabilities of GPT-4 to accurately interpret and classify the narrative descriptions of the accidents, thereby automating the categorization process and reducing manual effort.[23]

3.2. Visualization

Various data visualization tools and techniques were used to illustrate the findings and trends in the UAV accident data. Libraries such as Matplotlib, Seaborn, and Folium were employed to create charts, graphs, and maps.[12-14] These visualizations helped in identifying geographic distributions, temporal patterns, and key insights from the data. For instance, Matplotlib and Seaborn were used to generate detailed plots showing seasonal variations and accident trends over the years, while Folium was utilized to create interactive maps highlighting accident hotspots across the United States.

3.3. Python scripts for NTSB accident report analysis

The categorization script, *NTSB_analysis_with_gpt4_V3.0.py*, employs OpenAI's GPT-4 model to categorize UAV accident reports from the NTSB.[23] The process begins with data preprocessing, where the script reads raw accident reports and cleans and normalizes the text to ensure uniformity. This preprocessing step is crucial for removing inconsistencies and preparing the text for analysis. After preprocessing, the sanitized text is input into the GPT-4 model, which categorizes each report into predefined aviation occurrence categories based on the narrative descriptions. The categorized data is then saved for subsequent analysis.

The visualization script, *NTSB_reports_visualisation_and_map_V1.1.py*, focuses on visualizing the categorized UAV accident data.[23] The script begins by loading the categorized accident data. It then uses various data visualization tools, such as Matplotlib, Seaborn, and Folium, to generate visual representations of the data. The visualizations include line graphs and bar charts to illustrate temporal trends and accident frequencies over time. In addition, interactive maps created with Folium highlight geographic distributions and accident hotspots. These visualizations enable the identification of key insights, trends, and patterns in the accident data, facilitating a deeper understanding of UAV accident occurrences.

Together, these scripts streamline the processes of categorizing and visualizing NTSB accident reports, enhancing the efficiency and accuracy of data analysis.

4. Results

Following, OpenAI API was used to assign accident categories from the CICTT taxonomy to the NTSB UAV accident reports.[22] Table 1 lists names and codes of the categories assigned by the AI using the ICAO list.

The analysis revealed that the primary classification of UAV accidents is system and component failure, specifically categorized as "System and Component Failure or Malfunction (SCF-NP)." This category encompasses issues such as loss of control, transmission failures,

Table 1. UAV accident codes and category names

Code	Category name
FUEL	Fuel related
ICE	Icing
RAMP	Ground handling
WSTRW	Wind shear or thunderstorm
LOC-G	Loss of control – ground
SEC	Security related
SCF-PP	System/component failure or malfunction (powerplant)
MAC	Airprox/TCAS alert/loss of separation/near midair collisions/midair collisions
NAV	Navigation errors
LOC-I	Loss of control – inflight
ARC	Abnormal runway contact
SCF-NP	System/component failure or malfunction (non-powerplant)

and rotor malfunctions. Figure 1 shows the number of accidents per category.

Geographic distribution analysis showed that UAV accidents are widespread across the United States, with certain areas exhibiting higher concentrations of incidents. Hotspots were identified in regions with dense urban development and higher UAV activity. Figure 2 shows the map of the incidents.

This spatial analysis is crucial for targeted safety interventions and regulatory measures in specific areas. Visualizations, including heat maps and cluster maps created using Folium, effectively illustrate these geographic patterns, providing clear insights into regional accident trends.

Figure 3 shows notable yearly variations, with a significant spike in accidents observed in 2019. However, a downward trend in accidents per UAV after 2019 suggests that recent safety measures and technological advancements are beginning to have a positive impact.

The visualizations and tables generated from the data analysis provided a comprehensive view of the accident data. For instance, a table summarizing the frequency of different accident causes offered a quick reference to the most common issues, while a series of bar charts illustrated the monthly and yearly accident trends. Interactive maps highlighted accident hotspots, enabling a more intuitive understanding of the geographic distribution. These visual tools not only supported the findings but also enhanced the overall presentation of the data, making it accessible and interpretable for a broader audience.

Some limitations associated with the work are worth mentioning. A significant one is the limitation of the number of reports that supported the analysis. Expanding this initial exploratory approach to a larger set of reports would allow the testing of model predictive performance with added confidence. Developing an SME validated dataset with assigned CICTT categories could create opportunities for estimating predictive precision of the proposed models.

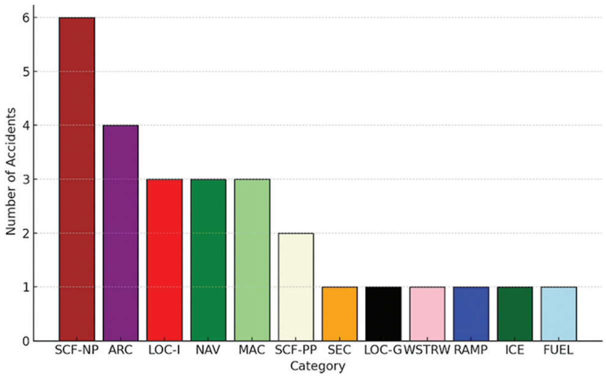


Figure 1. Number of accidents in each category

5. Discussion

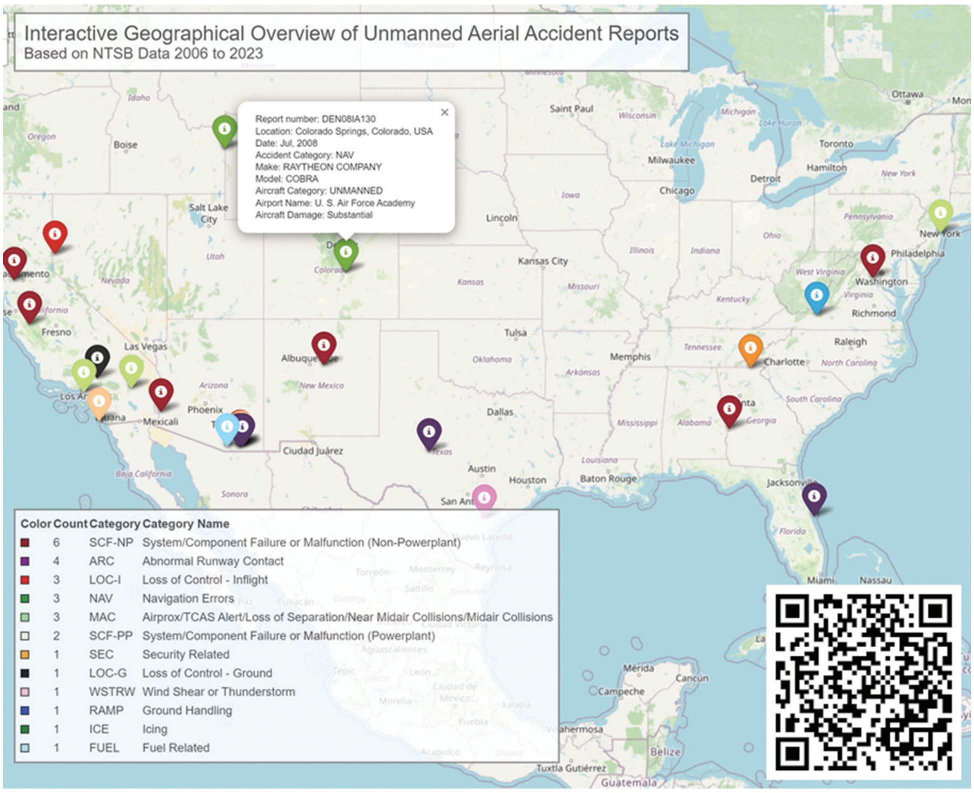
The results of this study underscore the critical importance of addressing system and component failures to enhance UAV safety. The prevalence of issues such as loss of control and navigation system failures suggests that technological improvements and stringent maintenance protocols are essential. The geographic distribution of accidents further highlights the need for localized safety interventions, particularly in urban areas where UAV operations are more frequent and complex. Monitoring the annual evolution can provide valuable insights for anticipating specific risk trends, allowing for more effective preemptive safety measures.

When compared with previous studies, our findings align with the broader consensus that UAV safety is predominantly compromised by technological failures. However, our use of AI-driven categorization offers a more nuanced understanding of these issues. Ferrigan[2] identified similar hazards but his findings lacked the granularity provided by AI categorization. In addition, our identification of specific geographic and temporal patterns offers new dimensions for understanding UAV safety, which were less explored in prior research. This highlights the added value of using advanced AI and data visualization techniques.

The integration of AI and data visualization has significant implications for improving UAV safety policies. AI, particularly NLP through GPT-4, enables efficient and accurate categorization of accident reports, which is essential for large-scale data analysis. This automation reduces the manual workload and increases the consistency of data interpretation. Data visualization tools like Matplotlib, Seaborn, and Folium transform raw data into insightful visual representations, making it easier for policymakers to identify trends and patterns. Together, these technologies provide a powerful framework for developing data-driven safety strategies, enhancing regulatory measures, and ultimately reducing the incidence of UAV accidents.

6. Future work

Future research should explore the potential of AI techniques for more sophisticated analysis of UAV accident data. These techniques can uncover complex patterns and correlations simpler models might miss. Further development of NLP methods is crucial for extracting deeper insights from accident reports. Advanced NLP models can analyze



Report number: DEN08IA130
Location: Colorado Springs, Colorado, USA
Date: Jul, 2008
Accident Category: NAV
Make: RAYTHEON COMPANY
Model: COBRA
Aircraft Category: UNMANNED
Airport Name: U. S. Air Force Academy
Aircraft Damage: Substantial

Color	Count	Category	Category Name
■	6	SCF-NP	System/Component Failure or Malfunction (Non-Powerplant)
■	4	ARC	Abnormal Runway Contact
■	3	LOC-I	Loss of Control - Inflight
■	3	NAV	Navigation Errors
■	3	MAC	Airprox/TCAS Alert/Loss of Separation/Near Midair Collisions/Midair Collisions
□	2	SCF-PP	System/Component Failure or Malfunction (Powerplant)
■	1	SEC	Security Related
■	1	LOC-G	Loss of Control - Ground
■	1	WSTRW	Wind Shear or Thunderstorm
■	1	RAMP	Ground Handling
■	1	ICE	Icing
■	1	FUEL	Fuel Related

Figure 2. Interactive map of the UAV accidents[23]

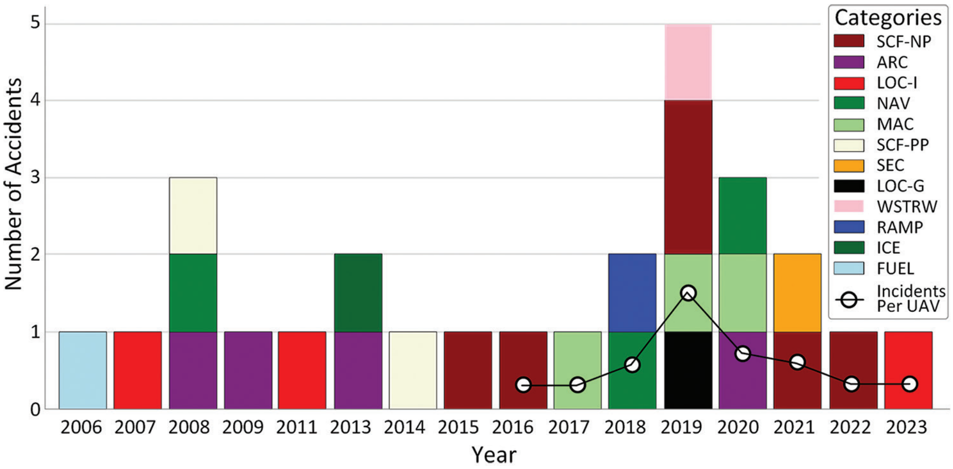


Figure 3. Accidents per year by category

narrative sections of reports to identify underlying causes and contributing factors more accurately. Machine learning algorithms should be developed and refined to classify accident scenarios and predict potential risks. These algorithms can enhance proactive safety measures by providing early warnings based on historical data and emerging trends. Interdisciplinary research combining expertise from aviation, AI, and data science can lead to comprehensive safety frameworks. Collaboration with regulatory bodies and industry stakeholders will be vital in implementing these advanced technologies effectively. Future work will involve a larger number of NTSB reports to enable robust statistical analysis. Future research can enhance reliability by ensuring consistent results across different initial conditions and demonstrating model convergence as more data is added, confirming stability and robustness. Additional research could employ a similar process to explore socioeconomic factors associated with such occurrences.

7. Conclusion

This research demonstrates the significant potential of AI in categorizing and analyzing UAV accident reports. By leveraging OpenAI's GPT-4 and various data visualization tools, we have identified patterns and trends in UAV accidents, highlighting the primary causes and their geographic and temporal distributions. Our findings underscore the importance of addressing system and component failures to improve UAV safety. The use of AI and data visualization not only streamlines the analysis process but also provides valuable insights that can inform safety policies and regulatory measures. The integration of these technologies is crucial for advancing UAV safety research and enhancing the overall safety standards in the UAV industry. Moving forward, continuous improvement of AI models and data collection methods will be essential to keep pace with the rapidly evolving UAV landscape, ensuring that safety measures remain robust and effective.

Acknowledgments

None.

Funding

None.

Conflict of interest

The authors declare that they have no competing interests.

Author contributions

Conceptualization: Eugene Pik
Formal analysis: Joao S. D. Garcia
Investigation: Eugene Pik
Methodology: Eugene Pik
Writing – original draft: Eugene Pik
Writing – review & editing: Joao S. D. Garcia

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

1. The original data presented in this study are openly accessible at https://www.ntsb.gov/Pages/AviationQueryV2.aspx.
2. The data analysis scripts for this paper are available at https://doi.org/10.5281/zenodo.10576209.

Further disclosure

Part of findings has been presented in a conference "ERAU Discovery Days" (https://commons.erau.edu/discovery-day/db-discovery-day-2024/poster-session-2/54/).

References

1. Pik E. *Commercial and Recreational Active UAV Fleet in the U.S. 2016-2022*; 2024.

   doi: 10.5281/zenodo.10573914

2. Ferrigan J. *Safety Risk Assessment for UAV Operation*; 2022. Available from: https://www.regulations.gov/document/faa-2022-0426-0004 [Last accessed on 2024 Jan 15].

3. Zhang X, Srinivasan P, Mahadevan S. Sequential deep learning from NTSB reports for aviation safety prognosis. *Saf Sci*. 2021;142:105390.

   doi: 10.1016/j.ssci.2021.105390

4. Yang C, Huang C. Natural language processing (NLP) in aviation safety: Systematic review of research and outlook into the future. *Aerospace*. 2023;10(7):600.

   doi: 10.3390/aerospace10070600

5. Kasprzyk PJ, Konert A. Reporting and investigation of unmanned aircraft systems (UAS) accidents and serious incidents. Regulatory perspective. *J Intell Robot Syst*. 2021;103(1):3.

   doi: 10.1007/s10846-021-01447-6

6. Nguyen C, Sagan V, Bhadra S, Moose S. UAV multisensory data fusion and multi-task deep learning for high-throughput maize phenotyping. *Sensors* (*Basel*). 2023;23(4):1827.

   doi: 10.3390/s23041827

7. Nanyonga A, Wild G. Impact of Dataset Size and Data Source on Aviation Safety Incident Prediction Models with Natural Language Processing. In: *2023 Global Conference on Information Technologies and Communications (GCITC)*; 2023. p. 1-7.

   doi: 10.1109/GCITC60406.2023.10426284

8. Lázaro FL, Madeira T, Melicio R, Valério D, Santos LFFM. Identifying human factors in aviation accidents with natural language processing and machine learning models. *Aerospace*. 2025;12(2):106.

   doi: 10.3390/aerospace12020106

9. New MD, Wallace RJ. Classifying aviation safety reports: Using supervised natural language processing (NLP) in an Applied Context. *Safety*. 2025;11(1):7.

   doi: 10.3390/safety11010007

10. Sarkar NI, Gul S. Artificial Intelligence-based autonomous

UAV networks: A survey. *Drones*. 2023;7(5):322.

doi: 10.3390/drones7050322

11. Kovanič Ľ, Topitzer B, Peťovský P, Blišťan P, Gergeľová MB, Blišťanová M. Review of photogrammetric and lidar applications of UAV. *Appl Sci*. 2023;13(11):6732.

    doi: 10.3390/app13116732

12. Filipe, Anema F, Story R, *et al*. *Python-Visualization/Folium: V0.15.1*; 2023.

    doi: 10.5281/zenodo.10255171

13. Caswell TA, Droettboom M, Lee A, *et al*. *Matplotlib REL: V3.5.1*; 2021.

    doi: 10.5281/zenodo.5773480

14. Waskom M. *Seaborn: Statistical Data Visualization*; 2021.

    doi: 10.5281/zenodo.4645478

15. National Transportation Safety Board. *NTSB Aviation Investigation Search*; 2023. Available from: https://www.ntsb.gov/pages/aviationqueryv2.aspx [Last accessed on 2024 Jan 11].

16. Pilgrim M. *Chardet: Universal Encoding Detector for Python 3*; 2023. Available from: https://pypi.org/project/chardet [Last accessed on 2024 Jan 15].

17. Solc T. Unidecode: ASCII Transliterations of Unicode Text; 2024. Available from: https://pypi.org/project/unidecode [Last accessed on 2024 Jan 15].

18. Van Rossum G, Drake FL. *The Python Language Reference*; 2012. Available from: https://docs.python.org/3/reference/index.html [Last accessed on 2024 Jan 15].

19. Harris CR, Millman KJ, Van der Walt SJ, *et al*. Array programming with NumPy. *Nature*. 2020;585(7825):357-362.

    doi: 10.1038/s41586-020-2649-2

20. The Pandas Development Team. *Pandas-dev/Pandas: Pandas*; 2023.

    doi: 10.5281/zenodo.3509134

21. OpenAI. *OpenAI Platform-Documentation*; 2024. Available from: https://platform.openai.com/docs/api-reference/introduction [Last accessed on 2024 Jan 11].

22. ICAO. *Aviation Occurrence Categories-Definitions and Usage Notes*; 2013. Available from: https://www.ntsb.gov/safety/data/documents/datafiles/occurrencecategorydefinitions.pdf [Last accessed on 2024 Jan 11].

23. Pik E. *GPT-4 Assisted Categorization and Visualization of NTSB UAV Accident Reports-Python Scripts*; 2024.

    doi: 10.5281/zenodo.10576209

ACCSCIENCE PUBLISHING

International Journal of AI for Materials and Design

ORIGINAL RESEARCH ARTICLE

Predicting effective thermal conductivity of sintered nano-Ag with artificial neural networks

Libo Zhao[1†], Jiahui Wei[1†], Yanwei Dai[1*], Daowei Wu[2], Yuting Zhang[2], Kui Li[3], and Fei Qin[1]

[1]Department of Mechanics, Institute of Electronics Packaging Technology and Reliability, Beijing University of Technology, Beijing, China

[2]Advanced Packaging Division, Xi'an Institute of Microelectronics Technology, Xi'an, Shaanxi, China

[3]R&d Innovation Center, Xi'an Institute of Microelectronics Technology, Xi'an, Shaanxi, China

(This article belongs to the *Special Issue: AI for Multiscale Analysis and Defect Identification in Packaging Structures and Semiconductor Chips*)

Abstract

Due to the demand for high reliability and thermal conductivity of high-power modules operating at high temperatures, sintered nano-silver (Ag) has garnered significant attention as an excellent interconnect and heat transfer layer, particularly for its thermal conductivity and other reliability research. Since the mechanical behavior and heat conduction capacity of sintered Ag is generally regulated by changes in temperature, its microstructure will change accordingly, affecting its performance. In this study, a machine learning model was used to evaluate and predict the thermal conductivity of sintered Ag, providing an effective method to analyze the influence of microstructural characteristics on its heat transfer properties. Image processing and model simulation of scanning electron microscopy images of sintered nano-Ag nanostructures were performed using MATLAB and Ansys software. A batch calculation of the thermal conductivity of 2D images of sintered nano-Ag nanostructures was performed to obtain sufficient data sets. Based on the artificial neural network model of Bayesian optimization, the equivalent thermal conductivity of different sintered nano-Ag microstructures was predicted with high accuracy using the microstructure image and characteristic parameters of sintered nano-Ag. The proposed method enables rapid, effective, and accurate evaluation and prediction of the thermal conductivity of sintered nano-Ag, contributing significantly to the reliability of power modules.

Keywords: Artificial neural networks; Sintered nano-Ag; Effective thermal conductivity; Finite element modeling

1. Introduction

Silicon carbide (SiC)-based power devices face limitations in achieving more effective energy conversion. To address the high reliability and thermal conduction demands of power modules operating at high temperatures, sintered nano-silver (Ag) has been developed and utilized frequently as the die-attaching material for SiC devices, due to its excellent performance in heat transfer and chip joining.

[†] *These authors contributed equally to this paper.*

***Corresponding author:**
Yanwei Dai
(ywdai@bjut.edu.cn)

Citation: Zhao L, Wei J, Dai Y, *et al.* Predicting effective thermal conductivity of sintered nano-Ag with artificial neural networks. *Int J AI Mater Design*. 2025;2(1):8-20. doi: 10.36922/ijamd.5744

Received: November 1, 2024

1st revised: December 10, 2024

2nd revised: December 19, 2024

3rd revised: January 13, 2025

Accepted: January 15, 2025

However, the mechanical behavior and heat conduction ability of sintered nano-Ag generally vary under different power cycling conditions, possibly due to damage accumulation and crack formation within its structure.[1] These variations in thermal performance could reduce the reliability of the modules and even lead to their failure. Thus, accurate evaluation and prediction of heat conductivity of sintered nano-Ag are essential.

To calibrate the effective thermal conductivity of sintered nano-Ag, different methods have been presented to understand the variations of effective thermal conductivity. Ordonez-Miranda *et al.*[2] measured and calculated the thermal conductivity of sintered nano-Ag for a sample with 22% porosity, demonstrating that pancake-shaped pores have a more prominent effect on thermal conductivity compared to sphere pores through numerical comparisons. Signor *et al.*[3] computed the thermal conductivity of sintered nano-Ag with finite element analysis using actual 3D microstructures. Recently, Sghuri *et al.*[4] tested the thermal conductivity of sintered nano-Ag under aging conditions to obtain the thermal conductivity of sintered nano-Ag more directly. Meanwhile, Hu *et al.*[5] explored the process-microstructure-thermal relation using focused ion beam scanning electron microscopy. Although these studies have reported various behavior of sintered nano-Ag, predicting its thermal conductivity remains a major challenge due to the complexity of the process and flaw-dependent thermal behavior.

Various analytical and numerical methods have been developed to predict the effective thermal conductivity of sintered nano-Ag. Zhao *et al.*[6] indicated that the existing models are not suitable for predicting the thermal conductivity of sintered nano-Ag due to its high porosity and complex microstructure. Qin *et al.*[7] have presented a semi-analytical formulation to predict the effective thermal conductivity of sintered nano-Ag by considering the modification in the microstructure. To investigate changes in the effective thermal conductivity of sintered nano-Ag due to crack formation, they also proposed a semi-analytical formulation to predict the effect of cracks on the heat conductivity of sintered nano-Ag.[8] Lately, the effect of mud cracking on the heat conductivity of sintered nano-Ag was also studied,[9] revealing variations in heat transfer behaviors of sintered nano-Ag in the entire SiC module. Kim *et al.*[10] also studied the effect of pore shape and porosity on the effective thermal conductivity of sintered nano-Ag.

With the development of artificial intelligence, the prediction of thermal conductivity for different materials with machine learning methods has garnered significant attention due to advantages, such as high accuracy, efficiency, and potential for physics-based interpretation.[11-16] Predicting mechanical and material properties using machine learning-assisted methods has also garnered significant attention in electronic packaging. Machine learning has been applied to different aspects of electronic packaging, such as materials extraction,[17-19] solder fatigue lifetime prediction,[20-22] defect detection,[23] and mechanical response prediction.[24] In the field of electronic packaging, Long *et al.*[25] proposed a convolution neural network (CNN)-assisted nanoindentation method to rapidly determine the mechanical properties of thin-film elastoplastic materials and predict the constitutive parameters with high accuracy. Recently, Du and coworkers[26,27] adopted CNN for support vector regression models to predict the thermal conductivity of sintered nano-Ag, demonstrating the potential applications of machine learning in predicting the thermal conductivity of sintered nano-Ag. However, machine learning requires large datasets. In addition, neural networks have demonstrated great potential in predicting the mechanical and thermal behavior of porous materials. For example, Wei *et al.*[28] used three different machine learning approaches to quickly and accurately predict the equivalent thermal conductivity of composite materials. Similar studies have been conducted in recent years.[18,29,30]

In this study, we focus on the research status and existing problems of physical and mechanical parameter evaluation methods for sintered nano-Ag nanomaterials. An image dataset, based on the Gaussian random reconstruction of sintered nano-Ag nanostructures, was proposed, and the equivalent thermal conductivity of sintered nano-Ag nanostructures was efficiently predicted based on a machine learning model. Image processing and model simulation were performed using MATLAB and Ansys software from scanning electron microscopy (SEM) images of sintered nano-Ag nanostructures; batch calculation of the thermal conductivity of 2D images of sintered nano-Ag nanostructures was conducted thereafter. According to the SEM image characteristics of actual sintered nano-Ag nanostructures, the images of the nanostructures at different sintering temperatures were generated by the Gaussian random model. Based on the artificial neural network (ANN) model of Bayesian optimization, the equivalent thermal conductivity corresponding to different sintered nano-Ag microstructures was accurately predicted using the microstructure image and characteristic parameters of sintered nano-Ag, with minimal loss and a high determination coefficient (0.96).

2. Methods

2.1. Effective thermal conductivity computation scheme

2.1.1. Finite element model of sintered nano-Ag

Image-to-parameter automated programming can be used to improve the efficiency of analytical calculations.

In this study, image processing and modeling simulation of microstructure images were performed using MATLAB and Ansys, respectively. The results from the batch calculation of the thermal conductivity of sintered nano-Ag microstructures enhanced the understanding of the physical relationship between sintered nano-Ag microstructure and heat transfer properties.

Figure 1 displays the finite element simulation flow chart of sintered nano-Ag microstructures, utilizing image-to-parameter automated programming. Using MATLAB, the SEM image of sintered nano-Ag (Figure 2A) was analyzed and converted into a grayscale image with two-phase regions of black and white (Figure 2B). The black region corresponds to the pore, while the white region denotes the nano-Ag nanoparticles. The grayscale images were then divided into *n* parts equally in the *x* and *y* directions (Figure 2C) to obtain *n* × *n* black and white pixel images. These images are stored as numerical matrices with values of 0 or 1.

After the pixel matrix was imported into Ansys, the voxel blocks were selected successively according to the coordinate position by the loop statement. The array parameters at the corresponding positions were analyzed to determine the material type of the voxel block, thereby assigning the appropriate material properties to the current voxel block. Since sintered nano-Ag is an isotropic material, distribution in the *x*-, *y*-, and *z*-directions are consistent. Therefore, 2D models can be used to simulate the heat transfer behavior of sintered nano-Ag, with microstructural characteristics. A plane model of nano-Ag and air was established, where the thermal conductivity of dense nano-Ag and air is 429 and 0.03 W/mK, respectively. As the difference in thermal conductivity between dense nano-Ag and air is approximately five orders of magnitude, modeling the pore regions has minimal impact on the heat conduction simulation results. However, since isolated islands (Figure 3A) often exist in practice, omitting the air unit would require additional boundary conditions to be applied separately, increasing the workload and complicating the calculations. Hence, the pores were filled with air in this study (Figure 3B). To ensure accuracy, each pixel block was further divided into four units during the meshing process. Boundary conditions (250 and 50°C) were applied to the upper and bottom boundaries, respectively, and adiabatic boundary conditions were applied to the other boundaries. Figure 4 displays the temperature distribution density contour plot of the model (Figure 4A) and the heat flux density of each element (Figure 4B).

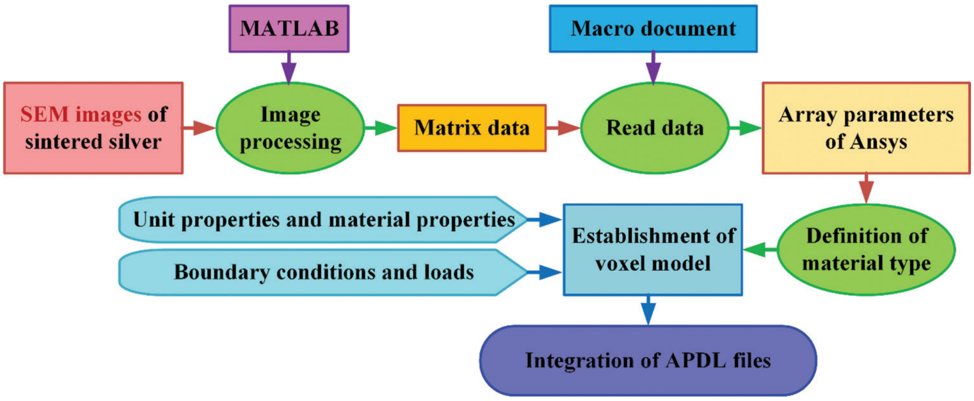


Figure 1. Finite element simulation flow chart of sintered nano-Ag microstructures
Abbreviations: SEM: Scanning electron microscopy; APDL: ANSYS parametric design language.

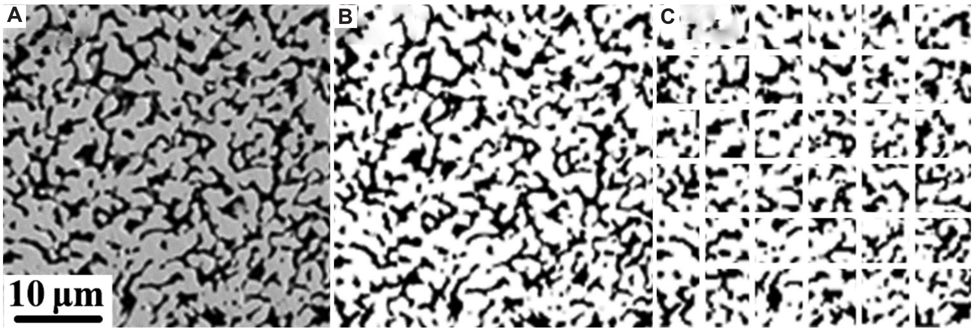


Figure 2. Gray transformation process of scanning electron microscopy (SEM) images of sintered nano-Ag: (A) SEM image of sintered nano-Ag; (B) grayscale images; and (C) image segmentation.

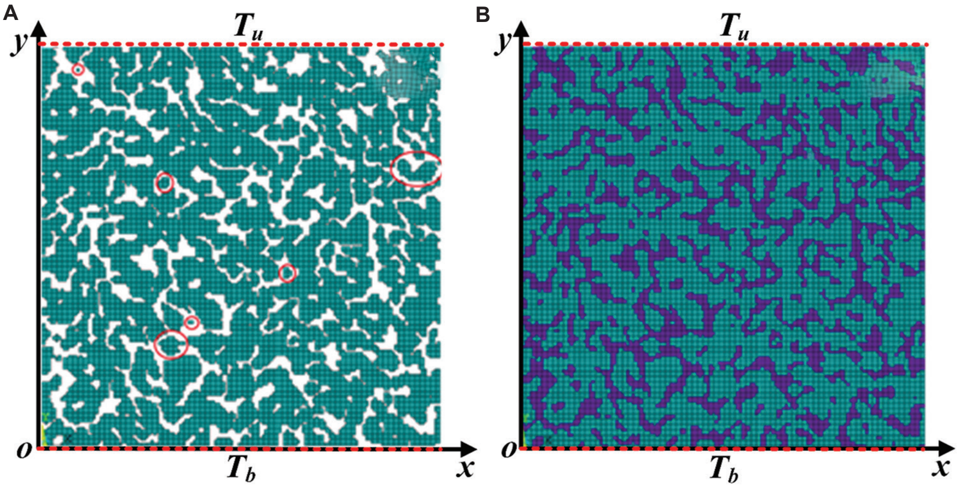


Figure 3. Comparison of microstructure finite element model before and after filling. (A) Schematic diagram of isolated islands in the model before air-filling. (B) Schematic diagram of the model after air-filling. Green boxes indicate geometric model of sintered nano Ag. Blue boxes indicate pore regions of sintered nano Ag. Red circles indicate isolated islands sintered nano Ag. Red dashed lines indicate the upper and lower boundary.
Abbreviations: T_u: Upper-temperature boundary condition; T_b: Lower temperature boundary condition.

2.1.2. Effective thermal conductivity computation

For homogeneous materials, according to Fourier's law of heat transfer,[31] the rate of heat flow in the y-direction (1D form of Fourier's law of heat transfer) can be expressed as:

$$q_y = -k\frac{\mathrm{d}T}{\mathrm{d}y} \tag{I}$$

where q_y is the heat flow in the y-direction, k is thermal conductivity, and $\mathrm{d}T/\mathrm{d}y$ is the temperature gradient in the y-direction.

A simple modification of Equation I yields the formula for thermal conductivity:

$$k = -\frac{q_y\left(y_\mathrm{u} - y_\mathrm{b}\right)}{T_\mathrm{u} - T_\mathrm{b}} \tag{II}$$

where y_u is the coordinate of the upper boundary in the y-direction, y_b is the coordinate of the bottom boundary in the y-direction, T_u is the temperature at the upper boundary, and T_b is the temperature of the bottom boundary.

In finite element analysis, the model is divided into units; thus, Equation II can be expressed as follows:

$$k_\mathrm{eq} = -\frac{q_y\left(y_u - y_\mathrm{b}\right)S}{\left(T_u - T_\mathrm{b}\right)S} = -\frac{q_y\left(y_\mathrm{u} - y_\mathrm{b}\right)\sum_{i=1}^{n} S_i}{\left(T_\mathrm{u} - T_\mathrm{b}\right)S}$$
$$= -\frac{\left(y_\mathrm{u} - y_\mathrm{b}\right)\sum_{i=1}^{n} q_{yi} S_i}{\left(T_\mathrm{u} - T_\mathrm{b}\right)S} \tag{III}$$

where S is the total area of the finite element model, S_i is the area of unit i, and q_{yi} is the heat flux value of unit i in the y-direction.

Equation III is a discrete formula for calculating the equivalent thermal conductivity k_eq, which applies to porous media models. After determining the heat flux and area of each unit in the dense nano-Ag region in the y-direction, the equivalent thermal conductivity of the entire model can be obtained using Equation III.

In calculating the equivalent thermal conductivity of sintered nano-Ag nanoparticles using the SEM images, the number of points n must be sufficient to ensure convergence of the model results. In two SEM images obtained from the same sintering process, 18 windows of 20 μm² were extracted, and the number of pixels was 50 × 50, 100 × 100, 150 × 150, and 200 × 200. Through simulation calculation, the thermal conductivity of each window under different extraction conditions was obtained (Figure 5A and B), with the mean and range of thermal conductivity presented for each extraction condition. Notably, the difference in calculated thermal conductivity between a pixel size of 150 × 150 and 200 × 200 is 3.3%.

2.2. Machine learning method

2.2.1. Evaluation index of the model

To evaluate the prediction accuracy of different models, five statistical indicators were used to characterize and compare the prediction accuracy of different data-driven models.[32-34]

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - y_i^*\right)^2 \tag{IV}$$

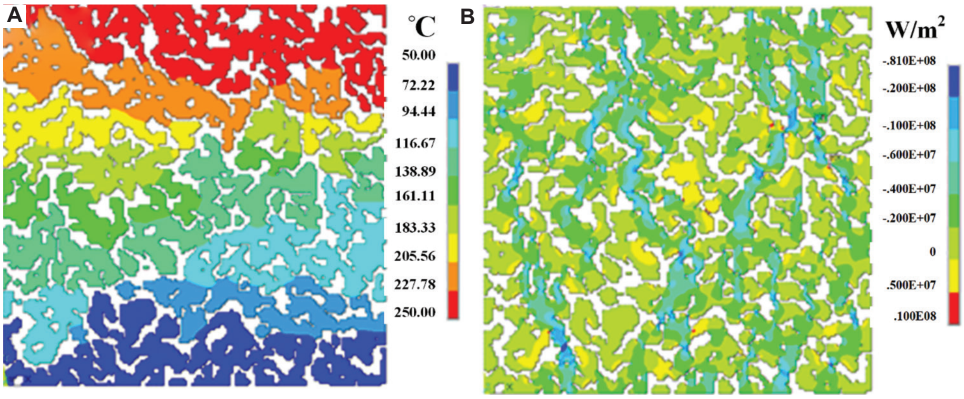


Figure 4. Finite element calculation results of the sintered nano-Ag microstructure. (A) Temperature cloud image of the model; and (B) heat flux density of each element in the *y*-direction.

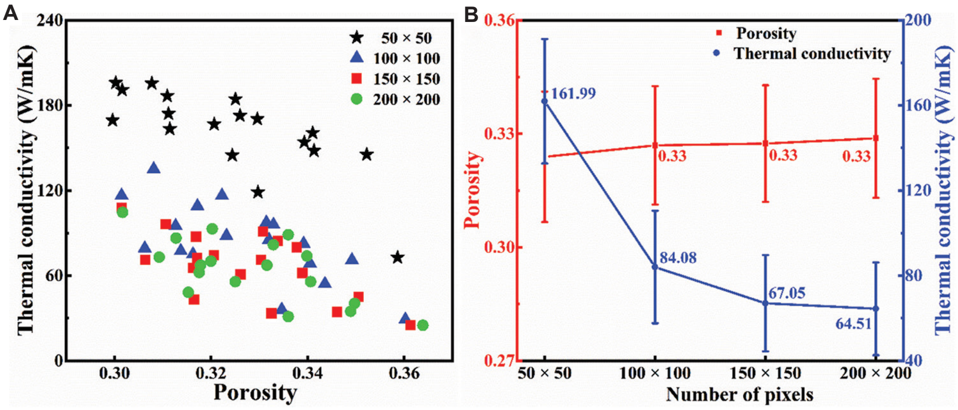


Figure 5. Finite element calculation results of models with different precision. (A) Thermal conductivity distribution of models corresponding to different porosity; and (B) the mean value and range of thermal conductivity under each extraction condition.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - y_i^*)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \tag{V}$$

Where y_i and y_i^* are the *i*-th actual output value and predicted output value, respectively; $\bar{y}$ and $\overline{y^*}$ are the average actual output value and the average predicted output value, respectively; and *n* is the number of samples. The mean squared error (MSE), as displayed in Equation IV, is an indicator used to measure the average squared difference between model predictions and actual observations. MSE is commonly used as the loss function to estimate the inconsistency between predicted values and actual values.[32] The coefficient of determination R^2, as displayed in Equation V, represents the determination coefficient, indicating the degree of fit between the regression model and actual data.[33,34] When R^2 is close to 1, it indicates that the model fits the actual data well.

2.2.2. Data preparation and network selection

A large amount of high-quality microstructure data of sintered nano-Ag is required to simulate the microstructural effect on the thermal conductivity of sintered nano-Ag. After an open operation and convolution kernel size adjustment, random images of crystal nucleus distribution generated by the Gaussian random model were used to simulate the growth process of a crystal nucleus (Figure 6). Finally, 186 images of sintered nano-Ag microstructure, corresponding to three different pixel sizes and sintering times, were obtained. Based on the images, a plane model of the microstructure was built in Ansys to calculate the thermal conductivity of the model in the *x*- and *y*-directions, while the boundary conditions (i.e., sintering temperatures) were applied to the upper and bottom boundaries. From a microstructural perspective, the input formats of neural network calculations can be divided into two categories: (i) One method involves extracting the index parameters of the microstructure as numerical inputs, and (ii) the other method directly uses microstructure images of sintered nano-Ag as input. The former provides better computational speed but offers limited information about the model; the latter enables more comprehensive and accurate feature extraction but

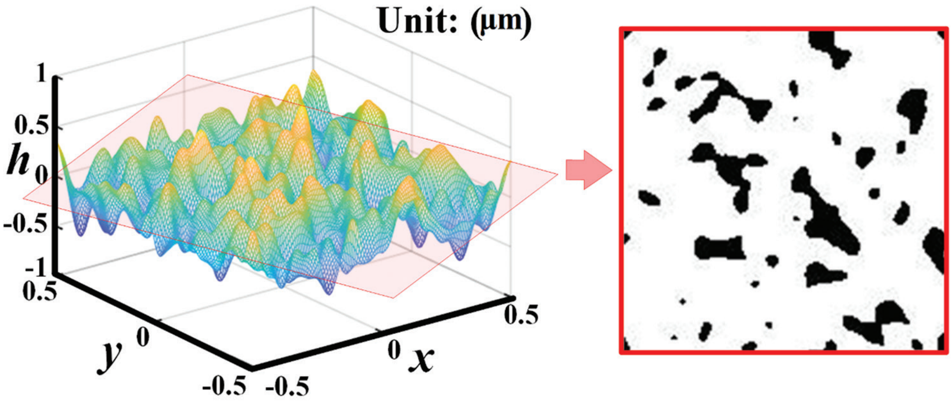


Figure 6. A 2D grayscale image of pores generated based on the 3D Gaussian random model. Note: x, y and h are the axes of the 3D Gaussian model, corresponding to the length, width, and height, respectively

requires more training data and time, making it harder to train. Given the advantages and disadvantages of both numerical features and image input types in the calculation of neural network models, numerical and image datasets of the model were established in this study to obtain more comprehensive information without affecting the calculation speed.

The characteristic parameters of the sintered nano-Ag SEM images were extracted by ImageJ, including average particle size, particle circumference, and porosity, to form a dataset of numerical input. Since the three characteristic parameters extracted are not directional – remaining unchanged regardless of image inversion or rotation – the simulated thermal conductivity in the *x*- and *y*-directions is averaged in the output dataset to obtain the average thermal conductivity as the output of the numerical input model. To fully utilize the 186 sintered nano-Ag microstructure images, the dataset was augmented by flipping each image left and right, flipping it up and down, and rotating it 180°, resulting in three additional images for each original. In other words, the original dataset can be quadrupled, resulting in a total of 744 datasets without additional simulation time. This approach enhances the amount of data used for model training and further improves model accuracy. Notably, the thermal conductivity of the microstructure model based on these three images is the same as that of the original image in the *x*- and *y*-directions. The numerical and corresponding image data serve as inputs, while the simulated thermal conductivity in the *x*- and *y*-directions is the output of the model when establishing the corresponding dataset. This process occurs simultaneously with the training of neural networks. Using the bootstrap method, the numerical and image datasets are divided into a training set and a testing set in a 7:3 ratio for model training.

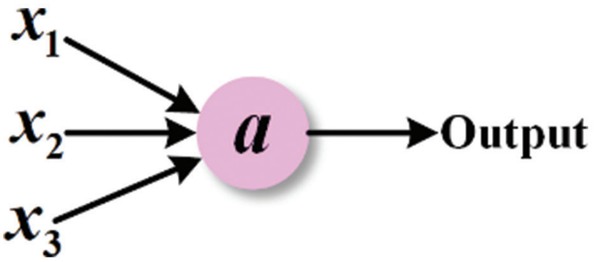


Figure 7. Perceptron structure diagram

2.2.3. Principles and hyperparameter tuning of machine learning models

The ANN consists of multiple fully connected layers, i.e., each neuron is connected to all the neurons in the previous layer.[35] In ANN, a simple model with several inputs and one output is called the perceptron, i.e., from the perspective of a single neuron locally (Figure 7). The perceptron consists of a linear relation and an activation function $\sigma(z)$. The output formula for the single perceptron can be expressed as:

$$a=\sigma(z)=\sigma(\sum_{i=1}^{n}\omega_i x_i+b) \tag{VI}$$

The lines between each neuron in ANN represent a weight coefficient w, with each neuron corresponding to a bias b. In addition, to satisfy the non-linear relationship between input and output, the activation function σ is added. Common activation functions include ReLU, Sigmoid, and Tanh. The ReLU activation function is selected for this study. As a common activation function, ReLU helps avoid the gradient disappearance problem by introducing non-linear transformation and sparse activation. This effectively increases the expressive power of the neural network, making the model more distinguishable. The specific formula of ReLU is expressed as follows:

$$\text{ReLU}: f\left(x\right)=\begin{cases} x \ \text{ if } x>0 \\ \theta x \ \text{ if } x\le 0 \end{cases} \tag{VII}$$

An ANN with more hidden layers and neurons is typically regarded as a deep neural network (DNN). As displayed in Figure 8, ANN is the result of multiple perceptrons in parallel and in series.[36] The network features three hidden layers, and the number of neurons in each hidden layer is denoted as i, j, and k, respectively. The output formula of the ANN network composed of neurons in n layers is expressed as follows:

$$a_i^l=\sigma\left(z_i^l\right)=\sigma\left(\sum_{j=1}^{n}\omega_{ij}^l a_j^{l-1}+b_i^l\right) \tag{VIII}$$

where a^l_i represents the i-th neuron of layer l in ANN; w^l_{ij} represents the weight coefficient from the j-th neuron of layer l-1 to the i-th neuron of layer l; and b^l_i represents the offset corresponding to the i-th neuron in the l layer. When expressed in matrix form, the formula can be simplified as:

$$a^l=\sigma\left(z^l\right)=\sigma\left(W^l a^{l-1}+b^l\right) \tag{IX}$$

The main computation process of neural networks involves forward propagation and backpropagation. The forward propagation algorithm uses several weight coefficient matrices W and bias vector b to perform a series of linear operations and activation operations with input vector x. From the input layer, the output of the previous layer is used to calculate the output of the next layer until the result of the final output layer is obtained. Backpropagation uses forward propagation to calculate the output of the training sample, with the loss function measuring the difference between the predicted and actual values. A typical backpropagation algorithm (BP) minimizes the loss function through iterative optimization using the gradient descent method to identify the appropriate linear coefficient matrix W and bias vector b for the hidden and output layers. The output calculated from the training samples should be equal to or close to the target value. The loss function is calculated as follows:

$$J(W,b,x,y)=\frac{1}{2}\left\|a^l-y\right\|_2^2=\frac{1}{2}\left\|\sigma\left(W^l a^{l-1}+b^l\right)-y\right\|_2^2 \tag{X}$$

Where a^1 is the predicted value of the output layer, and y is the target value of the output.

During model training, the most widely accepted Adam optimizer was selected. The performance and generalization of the model were evaluated using the MSE of the testing set. Subsequently, the hyperparameters of the neural network are tuned based on the MSE value to improve the performance and stability of the model. Grid search is the most widely used hyperparameter search algorithm, which determines the optimal value by searching all the points within the search range.[37] Generally, given a large search range and a small step size, the grid search method can identify the global maximum or minimum value, but it heavily consumes computing resources.

In contrast, a random search does not analyze all parameter values but samples a fixed number of parameters from a specified distribution. Random search can also be used to identify a global optimal solution if the set of random sample points is large enough. Compared with the grid search method, the random search method is faster, but its accuracy cannot be guaranteed. Bayesian optimization, an effective global optimization algorithm, was proposed by Snoek *et al.*[38] to be used for parameter tuning in machine learning. Its concept involves updating the posterior distribution of the objective function by continuously adding sample points through the Gaussian process until the posterior distribution closely approximates the true distribution. In short, it accounts for the last sampling point to better adjust the current sampling point, maximize the benefit of the next sampling point, and avoid unnecessary sampling to the greatest extent. Compared with other methods,

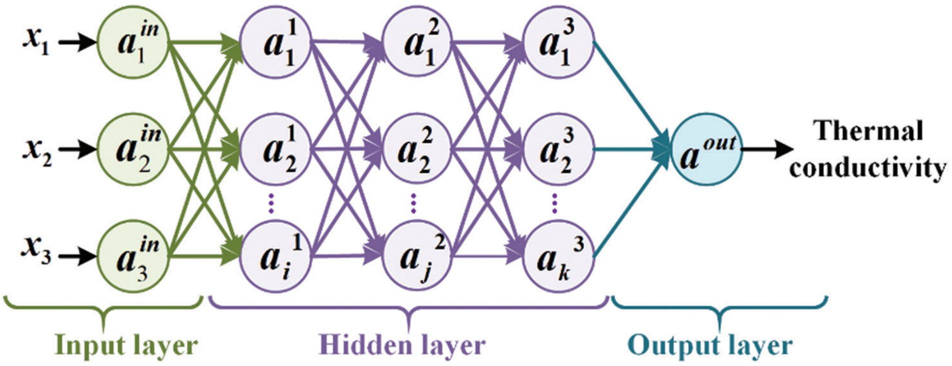


Figure 8. Artificial neural network structure diagram

Bayesian optimization has been widely recognized for its convenience and accuracy.

Bayesian optimization is derived from the famous "Bayes theorem":[39]

$$p(f|D) = \frac{p(D|f)p(f)}{p(D)} \tag{XI}$$

where f, $D = \{(x_1,y_1),(x_2,y_2),\ldots,(x_n,y_n)\}$, x_n, and y_n represent the unknown objective function, the set of observed sampling points, the decision vector, and the observed value of the sampling point, respectively; $p(D|f)$ and $p(f)$ represents the likelihood distribution of y and the prior probability distribution of f (that is, the assumption about the unknown objective function state); and $p(D)$ represents the marginal likelihood distribution of f. The function $p(D)$ is usually difficult to calculate because it involves the product and integral of the probability density function. However, since it does not depend on f, it is treated as a normalized constant in Bayesian optimization. In addition, $p(f|D)$ represents the posterior probability distribution of f, which describes the confidence of the unknown objective function after modifying the prior function from the observed data set.

The Bayesian optimization algorithm consists of two core parts: the probabilistic agent model and the acquisition function. The probabilistic agent model includes the prior probability model and the observation model. The former is $p(f)$, while the latter describes the mechanism by which the observed data are generated, the likelihood distribution $p(D|f)$. The posterior probability distribution $p(f|D)$, containing the observations of the latest evaluation points, is obtained using the Bayesian formula to update the probabilistic agent model. According to the posterior probability distribution, the next most "potential" evaluation point is selected by maximizing the collection function, and an effective collection function can ensure that the selected evaluation point sequence minimizes the total loss value:

$$Loss = \sum_{i=1}^{n} \left| y^{*} - y_i \right| \tag{XII}$$

where y^* represents the optimal solution of the current evaluation point.

The specific calculation process of the Bayesian optimization algorithm involves an iterative process of parameter updates, and its specific algorithm framework is presented in Table 1.[40]

Table 1. Bayesian optimization algorithm calculation process

Bayesian optimization algorithm
For n=1, 2, …, do
a. Obtain the next evaluation point x_{n+1} by maximizing the acquisition function α
b. Get the objective function value of the evaluation point y_{n+1};
c. Augment data $D_{n+1}=\{D_n, (x_{n+1}, y_{n+1})\}$;
d. Update probabilistic proxy model end for

Figure 9 displays the principle of the Bayesian optimization algorithm. Each repeat sampling generates a minimum value for the objective function. After the first random sampling of the function, the second sampling assesses points near the possible minimum value and in regions that have not been sampled. This approach helps avoid entrapment in the local optima, improves proxy function approximation of the true objective function, and facilitates finding the minimum value of the objective function. In simple terms, the Bayesian optimization algorithm selects the next sampling point to maximize the return.

For the tuning method, the type and range of tuning parameters need to be defined. In the ANN model, the learning rate, number of hidden layers, and number of neurons in each layer are selected as the tuning parameters. Among them, the learning rate in the optimizer is an important hyperparameter in the neural network, regulating the step size of each parameter update and directly affecting the convergence speed and performance of the model. The number of hidden layers and neurons in each layer determines the structure of the model. To streamline parameter adjustment, the number of neurons in each layer is set to be the same to ensure that the number of hidden layers and neurons in each layer can be used as two independent parameters for parameter adjustment.

Figure 10 presents the results of Bayesian optimization of hyperparameters in the ANN model. The learning rate ranges from $1e^{-4}$ to 0.1 and is distributed exponentially to improve optimization efficiency. The number of hidden layers is presented as an integer, ranging from 1 to 5, while the number of neurons is presented as an integer, ranging from 1 to 140. The evaluation index MSE (Equation IV) of the model is reflected by the color of the data points, and the corresponding value range is referred to the color scale on the right. Sampling occurs across the entire hyperparameter space, avoiding the local optima. The points around the final optimal result are relatively dense, indicating that the model undergoes fine-tuning and optimization at the later stages. The final hyperparameters include two hidden layers, each with 32 neurons, and a learning rate of $9.29e^{-4}$.

3. Results and discussion

The established ANN model was used to map the relationship between the input characteristics

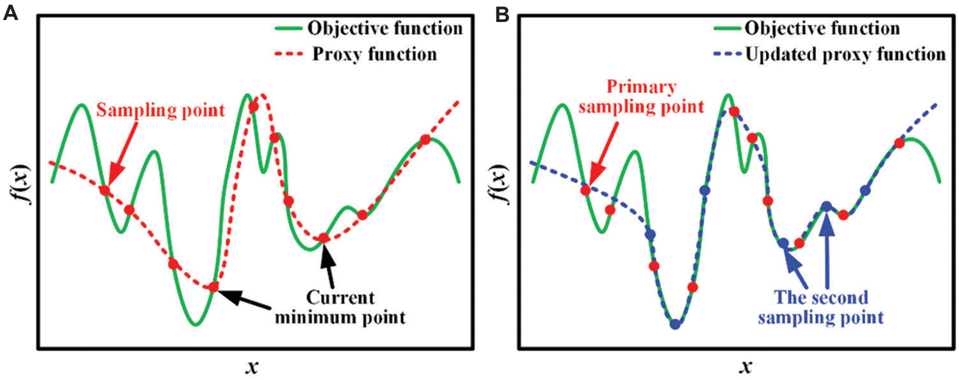


Figure 9. Principle of the Bayesian optimization algorithm. Results of the (A) first sampling and (B) second sampling.

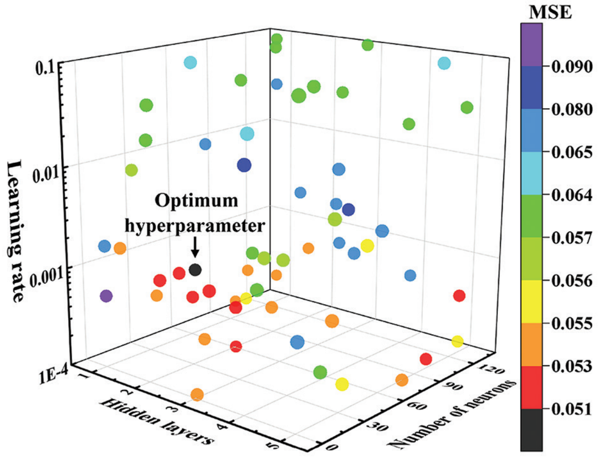


Figure 10. Hyperparameter Bayesian optimization results of the artificial neural network

Abbreviation: MSE: Mean square error.

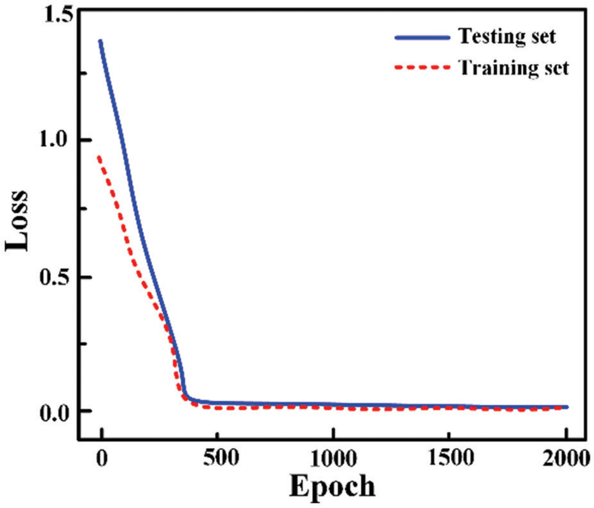


Figure 11. Loss curve of the artificial neural network

(microstructure characteristics of sintered nano-Ag) and the output value (thermal conductivity of sintered nano-Ag). According to the loss function defined by Equation XII, after 2000 training of the epoch, the training loss and testing loss decreased over time. All losses converge at about 400 cycles, dropping below 0.05, and remain stable in the subsequent training cycles (Figure 11). The results indicate that the 500 epochs reflect the actual training effect of ANN, without overfitting in the thermal conductivity prediction of sintered nano-Ag.

In addition to the MSE, the determination coefficient (R^2) was also used to evaluate the performance of the ANN model. The coefficient R^2 is defined by Equation V. A higher R^2 value indicates that the model has better prediction ability for the target parameters. Figure 12 displays the comparison between the predicted and actual thermal conductivity of sintered nano-Ag. As observed, the distribution of data points (blue dots) is focused around the

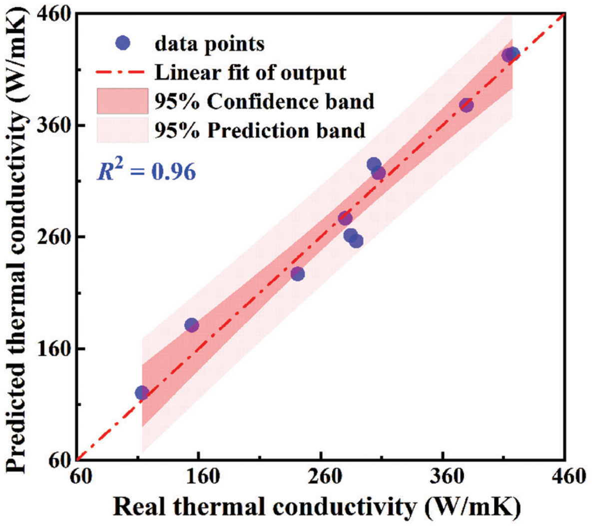


Figure 12. The predicting performance of thermal conductivity testing

ideal prediction trendline (red dashed line), indicating that the predicted parameters are relatively close to the actual parameters. The value of R^2 is as high as 0.96, indicating that the ANN model based on Bayesian optimization has a good predictive ability for the correlation analysis between microstructure characteristics and thermal conductivity of sintered nano-Ag. The 95% confidence region also confirms that the proposed optimal ANN model has good predictive performance. The confidence interval represents the knowledge level of the best fitting line and determines that the true linear fitting output is within the interval. In Figure 12, when the trained ANN model was used to predict thermal conductivity, the confidence interval between the predicted output and the real output was compared; it was found that most of the predicted points overlapped with the actual value, with smaller differences corresponding to higher prediction accuracy.

The factors that affect the thermal properties of sintered nano-Ag, such as sintering process parameters, affect the microstructure of sintered Ag nanoparticles. However, in existing studies, microstructure parameters are rarely used in thermal conductivity models.[30] Notably, our study utilized the microscopic reconstruction method, which combines Gaussian random reconstruction and finite element, effectively studying the thermal conductivity changes as the microstructure changes under the influence of sintering parameters. However, due to limitations in the dataset and the difficulty of obtaining it, we only analyzed the evolution of microscopic grains at different sintering times using the open operation method, without considering the effect of sintering temperature. Nevertheless, if the influence of sintering temperature changes on thermal conductivity is included, the model would remain valid. Rong *et al.*[29] mentioned in their study that features selected from a large descriptor space may limit the predictive accuracy of machine learning models. It is worth noting that we simultaneously used both the image dataset and the feature parameter dataset for training, effectively mitigating this issue and enhancing the model's generalization ability for different types of structures. We are working to integrate the aforementioned factors to study the microstructure evolution of sintered nano-Ag, aiming to obtain a more comprehensive dataset that captures the changes in thermal characteristics with respect to microstructure parameters.

4. Conclusion

We presented the image of sintered nano-Ag microstructure based on Gaussian reconstruction, where feature parameters and image data are separately extracted to form a dataset. The equivalent thermal conductivity of sintered nano-Ag is then predicted using a machine-learning model. In summary, our key findings are as follows:

(i). Image processing and modeling simulation methods for SEM images of sintered nano-Ag microstructure were established by MATLAB and Ansys software. Batch calculation of the thermal conductivity from 2D images of sintered nano-Ag microstructure was performed accordingly. Based on the SEM image characteristics of actual sintered nano-Ag microstructure, 186 images of the microstructure, obtained from using 3-pixel sizes and different sintering times, were reconstructed using the Gaussian random model. The thermal conductivity in the *x*- and *y*-directions of the microstructure plane model was obtained by finite element simulation.

(ii). Given the advantages and disadvantages of both the numerical parameters and the image input types for neural network analysis, the numerical dataset and the image dataset are established, respectively. An image was obtained from the reconstructed model of the sintered nano-Ag microstructure, and the dataset was enhanced four times. The model accuracy can be further improved by enhancing the amount of data used for model training.

(iii). Based on the Bayesian optimized ANN model, the average particle size, circumference, and porosity were taken as input parameters to predict the equivalent thermal conductivity of sintered nano-Ag. The final determination coefficient of the ANN model for the equivalent thermal conductivity prediction is 0.96. The results are of great significance for evaluating the microstructure and physical properties of sintered nano-Ag. Hence, future studies should focus on the physical mechanisms based on existing models.

Acknowledgments

The authors would like to express their gratitude for the support from the National Natural Science Foundation of China (No. 12272012) and acknowledge the AI tool SparkDesk (iFLYTEK) for enhancing the language quality and readability of the manuscript.

Funding

This research was financially supported by the National Natural Science Foundation of China (No. 12272012).

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author contributions

Conceptualization: Jiahui Wei, Yanwei Dai
Formal analysis: Libo Zhao, Jiahui Wei

Investigation: Jiahui Wei, Daowei Wu, Yuting Zhang, Kui Li, Fei Qin
Methodology: Yanwei Dai, Libo Zhao, Jiahui Wei
Writing – original draft: Libo Zhao, Jiahui Wei, Yanwei Dai
Writing – review & editing: Libo Zhao, Yanwei Dai, Jiahui Wei

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

The data presented in this study are available upon request from the corresponding author due to data protection.

References

1. Herboth T, Guenther M, Fix A, Wilde J. Failure Mechanisms of Sintered Silver Interconnections for Power Electronic Applications. In: *IEEE 63rd Electronic Components and Technology Conference*. Las Vegas, NV, USA; 2013. p. 1621-1627.

   doi: 10.1109/ECTC.2013.6575789

2. Ordonez-Miranda J, Hermens M, Nikitin I, Kouznetsova VG, van der Sluis O, Ras MA, Volz S. Measurement and modeling of the effective thermal conductivity of sintered silver pastes. *Int J Therm Sci*. 2016;108:185-194.

   doi: 10.1016/j.ijthermalsci.2016.05.014

3. Signor L, Kumar P, Tressou B, *et al*. Evolution of the thermal conductivity of Sintered silver joints with their porosity predicted by the finite element analysis of real 3D microstructures. *J Electron Mater*. 2018;47:4170-4176.

   doi: 10.1007/s11664-018-6253-2.

4. Sghuri A, Billaud Y, Signor L, Saury D, Milhet X. Experimental investigation of thermal conductivity during aging of nanoporous sintered silver. *Acta Mater*. 2023;257:119109.

   doi: 10.1016/j.actamat.2023.119109

5. Hu X, Martin HA, Poelma R, *et al*. Exploring the process-microstructure-thermal properties relationship of resin-reinforced Ag sintering material for high-power applications via 3D FIB-SEM nanotomography. *Mater Des*. 2024;244:113185.

   doi: 10.1016/j.matdes.2024.113185

6. Zhao Z, Zhang H, Zou G, *et al.* A predictive model for thermal conductivity of nano-Ag sintered interconnect for a SiC die. *J Electron Mater*. 2019;48:2811-2825.

   doi: 10.1007/s11664-019-06984-3

7. Qin F, Hu Y, Dai Y, An T, Chen P. Evaluation of thermal conductivity for sintered silver considering aging effect with microstructure based model. *Microelectron Reliab*. 2020;108:113633.

   doi: 10.1016/j.microrel.2020.113633

8. Qin F, Hu Y, Dai Y, *et al*. Crack effect on the equivalent thermal conductivity of porously sintered silver. *J Electron Mater*. 2020;49:5994-6008.

   doi: 10.1007/s11664-020-08325-1

9. Qin F., Zhao S, Dai Y, Hu Y, An T, Gong Y. Mud-cracking effect of sintered silver layer on quantifying heat transfer behavior of SiC devices under power cycling: Voronoi tessellation model. *IEEE Trans Compon Packag Manuf Technol*. 2022;12(6):964-972.

   doi: 10.1109/TCPMT.2022.3178226

10. Kim YJ, Park BH, Hyun SK, Nishikawa H. The influence of porosity and pore shape on the thermal conductivity of silver sintered joint for die attach. *Mater Today Commun*. 2021;29:102772.

    doi: 10.1016/j.mtcomm.2021.102772

11. Chen H, Du Z, Li X, Zhou H, Liu Z. Identification of pipe inner surface in heat conduction problems by deep learning and effective thermal conductivity transform. *Eng Comput*. 2020;37(9):3505-3523.

    doi: 10.1108/EC-01-2020-0012

12. Huang Q, Hong D, Niu B, Long D, Zhang Y. An interpretable deep learning strategy for effective thermal conductivity prediction of porous materials. *Int J Heat Mass Transfer*. 2024;221:125064.

    doi: 10.1016/j.ijheatmasstransfer.2023.125064

13. Qin G, Wei Y, Yu L, *et al*. Predicting lattice thermal conductivity from fundamental material properties using machine learning techniques. *J Mater Chem A*. 2023;11(11):5801-5810.

    doi: 10.1039/D2TA08721A

14. Li RY, Lee E, Luo TF. A unified deep neural network potential capable of predicting thermal conductivity of silicon in different phases. *Mater Today Phys*. 2020;12:100181.

    doi: 10.1016/j.mtphys.2020.100181

15. Yang ZH, Wu XX, He XD, Guan XF. A multiscale analysis-assisted two-stage reduced-order deep learning approach for effective thermal conductivity of arbitrary contrast heterogeneous materials. *Eng Appl Artif Intell*. 2024;136:108916.

    doi: 10.1016/j.engappai. 2024.108916

16. Kim TH, Park JH, Jung KW, Kim J, Lee EH. Application of convolutional neural network to predict anisotropic

effective thermal conductivity of semiconductor package. *IEEE Access*. 2022;10:51995-52007.

doi: 10.1109/ACCESS.2022.3174882

17. Du CJ, Zou GS, Zhanwen A, *et al*. Highly accurate and efficient prediction of effective thermal conductivity of sintered silver based on deep learning method. *Int J Heat Mass Transfer*. 2023;201:123654.

    doi: 10.1016/j.ijheatmasstransfer.2022.123654

18. Du CJ, Zou G, Feng B, *et al*. Predicting effective thermal conductivity of sintered silver by microstructural-simulation-based machine learning. *J Electron Mater*. 2023;52(4):2347-2358.

    doi: 10.1007/s11664-022-10172-1

19. Long X, Mao MH, Su TX, Su YT, Tian MK. Machine learning method to predict dynamic compressive response of concrete-like material at high strain rates, *Def Technol*. 2023;23:100-111.

    doi: 10.1016/j.dt.2022.02.003

20. Mao M, Wang W, Lu C, Jia F, Long X. Machine learning for board-level drop response of BGA packaging structure. *Microelectron Reliab*. 2022;134:114553.

    doi: 10.1016/j.microrel. 2022.114553

21. Sezer A, Altan A. Detection of solder paste defects with an optimization-based deep learning model using image processing techniques. *Soldering Surf Mount Technol*. 2021;33(5):291-298.

    doi: 10.1108/SSMT-04-2021-0013

22. Long X, Lu CH, Su YT, Dai YH. Machine learning framework for predicting the low cycle fatigue life of lead-free solders. *Eng Failure Anal*. 2023;148:107228.

    doi: 10.1016/j.engfailanal.2023.107228

23. Prisacaru A, Gromala P, Han B, Zhang GQ. Degradation estimation and prediction of electronic packages using data-driven approach. *IEEE Trans Ind Electron*. 2021;69(3):2996-3006.

    doi: 10.1109/TIE.2021.3068681

24. Samavatian V, Fotuhi-Firuzabad M, Samavatian M, Dehghanian P, Blaabjerg F. Iterative machine learning-aided framework bridges between fatigue and creep damages in solder interconnections. *IEEE Trans Compon Packag Manuf Technol*. 2021;12(2):349-358.

    doi: 10.1109/TCPMT.2021.3136751

25. Long X, Lu CH, Shen ZY, Su YT. Identification of mechanical properties of thin-film elastoplastic materials by machine learning. *Acta Mech Sol Sin*. 2023;36:13-21.

    doi: 10.1007/s10338-022-00340-5

26. Kuo HC, Chang CY, Yuan C, Chiang KN. Wafer-level packaging solder joint reliability lifecycle prediction using SVR-based machine learning algorithm. *J Mech*. 2023;39:183-190.

    doi: 10.1093/jom/ufad016

27. Samavatian V, Fotuhi-Firuzabad M, Samavatian M, Dehghanian P, Blaabjerg F. Correlation-driven machine learning for accelerated reliability assessment of solder joints in electronics. *Sci Rep*. 2020;10(1):14821.

    doi: 10.1038/s41598-020-71926-7

28. Wei H, Zhao SS, Rong QY, Bao H. Predicting the effective thermal conductivities of composite materials and porous media by machine learning methods. *Int J Heat Mass Transfer*. 2018;127:908-916.

    doi: 10.1016/j.ijheatmasstransfer.2018.08.082

29. Rong QY, Wei H, Huang XY, Bao H. Predicting the effective thermal conductivity of composites from cross sections images using deep learning methods. *Compos Sci Technol*. 2019;184:107861.

    doi: 10.1016/j.compscitech.2019.107861

30. Fei W, Narsilio GA, Disfani MM. Predicting effective thermal conductivity in sands using an artificial neural network with multiscale microstructural parameters. *Int J Heat Mass Transfer*. 2021;170:120997.

    doi: 10.1016/j.ijheatmasstransfer.2021.120997

31. Carslaw HS, Jaeger JC. *Conduction of Heat in Solids*. London: Clarendon Press; 1992.

    doi: 10.1007/978-1-4939-2565-0_2

32. Märtens M, Izzo D, Krzic A, Krzic A, Cox D. Super-resolution of PROBA-V images using convolutional neural networks. *Astrodynamics*. 2019;3:387-402.

    doi: 10.1007/s42064-019-0059-8

33. Manan A, Zhang P, Ahmad S, Ahmad J. Optimizing hybrid fibre-reinforced polymer bars design: A machine learning approach. *J Polym Mater*. 2024;41(1):15-44.

    doi: 10.32604/jpm.2024.053859

34. Suryawanshi A, Behera N. Application of machine learning for prediction dental material wear. *J Polym Mater*. 2023;40(3-4):305-316.

    doi: 10.32381/JPM.2023.40.3-4.11

35. Krishnamoorthy K, Prabhu N. Tensile failure characterization of glass/epoxy composites using acoustic emission RMS data. *J Polym Mater*. 2023;40(3-4):215-226.

    doi: 10.32381/JPM.2023.40.3-4.7

36. Peng H, Bai X. Comparative evaluation of three machine learning algorithms on improving orbit prediction accuracy. *Astrodynamics*. 2019;3(4):325-343.

    doi: 10.1007/s42064-018-0055-4

37. Li WB, Song Y, Cheng L, Gong SP. Closed-loop deep neural

network optimal control algorithm and error analysis for powered landing under uncertainties. *Astrodynamics*. 2023;7(2):211-228.

doi: 10.1007/s42064-022-0153-1

38. Snoek J, Larochelle H, Adams RP. Practical bayesian optimization of machine learning algorithms. *Adv Neural Inf Process Syst*. 2012;25:2951-2959.

    doi: 10.5555/2999325.2999464

39. Kaiser J, Xu CR, Eichler A, *et al*. Reinforcement learning-trained optimisers and bayesian optimisation for online particle accelerator tuning. *Sci Rep.* 2024;14(1):15733.

    doi: 10.1038/s41598-024-66263-y

40. Shahriari B, Swersky K, Wang ZY, Adams RP, Freitas N. Taking the human out of the loop: A review of bayesian optimization. *Proc IEEE*. 2015;104(1):148-175.

    doi: 10.1109/JPROC. 2015.2494218

ACCSCIENCE
PUBLISHING

International Journal of AI for
Materials and Design

ORIGINAL RESEARCH ARTICLE

Machine-learned molecular modeling of ruthenium: A Kolmogorov-Arnold Network approach

Zhiyu An[1] **and Jingjie Yeo**[2*]

[1]Department of System Engineering, Cornell University, Ithaca, New York, United States of America

[2]Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, New York, United States of America

(This article belongs to the *Special Issue: Applications of Deep Learning in Advanced Materials Processing*)

Abstract

Developing refractory high-entropy superalloys (RSAs) with performance advantages over nickel-based alloys is a critical frontier in materials science. Body-centered cubic (bcc)-based RSAs have attracted significant attention, with ruthenium (Ru) playing a key role in forming two-phase regions of A2 (disordered bcc) + B2 (ordered bcc), which could lead to superalloy-like microstructures. This study introduces the application of the Kolmogorov-Arnold Network (KAN) model to predict the mechanical and thermodynamic properties of Ru while comparing its performance against other commonly used machine-learned models. Utilizing density functional theory calculations as training data, the KAN model demonstrates superior accuracy and computational efficiency compared to conventional methods, while reducing descriptor complexity. The model accurately predicts a range of properties, including elastic constants, thermal expansion coefficients, and various moduli, with discrepancies within 6% of experimental reference data. Molecular dynamics simulations further validate the model's efficacy, accurately capturing Ru's phase transitions from hexagonal close-packed (hcp) to face-centered cubic structure and the melting point. This work presents the first application of KAN in materials science, demonstrating how its balanced performance and efficiency provide a new pathway for designing advanced materials, with unique advantages over conventional machine learning approaches in predicting material properties.

Keywords: Ruthenium; Kolmogorov-Arnold Network; Machine learning; Mechanical properties; Thermodynamic properties; Density-functional theory; Molecular dynamics

1. Introduction

Researchers have long sought to improve nickel-based superalloys for greater technical, economic, and social benefits by reducing their weight while improving their material properties under extreme loads. Naka and Khan[1] proposed a promising direction by combining B2-ordered NiAl compounds with transition metals having disordered body-centered cubic (bcc) structures (A2). This approach creates an A2 + B2 microstructure like the γ-γ' structure in traditional superalloys. These alloys, which meet the criteria for

***Corresponding author:**
Jingjie Yeo
(jingjieyeo@cornell.edu)

Citation: An Z, Yeo J. Machine-learned molecular modeling of ruthenium: A Kolmogorov-Arnold Network approach. *Int J AI Mater Design*. 2025;2(1):21-38.
doi: 10.36922/ijamd.8291

Received: December 30, 2024

Revised: January 26, 2025

Accepted: February 7, 2025

high-entropy alloys (HEAs)[2,3] and complex concentrated alloys,[4] demonstrate potential as refractory high-entropy superalloys (RSAs). In A2+B2 systems, the ordered intermetallic B2 phase is particularly important, as it often acts as a strengthening precipitate, similar to the role of γ' in traditional superalloys.[5] In addition, B2 phases often exhibit excellent strength retention at elevated temperatures and creep resistance compared to disordered phases, enhancing the alloy's performance under sustained loads at high temperatures.[6] Many B2 phases, particularly those containing Al, demonstrate superior oxidation resistance by forming protective oxide scales. However, some elements discovered by Naka and Khan are still not fully represented in RSAs that are currently synthesized, especially Pd, Pt, and ruthenium (Ru), which all form stable B2 phases with Al.[7]

Among the elements critical to RSA development, Ru has garnered significant attention due to its unique properties and potential ability to form stable ordered phases in multi-component systems. Cerba *et al.*[8] demonstrated that AlRu alloys melt at 2060°C and interact with Ru through eutectic reactions at 1920°C, suggesting potential routes for exploring A2 + B2 microstructures. Moreover, the B2 phase exists in some binary refractory metal systems involving Ru, such as Nb-Ru and Ru-Zr,[7] indicating its versatility in phase formation. Despite its importance, there remains a significant gap in our understanding of Ru's molecular structure and its influence on mechanical and thermodynamic properties, especially in the context of computational materials science. Specifically, there is currently no machine learning (ML) model capable of accurately predicting Ru's molecular structures along with their corresponding mechanical properties in a wide range of temperatures. This lack of comprehensive data and predictive tools hinders our ability to engineer the structures and material properties of RSAs where Ru is used. The corresponding knowledge gap hinders the efficient design and development of advanced RSAs and other high-temperature materials that incorporate Ru.

Ru is a platinum-group metal that crystallizes in a hexagonal close-packed (hcp) structure at ambient conditions.[9] Experimentally, Ru displays superconductivity with a critical temperature below 1699 K.[10] A hcp Ru also maintains its structural stability up to pressures of 150 GPa and temperatures of 960 K.[11] Güler *et al.*[12] used molecular dynamics (MD) simulations with an empirical interatomic potential to explore the temperature-dependent elastic, mechanical, and anisotropic properties of hcp Ru. Likewise, Lu *et al.*[13] used density functional theory (DFT) to calculate the high-pressure phase transition of Ru. However, computational approaches, such as DFT and MD, are often computationally intensive and limited in their ability to rapidly explore the vast parameter space of Ru's structure-property relationships. The development of an ML model that can accurately and rapidly predict Ru's molecular structures and associated mechanical properties would be a significant advancement. Such a model could potentially allow for a faster and more comprehensive exploration of Ru's structural and mechanical behavior under various environmental conditions.

High-throughput *ab initio* simulations are increasingly used for exploring the basic properties of engineering materials, such as determining the material properties.[14] When combined with materials informatics, ML models accelerate the exploration of these material properties.[15] The goal of the ML model is to supplement or replace high-fidelity modeling methods at the quantum chemical or classical level to predict the properties of molecules or materials directly from their structure or their chemical composition. ML methods usually use available experimental and *ab initio* data to build accurate statistical models that can be used to predict the properties of materials. This innovative combination can significantly enhance predictive capabilities and has applications in many fields of materials research, such as dielectric polymers,[16] critical temperatures of superconducting materials,[17] crystal structures,[18] perovskites,[19] and nanostructures.[20] Many of these properties were measured with high accuracy using ML approaches. For instance, de Jong *et al.*[21] developed a model using supervised ML with gradient-boosting regression to predict elastic properties of inorganic polycrystalline compounds, such as the bulk modulus K and shear modulus G. For metallic systems, Song *et al.*[22] constructed a general-purpose neural network potential (NNP) for 16 elemental metals and their alloys that achieves superior accuracy compared to traditional embedded-atom method potentials while maintaining computational efficiency. Their potential accurately captures complex interface behavior, phase stability, and mechanical properties across diverse metallic systems. Liyanage *et al.*[23] developed a NNP for the Cu-W system using the Behler-Parrinello framework. Their potential accurately reproduces metallurgically relevant properties, including elasticity, stacking faults, dislocations, and thermodynamic behavior in both elemental Cu and W, as well as Cu-W interfaces and solid solutions. This NNP enables large-scale atomistic simulations to investigate phenomena such as the influence of interface stress on mechanical properties in Cu-W nano-multilayer systems. Another advantage of ML models is their generalizability across different material systems, as it is easier to incorporate crystal data with defects.[24] Compared with other interatomic potentials, ML potentials are highly flexible, allowing for facile adjustment of parameters and refinement of datasets to improve accuracy and predictive performance.[25] This flexibility allows ML models to adapt to new data and evolve continuously, providing a dynamic and scalable approach for studying a wide range of materials and their complex behaviors. In addition, ML models enable faster updates as more data becomes available, streamlining the process of material discovery and optimization.

Among these ML models, the recently introduced Kolmogorov-Arnold Network (KAN), inspired by the Kolmogorov-Arnold Representation theorem, provides a new way to model complex systems.[26] Unlike multilayer perceptrons (MLPs) with fixed activation functions on nodes, KAN has a learnable activation function on the edge, parameterized as a spline function. This seemingly simple change enables KAN to outperform MLP in terms of accuracy and interpretability. In data fitting and partial differential equation solving tasks, KAN has demonstrated faster neural scaling laws than MLP, achieving comparable or better accuracy with a smaller network.[27] The architectural advantage of KAN makes it particularly suited for materials science applications. Its ability to process complex, non-linear relationships makes it ideal for tackling quantum phenomena, like Anderson localization, where disorder in a system leads to the localization of electronic wave functions.[28] KANs excel in extracting mobility edges in various tight-binding models, such as the Mosaic model[29] and the Aubry-André model,[30] where previous methods may struggle. Material properties often follow complex, non-linear relationships that cannot be well-captured by standard activation functions, making KAN's learnable activation functions especially valuable in this domain. The model's ability to discover and represent underlying physical patterns without requiring explicit physical constraints in the training process represents a significant advancement in materials informatics. By allowing users to incorporate prior knowledge and assumptions, KAN can collaborate with human intuition to simplify complex expressions while maintaining high accuracy. This makes KAN a powerful tool for predicting physical properties in materials and provides a robust framework for handling disorders and defects in computational material studies. Therefore, integrating KAN into the materials informatics workflow can enhance our understanding of complex material phenomena, accelerate the discovery of new materials with desired properties, and provide results that are easier to interpret than traditional neural networks.

Our study addresses the lack of ML models specifically tailored for Ru and introduces the novel application of KAN to predict material properties and construct ML interatomic potentials (Figure 1). Compared to other established neural network architectures including CalHousNet Feedforward Neural Network (CalHousNet), Accurate Neural Network Engine for Molecular Energies (ANI), Continuous-filter Convolutional Neural Network (SchNet), and traditional graph neural networks (GNNs), our approach simplifies data preprocessing and significantly reduces training time. When benchmarked against experimental data, our model demonstrates superior accuracy in predicting mechanical properties using molecular statics simulations, highlighting the precision of KAN. Furthermore, we successfully generated an interatomic potential capable of accurately capturing key physical phenomena using MD simulations, including phase transitions and the melting point of Ru. This development not only underscores the advantages of KAN in computational materials science but also lays the groundwork for incorporating additional elements into the framework. This advancement enables the development of multi-element interatomic potentials, potentially facilitating accurate simulations of more complex high-entropy material systems.

2. Methods

2.1. Molecular models and simulations

First-principles calculations were performed within DFT through the Quantum Open-Source Package for Research in Electronic Structure, Simulation, and Optimization (Quantum ESPRESSO) package,[31,32] using the Perdew–Burke–Ernzerhof[33] exchange-correlation functional and projector-augmented plane wave[34] potentials. We optimized the wavefunction and charge density parameters to cutoffs of 52.123 and 353.301 Ry, respectively, and used a 4 × 4 × 4 Monkhorst-Pack grid for k-point sampling. With this scheme, the total energy and force converged at 10^{-5} Ry and 10^{-4} Ry/Bohr, respectively. Through these calculations, we generated 699 data points to relate the changes in the volume and energy of the element Ru under various stress conditions, facilitating detailed analysis of its structural properties. From the DFT calculations, we obtained the initial crystal structure parameters, including lattice constants (a, b, c), angles (α, β, γ), unit cell volume, and atomic positions in both fractional (a, b, c) and Cartesian (x, y, z) coordinates. The DFT calculations further provided the corresponding total energies, stress tensors, and atomic forces. This combined dataset was then processed and structured into a comprehensive dataset using the pymatgen (Python Materials Genomics) package.[35] This dataset size is comparable to typical single-element training sets in literature; for instance, previous studies have demonstrated that accurate potentials can be developed with 461 and 284 structures for Ni and Mo, respectively.[36] While multi-element systems typically

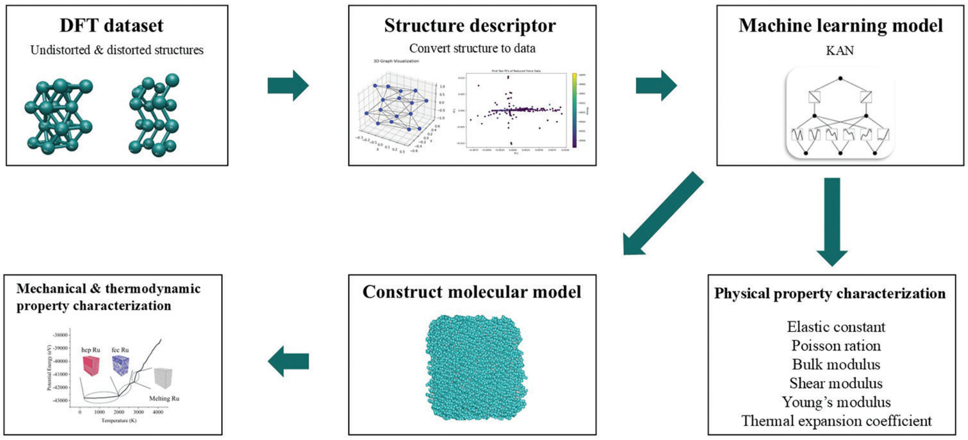


Figure 1. Schematic of the workflow of constructing the dataset using DFT calculations, converting the structures into machine-learnable descriptors, constructing the ML model using KAN, and measurements of material properties either directly or by constructing ML interatomic force fields for MD simulations.

Abbreviations: DFT: Density functional theory; ML: Machine learning; KAN: Kolmogorov-Arnold Network; MD: Molecular dynamics.

require thousands of structures to capture diverse atomic environments and interactions, single-element systems can achieve good accuracy with several hundred carefully selected configurations that comprehensively sample the relevant phase space.

MD simulations were performed using the open-source Large-scale Atomic/Molecular Massively Parallel Simulator (LAMMPS) software package to calculate elastic constants and melting point.[37] The system was constructed using periodic boundary conditions with an hcp lattice structure in a 10 × 10 × 10 unit cell configuration. Interatomic interactions were described using our tabulated KAN potential combined with a Lennard-Jones potential in a hybrid/overlay scheme. To calculate the elastic constants, the system was constructed with periodic boundary conditions using a lattice parameter of 2.70 Å, which was adopted from crystallographic data for Ru's hcp structure and documented in the crystallographic database maintained by Springer Materials.[38] We then performed energy minimization of this initial structure with a convergence criterion of 10^{-6} eV/Å for forces and 10^{-8} eV for energy, followed by anisotropic box relaxation. The elastic constants were then calculated through strain-stress relationships: C_{11}, C_{12}, and C_{13} were determined by applying uniaxial strain, for example, by applying a strain along the x-axis and measuring the resulting stress response in the x-, y-, and z-directions. C_{44} was determined through shear deformation in the xy-plane using a triclinic box transformation. All calculations were performed under quasi-static conditions, with system relaxation achieved through energy minimization after each deformation step. The melting point was determined using a heating method, where the system was first equilibrated at 300 K using a constant Number, Pressure, and Temperature (NPT) ensemble, followed by gradual heating from 300 to 3000 K. The temperature was controlled using a Nosé-Hoover thermostat with a damping parameter of 1.0, and the pressure was maintained at 0 bar using a Parrinello-Rahman barostat during the NPT phase. Neighbor lists were updated every timestep with a cutoff distance of 0.3 Å and a bin-based approach.

2.2. Training data generation

Developing robust ML models for material science heavily relies on training data that encompasses a wide variety of atomic environments. To this end, we performed structure relaxation on all symmetrically distinct configurations within a 16-atom supercell of Ru arranged in an hcp structure. Our objective was to meticulously explore the potential energy surface by optimizing these configurations to their lowest energy states, ensuring the atomic positions and lattice parameters closely matched experimental and theoretical benchmarks. This optimization would ensure that the material's behavior under varying temperatures and pressures was captured accurately. We focused on three primary structural types in our study:

(i) Undistorted ground state structure: This represents the element's most stable configuration, free from external strains or forces.

(ii) Distorted structures: By applying strains ranging from −10% to +10% in six distinct modes – uniaxial tension, uniaxial compression, biaxial tension, biaxial compression, shear, and torsional strain – to the bulk conventional cell, we generated atomic structures to analyze the material's behavior under mechanical stress.[39]

2.3. Training data processing

To effectively process crystal structures for ML applications, the Python Materials Genomics (pymatgen) Library[35] was utilized in conjunction with the Smooth Overlap of Atomic Positions (SOAP) descriptor, implemented through the DScribe Python package,[40] to convert structural data into tensor representations. SOAP provides a robust framework for representing atomic environments, as it ensures rotational, translational, and permutational invariance of the structural descriptors. This invariance means that equivalent atomic configurations yield identical representations regardless of rigid body transformations or reordering of atoms, which is essential for reliable structure-property predictions in crystalline systems. Each structure's atomic positions were denoted as $r_i = x_i, y_i, z_i$, where i denotes the index of the atom. These position vectors were flattened into a one-dimensional tensor for each structure:

$$T_{structure}=x_1\,y_1\,z_1 \ldots x_n\,y_n\,z_n \tag{I}$$

where n is the total number of atoms in the structure. The tensors from each structure were then stacked to form a two-dimensional tensor T, representing the entire dataset and making it suitable for ML models:

$$T=\begin{bmatrix} T_{structure\,1} \\ \cdots \\ T_{structure\,m} \end{bmatrix} \tag{II}$$

where m is the number of structures.

To address the high dimensionality of our data, we applied principal component analysis (PCA). We centered the data by subtracting the mean of each feature to produce a mean-centered data matrix:

$$X_{centered}=X-\mu \tag{III}$$

where X is the original data matrix and μ is a vector containing the mean values of each feature.

Next, we computed the covariance matrix C from the mean-centered data:

$$C=\frac{1}{m-1}X^{T}_{centered}X_{centered} \tag{IV}$$

where m is the number of structures.

We then performed eigen decomposition to extract eigenvalues (Λ) and eigenvectors (V), satisfying:

$$CV = V\Lambda \tag{V}$$

where V is the matrix of eigenvectors (principal components [PCs]) and Λ is the diagonal matrix of eigenvalues. The data were then projected onto the PCs to obtain a reduced data matrix X:

$$X=X_{centered}\,V_{reduced} \tag{VI}$$

where $V_{reduced}$ includes eigenvectors corresponding to the largest eigenvalues that capture 95% of the variance.

In the PCA of crystal structures, Figure 2A elucidates the cumulative explained variance ratio as a function of component number. The first PC accounts for 80.5% of the total variance, demonstrating its dominant role in capturing the dataset's primary features. The second PC contributes an additional 12.2%, bringing the cumulative explained variance to 92.7%. Notably, the first three PCs collectively explain 97.6% of the dataset's variance, surpassing the often-used 95% threshold for dimensionality reduction.[41] This rapid accumulation of explained variance, visualized by the steep rise in the cumulative contribution line, exhibits a characteristic elbow-shaped curve. The shape of this curve, with its sharp initial increase followed by a plateau, indicates highly efficient dimensionality reduction, suggesting that the complex crystal structure data can be effectively represented using just these three PCs.

Figure S1 presents a heatmap of structural PCA loadings, revealing the complex relationships between the original structural features and the three PCs. In this context, "loadings" refer to the coefficients that describe how much each original feature contributes to a given PC, with red indicating positive correlations and blue indicating negative correlations. The first PC displays a nuanced pattern of strong positive and negative loadings across features, suggesting that it encapsulates a multifaceted combination of structural attributes. This component exhibits the most variation in loadings, indicating its capacity to capture complex, opposing relationships among the original features. In contrast, the second PC exhibits predominantly positive loadings, with several features displaying very strong positive correlations. Importantly, the magnitude of loadings across all PCs indicates that each original structural feature is well-represented in the PC space, suggesting that the PCA effectively captures the key variance in the dataset without substantial loss of information from any feature.

Meanwhile, for energy and force data extracted from DFT calculations, we employed different preprocessing techniques. The energy data were directly converted into the tensor format without further modifications. In contrast, the force data, representing a three-dimensional vector (one per atom with components $F_i = (F_{ix}, F_{iy}, F_{iz})$ representing the forces along the x-, y-, and z-directions), required additional processing. We also applied PCA to reduce its dimensionality, as discussed above. Figure 2B

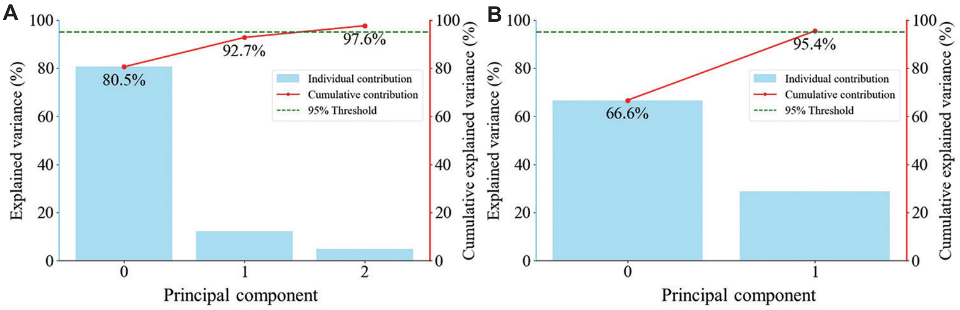


Figure 2. Principal component analysis (PCA) reveals dominant structural features and efficient dimensionality reduction in the structure and force data. (A) Cumulative explained variance ratio vs. structure component number. PC0 accounts for 80.5% of the total variance, with PC1 adding 12.2%, PC2 adding 4.9%, and the first three PCs collectively explaining 97.62%, exceeding the 95% threshold. (B) Cumulative explained variance ratio vs. force component number. PC0 accounts for 66.6% of the total variance, with PC1 adding 28.8%, exceeding the 95% threshold.
Abbreviation: PC: Principal component.

elucidates the cumulative explained variance ratio as a function of component number. The first PC accounts for 66.6% of the total variance, demonstrating its dominant role in capturing the dataset's primary features. The second PC contributes an additional 28.8%, bringing the cumulative explained variance to surpass the often-used 95% threshold for dimensionality reduction. Figure S2 presents a heatmap of force PCA loadings, indicating that each original force feature is well-represented in the PC space, suggesting that the PCA effectively captures the key variance in the dataset without substantial loss of information from any feature.

After performing PCA on the structural data, we normalized the target outputs to address the inherent scale differences between energy and force values, such that:

$$Z = \frac{X - \mu}{\sigma} \tag{VII}$$

where Z is the normalized value, X is the original value, μ is the mean of the data, and σ is the standard deviation of the data. This normalization is crucial as energy and force are measured in different units and exhibit distinct ranges of magnitudes, which could potentially bias the model training. By normalizing these quantities, we ensure that both energy and force components contribute equally to the loss function during model training, preventing the larger-scale variable from dominating the optimization process. This standardized representation of target variables helps maintain balanced gradients during training and enables the model to learn the relationships between structural features and both energy and force predictions with equal emphasis. The normalization process preserves the underlying data patterns while adjusting the scale, and the numerical ranges are standardized to comparable scales (Figure 3). This consistency in patterns before and after normalization confirms that our preprocessing approach effectively standardizes the data without distorting its inherent features.

Subsequently, we optimized the training process by leveraging graphical processing units (GPUs), which allowed for significantly faster computation and more efficient handling of complex tensor operations required by our ML algorithms. We configured our computational framework to dynamically allocate tasks across available GPUs, maximizing resource utilization and reducing processing times. The dataset is split into 60% for training, 20% for validation, and 20% for testing to ensure a comprehensive evaluation of model performance. As displayed in Figure 4, the distribution of samples across the PCs exhibits consistent cross-shaped patterns across all three subsets. This similarity in data distribution confirms that our splitting strategy has effectively maintained the representativeness of the original dataset in each subset. The validation set allows us to monitor model performance during training and prevent overfitting, while the test set provides an unbiased evaluation of the final model performance.

2.4. KAN model

The KAN model was configured with a network width of [3, 4, 3] and utilized a grid size of 8 across three dimensions (k = 3).[27] To enhance the training process, we employed the Adam optimizer, a stochastic gradient descent method known for its ability to compute adaptive learning rates for each parameter.[42] This optimization method is particularly effective in managing sparse gradients and dynamically adjusting the learning rate, which significantly improves the speed of convergence and the overall performance of the model.

By adopting the Adam optimizer, we ensured that our model efficiently navigated through the complex loss landscape, which led to faster convergence and improved predictive accuracy. In addition, we implemented L2

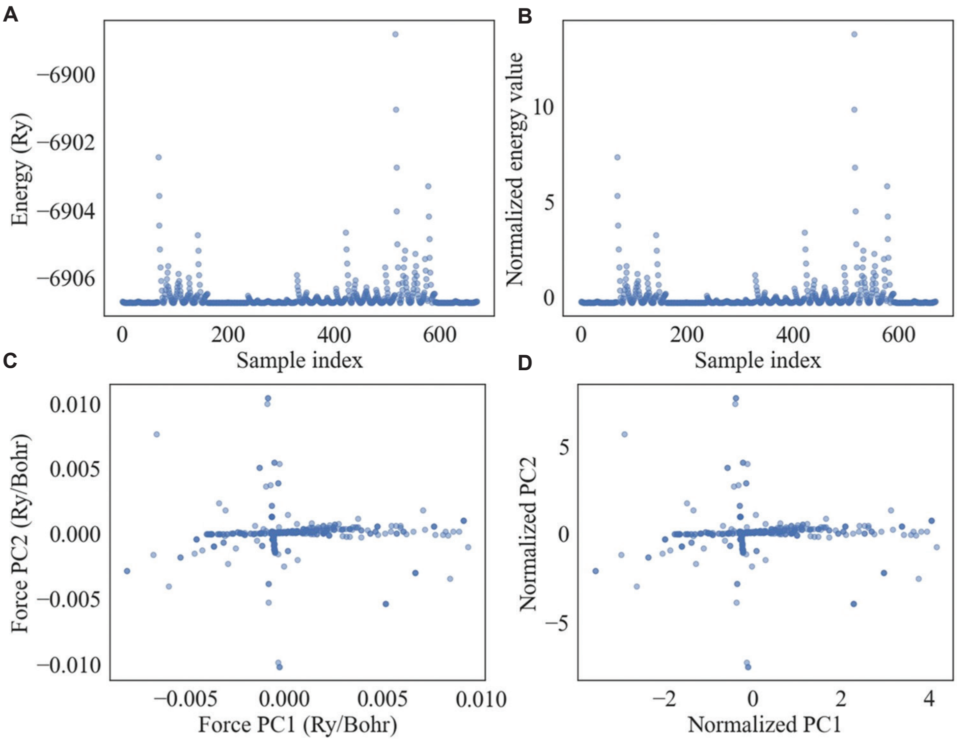


Figure 3. Distribution patterns before and after normalization. (A) Original energy values of the dataset samples; (B) normalized energy values displaying preserved distribution pattern; (C) original force features in the PC1-PC2 space; and (D) normalized force features demonstrating maintained relative relationships. The consistent patterns between pre- and post-normalization data validate the effectiveness of the standardization process.

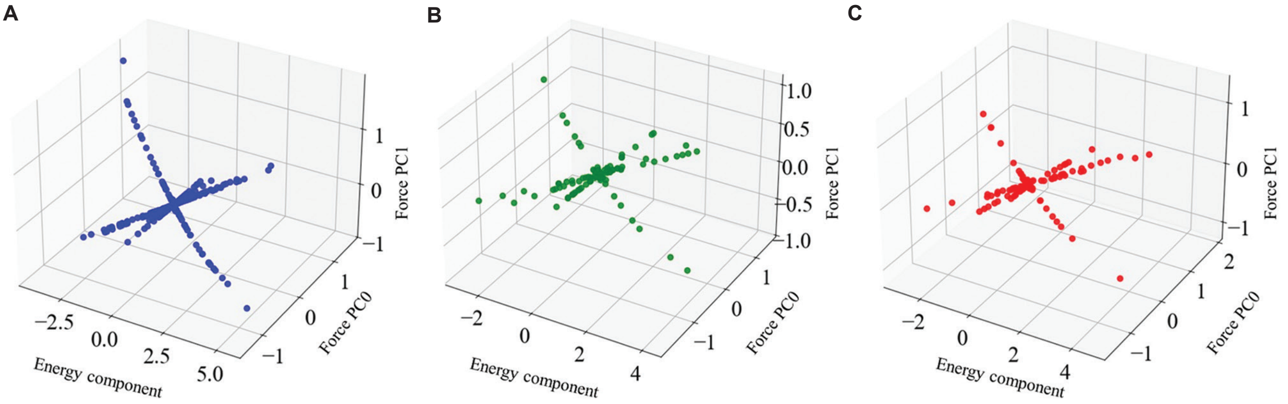


Figure 4. Scattering plots of (A) training dataset, (B) validation dataset, and (C) test dataset, demonstrating consistent cross-shaped patterns across all three subsets.

regularization ($\lambda = 0.02$)[43] and entropy regularization (λ_entropy = 4.0)[32] to enhance the model's ability to generalize. This is crucial for preventing overfitting and maintaining robust performance on new, unseen data.

2.5. CalHousNet feedforward neural network model (CalHousNet)

We implemented CalHousNet, a fully connected feedforward neural network using the PyTorch framework.[44] The model employs a sequential architecture comprising three distinct layers designed for regression analysis. The input layer accepts three features, corresponding to the reduced-dimensionality data obtained through PCA of atomic structures. These inputs are processed through a hidden layer containing three nodes, where each node implements a linear transformation followed by a ReLU activation function.[45] This activation introduces essential non-linearity into the model, enabling the capture of complex relationships between input features and target outputs. The output layer consists of a single node performing a final linear transformation, specifically configured for continuous value prediction in regression tasks.

The training process employs the Adam optimizer with a learning rate of 0.001.[42] This optimizer was selected for its ability to dynamically adjust learning rates for each parameter based on first- and second-moment estimates of the gradients, enhancing both convergence speed and model performance. The training procedure processes data in batches to optimize memory usage and computational efficiency, leveraging stochastic gradient descent for gradient computation. This batch processing approach, combined with multiple training epochs, allows the model to progressively refine its predictions through repeated iterations over the complete dataset.

2.6. Graph neural network model

We developed a GNN model specifically designed to handle the structural complexities of Ru crystals. The model presents atomic structures as graphs, where each node corresponds to an atom and incorporates both static and dynamic properties. The node feature vector combines the constant atomic number (Z = 44 for Ru) with the components of the force vector acting on each atom (F_{ix},F_{iy},F_{iz}). Thus, each node incorporated both elemental and dynamic properties and is represented as:

$$x_i = [Z, F_{ix}, F_{iy}, F_{iz}] \qquad \text{(VIII)}$$

The edge structure of the graph is determined by a predefined connectivity matrix characteristic of Ru crystal structures. Each edge carries geometric information, specifically the Euclidean distance between connected atoms, calculated from their respective position vectors. This distance-based edge attribution ensures that spatial relationships between atoms are explicitly encoded in the graph structure, allowing the model to learn from both local atomic environments and their spatial arrangements:

$$d_{ij} = ||r_i - r_j|| \qquad \text{(IX)}$$

where r_i and r_j are the position vectors of atoms i and j, respectively. The neural network architecture implements a hierarchical processing scheme through multiple graph convolutional layers. The initial layer transforms the four-dimensional input features into a 64-dimensional latent space, initiating the abstract representation of the atomic environment. The network progressively expands this representation through subsequent layers, first to 128 dimensions and then to 256 dimensions. Each convolutional operation is followed by a ReLU activation function,[45] introducing non-linearity and sparsity into the representation, thereby focusing the model's attention on the most significant features of the atomic structure.

The feature aggregation process culminates in a global max pooling operation that consolidates information across the entire graph structure. This pooled representation then passes through a series of fully connected layers, systematically reducing the dimensionality from 256 to 128, and finally to the output value representing the predicted energy and force of the crystal structure.

The model optimization utilizes the Adam optimizer with an initial learning rate of 0.001, carefully chosen to balance training efficiency with optimization stability. This configuration allows for effective convergence while mitigating the risk of overshooting the optimal parameters during training. The gradient-based optimization process enables the model to learn the complex relationships between atomic structure and crystal energy while maintaining numerical stability throughout the training procedure.

2.7. Accurate neural network engine for molecular energies model

We implemented a standard ANI model following the original framework specifications.[46] The model utilizes an atomic environment vector (AEV) computed with radial and angular cutoffs of 5.2 and 3.5 Å, respectively.[47] The radial basis functions employ EtaR parameters of 16.0, 8.0, 4.0, and 2.0, with corresponding ShfR values of 0.0, 0.9, 1.8, and 2.7. The angular component uses EtaA values of 8.0, 4.0, and 2.0, a zeta parameter of 32.0, and both ShfA and ShfZ values set to 0.0, 0.9, and 1.8.

The neural network architecture consists of three primary blocks processing the AEV inputs. The first block transforms the input to 128 dimensions using a linear layer with SELU activation,[48] followed by layer normalization and 0.1 dropout. The second block maintains this dimension with identical transformation and regularization schemes. The third block halves the dimensionality while retaining SELU activation and layer normalization. The network concludes with a linear transformation to three output dimensions.

We also employed a batch size of 16 across 70 epochs of training, with an initial learning rate of 0.001 managed by the Adam optimizer. The learning rate was dynamically adjusted using a ReduceLROnPlateau scheduling mechanism, which reduced the rate by a factor of 0.5 when validation performance plateaued for five consecutive epochs.[44] Training stability was maintained through gradient clipping with a maximum norm of 1.0, while model selection was guided by an early stopping mechanism monitoring validation loss.

2.8. Continuous-filter convolutional neural network model

We implemented the SchNet model, a specialized deep-learning architecture designed for molecular systems.[49] The model consists of an embedding layer followed by multiple interaction blocks and separate prediction heads for energy and forces. The input features are first normalized through a batch normalization layer, then transformed into a 128-dimensional hidden representation through a linear embedding layer with subsequent batch normalization. The model comprises three SchNet interaction blocks containing a filter network and two dense layers. The filter network processes distance-based features through linear transformations and ReLU activations, while the dense layers maintain the hidden dimension of 128 channels. Each interaction block employs layer normalization and the SiLU activation function to introduce non-linearity and ensure stable training.[50] The model implements two parallel output pathways: one for energy prediction and another for forces. Both pathways share a similar architecture, starting with a 128-dimensional representation that is progressively reduced through multiple layers. Each pathway includes two batch normalization layers, ReLU activations, and a dropout rate of 0.1 for regularization.[45] The energy pathway concludes with a single output unit, while the forces pathway produces a two-dimensional output.

The training process employs a sophisticated optimization strategy with hierarchical learning rates implemented through the AdamW optimizer. The embedding layer uses a learning rate of 10^{-5}; interaction blocks operate at 2×10^{-5}; and both output pathways utilize 5×10^{-5}. A weight decay of 10^{-4} is applied across all parameters. The learning rate schedule combines a warm-up period lasting 20% of the total training steps with cosine decay thereafter. The model is trained for 100 epochs with a batch size of 2048 and employs gradient accumulation over 24 steps to stabilize training. A progressive loss weighting scheme is implemented, where force prediction training begins after 50 epochs and gradually increases in importance. The energy weight starts at 1.0 and decreases to 0.75, while the force weight increases from 0.0 to 0.5 over 50 epochs. Training stability is maintained through gradient clipping with a maximum norm of 0.01.

2.9. Model evaluation

The evaluation of model performance is conducted through a comprehensive loss function framework implemented across training, validation, and test datasets. Specifically, we employed mean absolute error (MAE) and mean squared error (MSE), defined as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N} | y_i - \widehat{y_i} | \tag{X}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \widehat{y_i}) \tag{XI}$$

where N is the number of data points, y_i represents the true values (actual labels), and $\widehat{y_i}$ represents the measured values, including both energies and forces. This tripartite evaluation strategy serves distinct purposes: the training loss guides the optimization process, the validation loss enables model selection and hyperparameter tuning, and the test loss provides an unbiased estimate of model generalization. This systematic computation of loss metrics across training, validation, and test datasets provides a robust mechanism for detecting potential overfitting phenomena, characterized by divergence between these performance metrics. This comprehensive monitoring strategy ensures that the model develops representations that generalize effectively to unseen data rather than merely memorizing training examples.

3. Results and discussion

3.1. Model evaluation

3.1.1. Model comparison

As presented in Table 1, KAN demonstrated exceptional performance and practical advantages. KAN achieved impressive accuracy with MAE values of 0.36, 0.30, and

Table 1. Performance metrics comparison of KAN against other typical ML models (SchNet, ANI, GNN, and CalHousNet) for MAE and MSE across training, validation, and test sets

Model	Evaluation	Training sets	Validation sets	Test sets
KAN	MAE	0.36	0.30	0.31
	MSE	0.79	0.50	0.49
SchNet	MAE	0.46	0.38	0.41
	MSE	0.88	0.55	0.59
ANI	MAE	0.79	0.77	0.79
	MSE	1.01	0.93	0.97
GNN	MAE	0.15	0.13	0.22
	MSE	0.11	0.04	0.50
CalHousNet	MAE	0.34	0.27	0.29
	MSE	0.76	0.34	0.51

Abbreviations: KAN: Kolmogorov-Arnold Network; ML: Machine learning; MAE: Mean absolute error; MSE: Mean squared error.

0.31 across training, validation, and test sets, respectively, alongside stable MSE metrics (0.79, 0.50, and 0.49). While the GNN demonstrated marginally superior numerical performance (MAE: 0.15, 0.13, and 0.22), it demands substantially more complex data preprocessing and feature engineering, significantly limiting its practical applicability. In contrast, KAN's straightforward implementation and minimal preprocessing requirements make it particularly valuable for real-world applications where computational efficiency and ease of deployment are crucial.

As featured in Figures S3-S7, the training dynamics of KAN display remarkable learning efficiency and stability. During the initial 30 epochs, KAN exhibits rapid learning characterized by a sharp decline in MAE, indicating swift comprehension of the underlying data patterns. The learning curve demonstrates consistent convergence, with both training and validation errors stabilizing after 50 epochs, suggesting robust generalization capabilities. This stability in later epochs, coupled with the close alignment between training and validation performance, indicates that KAN effectively avoids overfitting while maintaining strong predictive power.

Other benchmarked models displayed less competitive performance-to-complexity ratios. SchNet demonstrated moderate but inferior accuracy (MAE: 0.46, 0.38, and 0.41) while requiring specialized graph-based data structures. ANI exhibited the highest error metrics (MAE consistently around 0.79), suggesting limitations in its feature extraction capabilities. CalHousNet's performance (MAE: 0.34, 0.27, and 0.29) approached KAN's accuracy but lacked its efficient training dynamics and straightforward implementation. The consistent performance of KAN across diverse datasets, combined with its minimal preprocessing requirements and stable training behavior, establishes it as the most practical and efficient choice for large-scale molecular property prediction tasks.

In contrast, KAN provides a balanced approach, achieving good performance with less complex data preparation and lower computational demands, as demonstrated in Figure S8. Leveraging the principles of the Kolmogorov-Arnold representation theorem, KAN captures complex non-linear relationships efficiently, without the need for extensive preprocessing or high computational overhead. This efficient utilization of resources combined with robust predictive capabilities grants KAN a significant advantage in computational materials science, making it a preferred choice for researchers seeking to balance performance with practical operational demands. By leveraging KAN's predictive accuracy and using molecular statics, we compute properties such as elastic constants, thermal expansion coefficients, Poisson's ratio, bulk modulus, shear modulus, and Young's modulus, each essential for understanding the material's mechanical behavior.

To facilitate direct comparison in their natural units, we performed denormalization of the model predictions on the test dataset. Figure 5 presents the distribution of prediction errors in their original units, with energy errors presented in Ry (Figure 5A) and force errors in Ry/au (Figure 5B). After denormalization, the box plots reveal that KAN maintains excellent prediction accuracy with median energy errors around 0.1 Ry, while most predictions fall within a narrow range of 0.05 – 0.2 Ry, demonstrating robust performance. For force predictions, the median error is approximately 0.0003 Ry/au, with most predictions showing errors between 0.0001 – 0.0005 Ry/au. The presence of a few outliers (i.e., points above the whiskers) in both plots indicates occasional challenging cases, but these represent a small fraction of the predictions. This analysis in physical units further validates KAN's strong predictive capabilities, as it maintains high accuracy even when evaluated in the original, unnormalized scale of calculations.

3.1.2. Mechanical properties measurement

Various mechanical properties were measured, as follows:

(i) Elastic constants calculation: Elastic constants, including C_{11}, C_{12}, C_{13}, and C_{44}, were derived by applying the relevant strains to the crystal structure and measuring the resultant stress tensors with KAN. The strain tensor ϵ is applied to the structure's lattice:

$$R' = R(I + \epsilon) \tag{XII}$$

where R is the original lattice matrix, and I is the identity matrix. The relationship between stress (σ) and strain (ϵ) tensors is defined as:

$$\sigma = C\epsilon \tag{XIII}$$

The stress (σ) and strain (ϵ) data were linearly regressed to determine the material constants (C) from the slope of the stress-strain relationship.

(ii) Thermal expansion coefficient: The thermal expansion coefficient was determined by simulating temperature-induced lattice expansions. The expansion was modeled by scaling lattice vectors according to a temperature-dependent factor (α), and the resulting volumetric changes were used to compute α by fitting these changes over a range of temperatures. The lattice vectors are scaled based on a temperature-dependent expansion factor (α):

$$R' = R(1 + \alpha\Delta T) \tag{XIV}$$

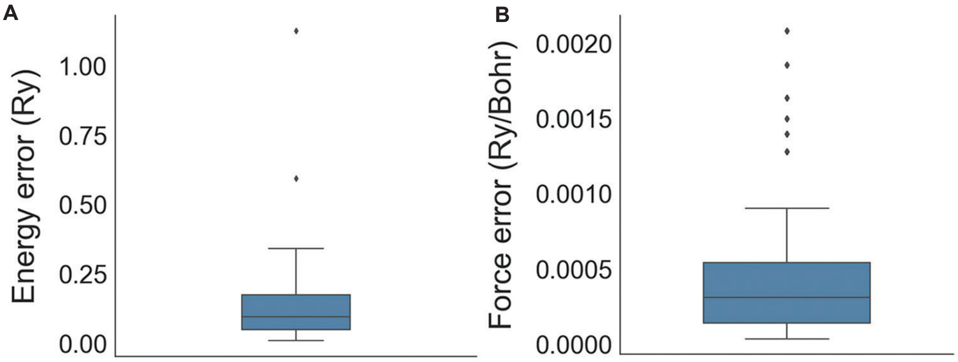


Figure 5. Performance evaluation of KAN through denormalized energy and force predictions on test dataset: box plot distribution of (A) energy errors (Ry) and (B) force errors (Ry/au), demonstrating high predictive accuracy in physical units.
Abbreviation: KAN: Kolmogorov-Arnold Network.

The change in volume (V) is computed for different temperatures and the thermal expansion coefficient (α) is determined by fitting the volume change:

$$\alpha = \frac{1}{V_O}\frac{\Delta V}{\Delta T} \tag{XV}$$

where V_0 is the initial volume.

(ii) Poisson's ratio: This ratio was calculated by applying uniaxial stress and using KAN to predict the resultant strain tensors. Stress (σ) is applied in one direction, and the resulting strain tensor (ϵ) is measured by KAN. Poisson's ratio (ν) is calculated as the negative ratio of transverse strain to longitudinal strain:

$$\upsilon = \frac{\epsilon_{\perp}}{\epsilon_{\|}} \tag{XVI}$$

(iii) Bulk modulus: By applying hydrostatic pressure and predicting the volumetric changes with KAN, the bulk modulus (K) was calculated from the relationship between pressure and volume change:

$$K = -V_0\frac{dP}{dV} \tag{XVII}$$

(iv) Shear modulus: Similarly, the shear modulus (G) was determined by applying shear stress and predicting the corresponding shear strain, with G calculated from the linear relationship between these two measurements.

$$G = \frac{\tau}{\gamma} \tag{XVIII}$$

(v) Young's modulus: Young's modulus (E) was computed by applying uniaxial stress, predicting the resultant longitudinal strain with KAN, and calculating E from the stress-strain relationship:

$$E = \frac{\sigma}{\epsilon} \tag{XIX}$$

Table 2 highlights the precision of KAN's predictions. Notably, the computed values for properties such as the elastic constants (C_{11}, C_{12}, C_{13}, and C_{44}) demonstrate minimal deviations from established reference data,[51] demonstrating the model's high accuracy. For example, C_{11} was measured at 5.737 GPa against a reference of 5.763 GPa, and C_{44} at 1.956 GPa against 1.84 GPa. In addition, the thermal expansion coefficient, Poisson's ratio, bulk modulus, shear modulus, and Young's modulus all closely align with their respective reference values, with discrepancies within 6%. These results underscore the robustness and reliability of KAN in accurately predicting and understanding material properties, thereby providing a comprehensive tool for materials science research and development.

3.2. MD simulations

To ensure the consistency and reliability of the predictions made by KAN, it is crucial to set the model to evaluation mode during the inference phase. This adjustment is necessary to stabilize the output by deactivating any layers that introduce stochastic behavior, such as Dropout and BatchNorm2d layers. By disabling these layers, we ensure that the model's predictions are stable and reproducible, free from the variability that these layers might otherwise introduce during training. For input data, KAN processes the interatomic distances as a tensor that spans from 1 to 10 Å. This range is intentionally broad to capture a variety of atomic interactions, from extremely close to relatively distant separations, thus providing a detailed dataset for modeling potential energy landscapes. To align with the model's architecture, the distance tensor is expanded to match the required input dimensionality of the KAN.

As displayed in Figure 6A, our calculations yield an equilibrium distance r_0 of 2.89 Å for Ru atoms. This result

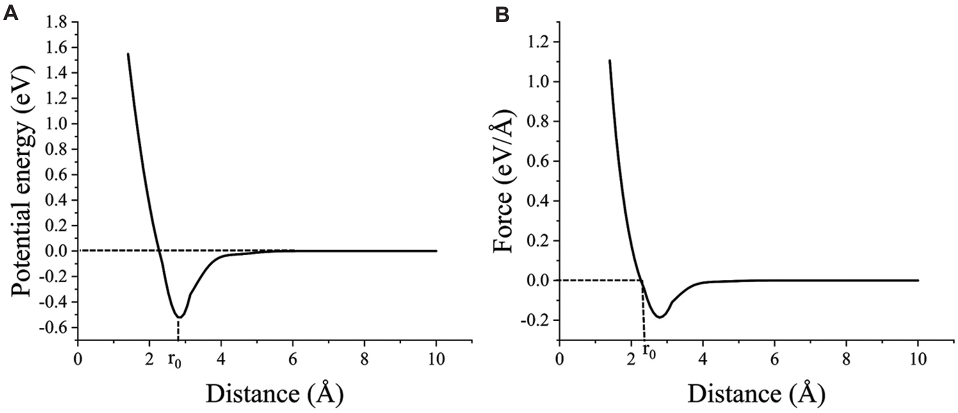


Figure 6. Interatomic potential energy and force curves for Ru atoms. (A) Potential energy curve showing the equilibrium distance, $r_0 \approx 2.89$ Å, at the turning point of the curve. (B) Corresponding force curve derived from the gradient of the potential energy, which crosses zero at r_0. Both curves demonstrate strong repulsion at short distances (left of r_0) and weaker attraction at longer distances (right of r_0).

Table 2. Comparison of the material properties directly calculated from the KAN model

Property	KAN (%)	Reference[51]
C_{11}	5.726 (0.6)	5.763
C_{12}	1.995 (5.8)	1.872
C_{13}	1.667 (0.3)	1.673
C_{44}	1.966 (6.0)	1.84
Thermal expansion coefficient (10^{-6} K^{-1})	6.44 (0.6)	6.4
Poisson's ratio	0.28 (6.0)	0.3
Bulk modulus (GPa)	217 (1.3)	220
Shear modulus (GPa)	167 (3.4)	173
Young's modulus (GPa)	435 (2.7)	447

Note: The KAN values agree with the reference values (Ref), where parentheses indicate the deviation as a percentage difference from the reference data.[51]
Abbreviation: KAN: Kolmogorov-Arnold Network.

closely aligns with the theoretically predicted nearest-neighbor distance of 2.71 Å in the hcp structure of Ru, differing by only about 6.6%.[38] This small discrepancy may be attributed to various factors, including the specifics of our computational model, potential stress or defects in real materials, and temperature effects.

The calculation of forces between atoms is another critical component of the model's functionality. Forces are derived by computing the gradient of the potential energy with respect to distance, as defined by the equation:

$$F=-\nabla E \qquad \text{(XX)}$$

where E represents the potential energy and F represents the forces acting between atoms. This calculation is performed by defining a function that outputs the energy for given distances and using automatic differentiation to compute the Jacobian matrix of these outputs. The Jacobian matrix provides the necessary gradients, and the net forces are then calculated by summing these derivatives across all relevant distances.

Figure 6B displays the corresponding force curve, which crosses zero at r_0, confirming the equilibrium position identified in the potential energy plot. To the left of r_0 (at shorter distances), a steep increase in potential energy is observed, corresponding to a strong repulsive force. This reflects the Coulomb repulsion between atomic nuclei at close proximity. To the right of r_0 (at longer distances), the potential energy rises more gradually, with the force becoming negative, indicating a weaker attractive force likely due to electron cloud overlap. These observations validate the consistency of our calculations with known material properties.

This meticulous computation of the Jacobian ensures that the forces derived from the model are accurate reflections of the underlying potential energy predictions. The final step in the workflow involves formatting the computed forces and energies into a LAMMPS potential file. This file, which includes potential energy values for a spectrum of interatomic distances, is compatible with LAMMPS, a widely used MD simulation software. By generating this file, we enable the integration of our model's outputs into broader MD simulations, facilitating detailed and practical analyses of material behaviors under various conditions.

3.2.1. Elastic constant

We used LAMMPS software to calculate the elastic constants using the interatomic potential generated by the KAN model. Elastic constants, such as C_{11}, C_{12}, C_{13}, and C_{44}, are important for describing the mechanical response of materials to various types of deformation and are integral for analyzing mechanical properties. We optimized the Ru crystal structure to its minimum energy configuration using the potential derived from the KAN model. We then subjected the structure to specific strain conditions – uniaxial, biaxial, and shear – to calculate the stress responses within Ru and the respective elastic constants. For a comparative analysis, we applied identical strain conditions using a well-established potential model known as the Finnis-Sinclair-type interatomic potential (FS potential).[52] This method is commonly used for simulating metallic systems and serves as a benchmark to validate the predictions made by the KAN interatomic potential.

Our results revealed that the elastic constants C_{11}, C_{12}, C_{13}, and C_{44} measured by our MD simulations with the KAN potential exhibited remarkable alignment with reference experimental values, displaying minimal deviations (Table 3). Our model's predictions were also comparable to those of the FS potential.

3.2.2. Thermodynamic properties

To validate the ability of the interatomic potential generated by KAN to predict thermodynamic properties, we conducted a detailed MD simulation using LAMMPS. The primary objective was to determine the melting point of Ru, leveraging the potentials derived from KAN. The MD simulation was initiated with Ru in its hcp lattice structure with fully periodic boundary conditions. The simulation utilized hybrid/overlay pair styles combining Lennard-Jones (LJ) and table potentials. This combination ensured a smooth transition to zero at larger distances, since the table potential only described atomic interactions in the range of 0.01 – 5 Å, and the LJ potential was incorporated to account for interactions beyond this range. The LJ parameters were carefully chosen to complement the table potential, ensuring continuous and physically meaningful interactions across all relevant distances. This hybrid approach allowed for precise modeling of short-range interactions using the detailed table potential, while also capturing the essential features of longer-range forces through the LJ potential, resulting in a more complete description of the system's behavior.

Ru atoms were assigned their appropriate masses, and the system was configured according to defined lattice parameters. An initial step of energy minimization brought the system to its minimum energy configuration, setting a stable baseline for subsequent dynamic processes. Following stabilization, the system was subjected to a controlled heating process under an NPT ensemble, allowing both temperature and volume to adjust naturally, mirroring real-world thermodynamic conditions. The temperature was incrementally increased from 300 K to over 4000 K. Throughout this process, we closely monitored the system's thermodynamic properties, such as potential energy and volume.

To obtain accurate relationships between the different phases, we performed common neighbor analysis[53] and calculated the radial distribution function (RDF)[54] on the lattice structure at varying temperatures during the simulation. As depicted in Figure 7, the simulation revealed a phase transition in Ru from its initial hcp structure to a face-centered cubic (fcc) configuration at approximately

Table 3. The elastic constants calculated from the different interatomic potentials in LAMMPS

Elastic constant (Mbar)	Table potential	FS potential[a]	Ref[b]
C_{11}	5.740	5.720	5.763
C_{12}	1.866	1.831	1.872
C_{13}	1.669	1.668	1.673
C_{44}	1.856	1.817	1.84

Note: The values display good agreement with the: [a]FS potential[52] and [b]reference values.[51]
Abbreviations: LAMMPS: Large-scale Atomic/Molecular Massively Parallel Simulator; FS: Finnis-Sinclair.

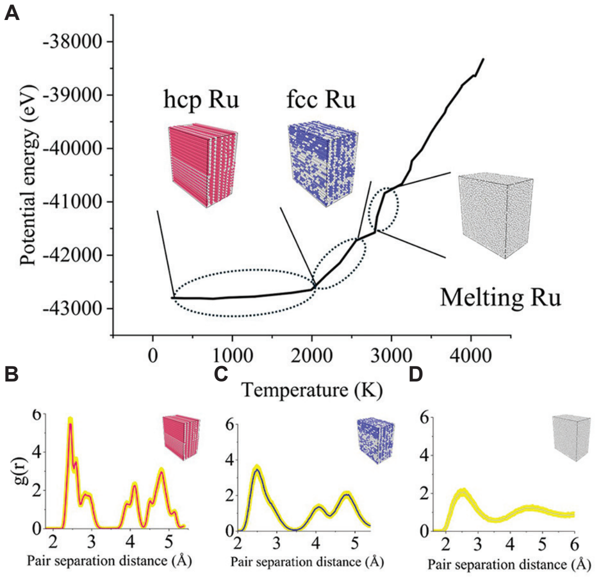


Figure 7. Phase transitions and potential energy evolution of Ru as a function of temperature. (A) Key phase changes (A) from hcp to fcc Ru at ~1882 K, followed by melting at ~2416 K; insets depict the atomic configurations for hcp, fcc, and liquid Ru. (B-D) The time-averaged radial distribution functions (RDF) for the hcp, fcc, and liquid phases, respectively. The well-defined RDF peaks in the (B) hcp and (C) fcc phases indicate short- and medium-range atomic order typical of crystalline structures, while the broadened peaks in (D) the liquid phase signify the loss of long-range order. Shaded regions around the RDF curves represent error bars, capturing statistical variability in atomic positions as the temperature rises.
Abbreviations: hcp: Hexagonal close-packed; fcc: Face-centered cubic.

1882 K; the proportion of the hcp phase also decreases from 100% initially to 0%. As displayed in Figure 7B, before 1882 K, Ru exhibits an hcp crystal structure.

The first significant peak appears at about 2.70 Å, representing the distance between the nearest neighboring atoms. The second and third peaks correspond to the distances of the second and third nearest neighbor atoms, respectively, between 4.3 – 4.4 and 5.1 – 5.4 Å. These peaks reflect the short- and medium-range orders of the hcp crystal, and the characteristic peaks are clear and relatively sharp. As observed in Figure 7C, around 1882 K, when the temperature rises, the crystal structure of hcp becomes unstable and gradually transforms, which is manifested by the obvious shift of the peak position of the characteristic peak in RDF. The second and third peaks gradually become wider and begin to shift, which makes the characteristic peaks relatively more symmetrical and regular, reflecting the high symmetry of atomic stacking in the fcc structure. This indicates that the regularity of atomic arrangement has changed during the transformation of hcp to fcc. This phase transition is consistent with the known behavior of Ru under heating and aligns with results from previous studies.[55]

Further heating led to a significant increase in potential energy at around 2416 K, which we identified as the melting point, marking the transition from the fcc solid phase to the liquid phase. As displayed in Figure 7D, the characteristic peaks in the RDF become broader and smoother, reflecting that the long-range order of the crystal structure completely disappears. There is only the first short-range order peak in the RDF, while the subsequent peaks gradually disappear, which is different from the crystal structure of the solid. The melting point observed during the simulation is strikingly close to the experimentally known true melting point of Ru, affirming the potential's accuracy.[56] The energy continued to rise beyond this temperature, consistent with the behavior of melting materials. Although a slight deviation is observed, this can be attributed to inherent limitations of the simulation model, such as quantum effects,[57] long-range electrostatic interactions,[58] and thermal and statistical fluctuations,[59] rather than a deficiency in KAN's potential.

4. Conclusion

This study demonstrates the efficacy of the KAN model in predicting the mechanical and thermodynamic properties of Ru with high accuracy and computational efficiency. The KAN model not only achieves comparable accuracy to prior literature but also offers significant advantages in terms of reduced computational requirements and descriptor complexity. Compared to traditional neural networks and GNN, our KAN-based model demonstrated superior performance in terms of learning speed, accuracy, and computational efficiency. While GNNs reported comparable accuracy, they required more complex data preparation and higher computational resources. KAN's ability to capture complex non-linear relationships efficiently, without extensive preprocessing or high computational overhead, makes it particularly suitable for computational materials science applications, particularly in the exploration of RSAs. Our model's predictions of elastic constants, thermal expansion coefficient, Poisson's ratio, bulk modulus, shear modulus, and Young's modulus displayed excellent agreement with experimental reference data, with discrepancies within 6%. This high level of accuracy validates the reliability of our KAN-based approach. Furthermore, the integration of KAN models with MD simulations accurately captured Ru's phase transitions, including the transition from hcp to fcc structure and melting point.

However, it is important to acknowledge the limitations of our approach. The slight deviation observed in the melting point prediction, while small, highlights the inherent challenges in accurately modeling complex material behaviors. This deviation may be attributed to several factors, including quantum effects not accounted for in classical MD simulations, incomplete representation of long-range electrostatic interactions, and thermal and statistical fluctuations inherent in MD simulations. In addition, while our model performs well for Ru, its applicability to a wider range of elements and compounds warrants further investigation. The model's accuracy may vary for materials with significantly different electronic structures or bonding characteristics. Although more efficient than some alternatives, the KAN approach still requires substantial computational resources for training and large-scale simulations, which may limit its applicability in certain research settings.

Despite these limitations, the success of KAN in characterizing Ru properties has significant implications for the broader field of computational materials science. By providing a method that balances high-fidelity predictions with practical computational demands, this approach could accelerate the discovery and design of novel RSAs and other advanced materials. KAN's ability to generate accurate interatomic potentials for MD simulations further enhances its utility in predicting complex material behaviors under various conditions. Looking ahead, the KAN framework demonstrates promise for extension to other elements and more complex alloy systems. For transferring this model to other materials, the approach differs depending on the system's complexity. For single elements, we can leverage the existing successful KAN structure that was developed for Ru. This transfer simply requires inputting different DFT-calculated datasets for the new target element while maintaining the same architectural framework. However, when considering binary or higher-order materials systems, the KAN structure needs to be appropriately adjusted to accommodate more complex crystal structures and atomic interactions. This may involve modifying the input layer and potentially adjusting the network's depth or width to capture the additional complexity of multi-element systems. In addition, integrating KAN with high-throughput screening methods could further expedite the materials discovery process. Efforts to incorporate quantum mechanical effects and improve the handling of long-range interactions could enhance the model's accuracy, especially for extreme conditions and diverse material types. This research not only advances our understanding of Ru but also provides a valuable tool for the broader materials science community. The successful implementation of KAN in predicting Ru properties represents a significant advancement in materials informatics, establishing a more efficient and accurate pathway for material property prediction that could revolutionize the discovery and design of complex systems like HEAs and refractory superalloys while setting a new standard for ML applications in materials research.

Acknowledgments

This work used SDSC Expanse at the San Diego Supercomputer Center through allocation BIO240093 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services and Support program supported by National Science Foundation (NSF) grants #2138259, #2138286, #2138307, #2137603, and #2138296.

Funding

J.Y. acknowledges support from the US NSF under the award CMMI-2338518.

Conflict of interest

Jingjie Yeo is an Editorial Board Member of this journal but was not in any way involved in the editorial and peer-review process conducted for this paper, directly or indirectly. Separately, other authors declared that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Author contributions

Conceptualization: All authors
Data curation: Zhiyu An
Formal analysis: Zhiyu An
Funding acquisition: Jingjie Yeo
Investigation: Zhiyu An
Methodology: Zhiyu An
Project administration: Jingjie Yeo
Software: Zhiyu An
Resources: Jingjie Yeo
Supervision: Jingjie Yeo
Validation: Jingjie Yeo
Visualization: Zhiyu An
Writing – original draft: Zhiyu An
Writing – review & editing: Jingjie Yeo

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

The code and dataset for our implementation of the KAN model can be found at https://github.com/ZhiyuAn-byte/KAN-Model-for-Ru-element (doi: 10.5281/zenodo.13856262)

References

1. Naka S, Khan T. Designing novel multiconstituent intermetallics: Contribution of modern alloy theory in developing engineered materials. *J Phase Equilibria*. 1997;18(6):635-649.

   doi: 10.1007/BF02665823.

2. Yeh JW, Chen SK, Lin SJ, *et al*. Nanostructured high-entropy alloys with multiple principal elements: Novel alloy design concepts and outcomes. *Adv Eng Mater.* 2004;6(5):299-303.

   doi: 10.1002/adem.200300567

3. Cantor B, Chang ITH, Knight P, Vincent AJB. Microstructural development in equiatomic multicomponent alloys. *Mater Sci Eng A*. 2004;375-377:213-218.

   doi: 10.1016/j.msea.2003.10.257

4. Miracle DB, Senkov ON. A critical review of high entropy alloys and related concepts. *Acta Mater.* 2017;122:448-511.

   doi: 10.1016/j.actamat.2016.08.081

5. Soni V, Gwalani B, Senkov O, *et al*. Phase stability as a function of temperature in a refractory high-entropy alloy. *J Mater Res.* 2018;33:1-12.

   doi: 10.1557/jmr.2018.223

6. Senkov ON, Crossman B, Rao SI, *et al*. Mechanical properties of an Al10Nb20Ta15Ti30V5Zr20 A2/B2 refractory superalloy and its constituent phases. *Acta Mater.*

2023;254:119017.

doi: 10.1016/j.actamat.2023.119017

7. Miracle DB, Tsai MH, Senkov ON, Soni V, Banerjee R. Refractory high entropy superalloys (RSAs). *Scr Mater.* 2020;187:445-452.

   doi: 10.1016/j.scriptamat.2020.06.048

8. Prins SN, Cornish LA, Stumpf WE, Sundman B. Thermodynamic assessment of the alru system. *Calphad*. 2003;27(1):79-90.

   doi: 10.1016/S0364-5916(03)00033-6

9. Haynes WM, editor. *CRC Handbook of Chemistry and Physics*. 95th ed. Boca Raton, FL: CRC Press/Taylor and Francis; 2015.

10. Hulm JK, Goodman BB. Superconducting properties of rhenium, ruthenium, and osmium. *Phys Rev.* 1957;106(4):659-671.

    doi: 10.1103/PhysRev.106.659

11. Anzellini S, Errandonea D, Cazorla C, *et al*. Thermal equation of state of ruthenium characterized by resistively heated diamond anvil cell. *Sci Rep.* 2019;9(1):14459.

    doi: 10.1038/s41598-019-51037-8

12. Güler E, Uğur Ş, Güler M, Uğur G. Molecular dynamics exploration of the temperature-dependent elastic, mechanical, and anisotropic properties of Hcp ruthenium. *Eur Phys J Plus*. 2024;139(5):372.

    doi: 10.1140/epjp/s13360-024-05177-0

13. Zhi-Peng LU, Wen-Jun Z, Tie-Cheng L, Chuan-Min M, Liang X, Xu-Hai L. Structural phase transition of ru at high pressure and temperature. *Acta Phys Sin.* 2013;62(17):176402-176402.

    doi: 10.7498/aps.62.176402

14. Jain A, Ong SP, Hautier G, *et al*. Commentary: The materials project: A materials genome approach to accelerating materials innovation. *APL Mater.* 2013;1(1):011002.

    doi: 10.1063/1.4812323

15. Mobarak MH, Mimona MA, Islam MA, *et al*. Scope of machine learning in materials research-a review. *Appl Surf Sci Adv.* 2023;18:100523.

    doi: 10.1016/j.apsadv.2023.100523

16. Mannodi-Kanakkithodi A, Pilania G, Huan TD, Lookman T, Ramprasad R. Machine learning strategy for accelerated design of polymer dielectrics. *Sci Rep.* 2016;6:20952.

    doi: 10.1038/srep20952

17. Stanev V, Oses C, Kusne AG, *et al*. Machine learning modeling of superconducting critical temperature. *NPJ Comput Mater.* 2018;4(1):29.

    doi: 10.1038/s41524-018-0085-8

18. Faber FA, Lindmaa A, Von Lilienfeld O.A, Armiento, R. Machine learning energies of 2 million elpasolite $(AB{C}_{2}{D}_{6})$ crystals. *Phys Rev Lett.* 2016;117(13):135502.

    doi: 10.1103/PhysRevLett.117.135502

19. Pilania G, Balachandran PV, Kim C, Lookman T. Finding new perovskite halides via machine learning. *Front Mater.* 2016;3:19.

    doi: 10.3389/fmats.2016.00019

20. Malkiel I, Mrejen M, Nagler A, Arieli U, Wolf L, Suchowski H. Plasmonic nanostructure design and characterization via deep learning. *Light Sci Appl.* 2018;7(1):60.

    doi: 10.1038/s41377-018-0060-7

21. De Jong M, Chen W, Notestine R, *et al*. A statistical learning framework for materials science: Application to elastic moduli of k-nary inorganic polycrystalline compounds. *Sci Rep.* 2016;6(1):34256.

    doi: 10.1038/srep34256

22. Song K, Zhao R, Liu J, *et al*. General-purpose machine-learned potential for 16 elemental metals and their alloys. *Nat Commun.* 2024;15(1):10208.

    doi: 10.1038/s41467-024-54554-x

23. Liyanage M, Turlo V, Curtin WA. Machine learning potential for the Cu-W system. *Phys Rev Mater.* 2024;8(11):113804.

    doi: 10.1103/physrevmaterials.8.113804

24. Liu Y, Zhao T, Ju W, Shi S. Materials discovery and design using machine learning. *High Throughput Exp Model Res Adv Batter.* 2017;3(3):159-177.

    doi: 10.1016/j.jmat.2017.08.002

25. Anstine DM, Isayev O. Machine learning interatomic potentials and long-range physics. *J Phys Chem A.* 2023;127(11):2417-2431.

    doi: 10.1021/acs.jpca.2c06778

26. Kolmogorov A. On the representation of continuous functions of several variables as superpositions of continuous functions of a smaller number of variables. *Dokl Akad Nauk*. 1956;108(2):25-46.

27. Liu Z, Wang Y, Vaidya S, *et al. KAN: Kolmogorov-Arnold Networks*; 2024. Available from: https://arxiv.org/abs/2404.19756 [Last accessed on 2025 Jan 26].

28. Anderson PW. Absence of diffusion in certain random lattices. *Phys Rev.* 1958;109(5):1492-1505.

    doi: 10.1103/PhysRev.109.1492

29. Singer SJ, Nicolson GL. The fluid mosaic model of the structure of cell membranes. *Science*. 1972;175(4023):720-731.

    doi: 10.1126/science.175.4023.720

30. Li Y, Zhang JH, Mei F, Ma J, Xiao L, Jia S. Generalized Aubry-André-Harper models in optical superlattices. *Chin*

Phys Lett. 2022;39(6):063701.

doi: 10.1088/0256-307x/39/6/063701

31. Giannozzi P, Andreussi O, Brumme T, *et al.* Advanced capabilities for materials modelling with quantum ESPRESSO. *J Phys Condens Matter*. 2017;29(46):465901.

    doi: 10.1088/1361-648X/aa8f79

32. Giannozzi P, Baroni S, Bonini N, *et al*. QUANTUM ESPRESSO: A modular and open-source software project for quantum simulations of materials. *J Phys Condens Matter*. 2009;21(39):395502.

    doi: 10.1088/0953-8984/21/39/395502

33. Perdew JP, Burke K, Ernzerhof M. Generalized gradient approximation made simple. *Phys Rev Lett.* 1996;77(18):3865-3868.

    doi: 10.1103/PhysRevLett.77.3865

34. Blöchl PE. Projector augmented-wave method. *Phys Rev B*. 1994;50(24):17953-17979.

    doi: 10.1103/PhysRevB.50.17953

35. Ong SP, Richards WD, Jain A, *et al*. Python materials genomics (Pymatgen): A robust, open-source python library for materials analysis. *Comput Mater Sci.* 2013;68:314-319.

    doi: 10.1016/j.commatsci.2012.10.028

36. Chen X, Wang LF, Gao XY, *et al.* Machine learning enhanced empirical potentials for metals and alloys. *Comput Phys Commun.* 2021;269:108132.

    doi: 10.1016/j.cpc.2021.108132

37. Thompson AP, Aktulga HM, Berger R, *et al.* LAMMPS-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput Phys Commun.* 2022;271:108171.

    doi: 10.1016/j.cpc.2021.108171

38. Villars P, Cenzual K. *Ru Hcp (Ru) Crystal Structure: Datasheet from "PAULING FILE Multinaries Edition*. Springer Materials; 2022. Available from: https://materials.springer.com/isp/crystallographic/docs/sd_1244358 [Last accessed on 2025 Jan 26].

39. De Jong M, Chen W, Angsten T, *et al*. Charting the complete elastic properties of inorganic crystalline compounds. *Sci Data*. 2015;2(1):150009.

    doi: 10.1038/sdata.2015.9

40. Himanen L, Jäger MOJ, Morooka EV, *et al.* DScribe: Library of descriptors for machine learning in materials science. *Comput Phys Commun.* 2020;247:106949.

    doi: 10.1016/j.cpc.2019.106949

41. D'Agostino D, Serani A, Campana EF, Diez M. Nonlinear methods for design-space dimensionality reduction in shape optimization. In: Nicosia G, Pardalos P, Giuffrida G, Umeton R, editors. *Machine Learning, Optimization, and Big Data*. Cham: Springer International Publishing; 2018. p. 121-132.

42. Kingma DP, Jimmy B. Adam: A Method for Stochastic Optimization. In: 3rd *International Conference for Learning Representationsy*, CoRR; 2015

43. Cortes C, Mohri M, Rostamizadeh A. L2 Regularization for Learning Kernels. In: *Twenty-Fifth Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*; 2004. p. 109-16.

44. Paszke A, Gross S, Massa F, *et al*. Pytorch: An imperative style, high-performance deep learning library. *Adv Neural Inf Process Syst.* 2019;32:8024-8035.

45. Agarap AF. *Deep Learning Using Rectified Linear Units (Relu)*. [Preprint]; 2018.

46. Smith JS, Isayev O, Roitberg AE. ANI-1: An extensible neural network potential with DFT accuracy at force field computational cost. *Chem Sci.* 2017;8(4):3192-3203.

    doi: 10.1039/C6SC05720A

47. Najm HN, Yang Y. *AEVmod-Atomic Environment Vector Module Documentation*; United States. California: Sandia National Laboratories; 2021.

    doi: 10.2172/1817835.

48. Zhang G, Haopeng, L. *Effectiveness of Scaled Exponentially-Regularized Linear Units (SERLUs)* [Preprint]; 2018.

49. Schütt KT, Sauced HE, Kindermans PJ, Tkatchenko A, Müller KR. SchNet-a deep learning architecture for molecules and materials. *J Chem Phys.* 2018;148(24):241722.

    doi: 10.1063/1.5019779

50. Elfwing S, Uchibe E, Doya K. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Netw.* 2018;107:3-11.

    doi: 10.1016/j.neunet.2017.12.012

51. Fast L, Wills JM, Johansson B, Eriksson O. Elastic constants of hexagonal transition metals: Theory. *Phys Rev B*. 1995;51(24):17431-17438.

    doi: 10.1103/PhysRevB.51.17431

52. Fortini A, Mendelev MI, Buldyrev S, Srolovitz D. Asperity contacts at the nanoscale: Comparison of Ru and Au. *J Appl Phys.* 2008;104(7):074320.

    doi: 10.1063/1.2991301

53. Honeycutt JD, Andersen HC. Molecular dynamics study of melting and freezing of small lennard-jones clusters. *J Phys Chem.* 1987;91(19):4950-4963.

    doi: 10.1021/j100303a014

54. Abbaspour M, Jorabchi MN, Akbarzadeh H, Ebrahimnejad A. Investigation of the thermal properties of phase change materials encapsulated in capped carbon nanotubes

using molecular dynamics simulations. *RSC Adv.* 2021;11(40):24594-24606.

doi: 10.1039/D1RA02033A

55. Lu ZP, Zhu WJ, Lu TC, *et al*. Structural phase transition of Ru at high pressure and temperature. *Wuli Xuebao Acta Phys Sin.* 2013;62(17):176402.

    doi: 10.7498/aps.62.176402

56. Savitskii EM. *Noble Metals: A Handbook*; Izdatel'stvo Metallurgiia. Moscow: USSR; 1984.

57. Dammak H, Hayoun M, Brieuc F, Geneste G. Nuclear quantum effects in molecular dynamics simulations. *J Phys Conf Ser*. 2018;1136:012014.

58. Piana S, Lindorff-Larsen K, Dirks RM, Salmon JK, Dror RO, Shaw DE. Evaluating the effects of cutoffs and treatment of long-range electrostatics in protein folding simulations. *PLoS One*. 2012;7(6):e39918.

    doi: 10.1371/journal.pone.0039918

59. Hoyt JJ, Trautt ZT, Upmanyu M. Fluctuations in molecular dynamics simulations. *Multiscale Model Mov Interfaces Mater.* 2010;80(7):1382-1392.

    doi: 10.1016/j.matcom.2009.03.012

ACCSCIENCE
PUBLISHING

International Journal of AI for Materials and Design

ORIGINAL RESEARCH ARTICLE

Machine learning-based model predictive control for multizone building automation: A case study

Pradeep Shakya[1], Shiva Sreenivasan[2], Baskaran Krishnamoorthy[1], Shiyu Yang[3], and Man Pun Wan[2]*

[1]Sustainable Built Environments, Energy Research Institute @ Nanyang Technological University (ERI@N), 1 Cleantech Loop, Singapore

[2]Thermal and Fluids Division, School of Mechanical and Aerospace Engineering (MAE), Nanyang Technological University (NTU), 50 Nanyang Avenue, Singapore

[3]Department of Fluid Dynamics, Institute of High Performance Computing (IHPC), Agency for Science, Technology, and Research (A*STAR), 1 Fusionopolis Way, Singapore

Abstract

In Singapore's hot and humid climate, air-conditioning and mechanical ventilation (ACMV) systems account for over 60% of commercial building energy consumption, driving efforts to enhance energy efficiency through predictive control strategies such as model predictive control (MPC) to overcome the limitations of conventional reactive building automation systems. This paper presents a multizone MPC system designed to optimize energy consumption and thermal comfort in a commercial building's ACMV system in Singapore. The system was implemented in a multi-use test building with real occupancy and a deployment area of approximately 850 m^2, partitioned into six learning zones, two office spaces, and three open spaces. The ACMV system serving the deployment area consisted of two primary air-handling units and 16 fan coil units, where chilled water was supplied to the cooling coils, and conditioned air was distributed through motorized diffusers. To facilitate predictive control, data-driven thermal prediction models were developed for each zone using a non-linear autoregressive exogenous network with exogenous inputs trained on historical data and disturbances. Thermal comfort optimization was guided by the predictive mean vote, which was targeted at 0, representing thermal neutrality (as per ASHRAE 55 standards), and constrained within a range of −0.5 – 0.5. Performance comparisons demonstrated that the MPC system achieved over 42% energy savings compared to the original thermostat-based control while enhancing thermal comfort. Despite its advantageous control performances, challenges for large-scale deployment remain, including implementation costs, scalability, and model accuracy. Future work can address these challenges by developing comfort models that leverage existing building sensors.

Keywords: Model predictive control; Coordinated multisystem control; Energy saving; Thermal comfort; Visual comfort; High-performance building

***Corresponding author:**
Man Pun Wan
(mpwan@ntu.edu.sg)

Citation: Shakya P, Sreenivasan S, Krishnamoorthy B, Yang S, Wan MP. Machine learning-based model predictive control for multizone building automation: A case study. *Int J AI Mater Design*. 2025;2(1):39-53. doi: 10.36922/ijamd.8161

Received: December 24, 2024

Revised: February 7, 2025

Accepted: February 20, 2025

1. Introduction

In Singapore, owing to its hot and humid climate throughout the year, commercial buildings typically spend more than 60% of their electricity consumption on air conditioning, mechanical, and ventilation (ACMV) systems.[1] On March 31, 2020, Singapore submitted its Long-Term Low Emissions Development Strategy to the United Nations Framework Convention on Climate Change, pledging to halve its emissions from their peak to 33 MTCO$_2$e and to achieve net-zero emissions by 2050.[2] At present, the Building Construction Authority (BCA) has reported that high-performance buildings in Singapore have achieved more than 65% improvement in energy efficiency compared to 2005 levels. The BCA further aims to raise the energy efficiency improvement of Singapore's buildings to 80% relative to 2005 levels by 2030 through its Green Building Innovation Cluster program.[3]

The conventional approach in current building automation and control systems is predominantly "reactive," generating control signals based on deviations of previously measured information from a control setpoint (thermostat-based). Consequently, due to the different thermal characteristics of buildings and the non-linear operation of their ACMV systems, reactive control strategies are unable to achieve optimal efficiency and occupant comfort. These strategies only respond to transient disturbances – changes in occupancy, internal heat load profiles, and external weather conditions – as well as to building dynamics after the disturbances occur. This limitation highlights the necessity for predictive control strategies that can anticipate and mitigate such disturbances before they adversely impact building performance.

Studies on predictive control solutions for optimizing ACMV operation in recent decades can broadly be categorized based on model architecture into (i) model predictive control (MPC) and (ii) reinforced learning (RL).[4] MPC uses prediction models to forecast future system behavior and proactively adjust control actions, making it particularly well-suited for complex systems with non-linear dynamics. Within the MPC framework, three primary techniques are used to develop prediction models: physics-based (white box), data-driven (black box), and hybrid (gray box) methods.[5] Physics-based models rely on fundamental principles such as mass and energy conservation and can be implemented using software tools such as EnergyPlus,[6] TRNSYS,[7] eQuest,[8] and Modelica.[9] However, these models require significant expertise, resulting in higher implementation costs, particularly in complex real-world buildings. In contrast, data-driven models apply regression techniques to large datasets from real buildings or simulations, necessitating extensive data collection. Hybrid models combine elements of both approaches, estimating certain physical parameters through data-driven methods.

Data-driven MPC strategies have demonstrated considerable promise in achieving both optimal thermal comfort and high energy efficiency. Over the years, researchers such as Yang *et al.*,[10] Yang and Wan[11] have enhanced MPC by simplifying building models through linearization techniques. Široký *et al.*[12] reported energy savings of 15 – 30% in a university building by integrating MPC with weather forecasting capabilities, whereas Ma *et al.*[13] achieved a 19% efficiency improvement using a hierarchical MPC system for a chiller plant. In experimental studies, Pang *et al.*[14] demonstrated that MPC reduced chilled water consumption by 42% in a radiant slab system compared to heuristic methods. Despite these promising results, two primary obstacles hinder the widespread adoption of MPC: the need for highly accurate predictive models and the significant computational cost associated with solving complex optimization problems across multiple zones. To address these issues, Yang *et al.*[15] proposed a two-level distributed computation scheme that optimizes thermal comfort and internal air quality across multiple zones using MPC. In addition, an event-triggered optimization mechanism has been shown to reduce the computational load by 12–22% while achieving over 9% energy savings,[16] further highlighting MPC's potential in real-world applications.

Alongside MPC, recent advancements in machine learning (ML) have introduced powerful tools for predicting building energy performance. A review by Fathi *et al.*[17] examined 14 ML methods relevant to this domain. Among these, support vector machines (SVM) are particularly effective for regression tasks involving non-linear relationships, especially when using the radial basis function kernel, which captures non-linear interactions more effectively compared to sigmoid and polynomial kernels.[18,19] Random forest (RF), an ensemble learning method based on decision trees, is adept at managing non-linear interactions and offers improved prediction stability, albeit at the expense of increased computational time.[20] In the realm of artificial neural networks (ANN), the non-linear autoregressive network with exogenous inputs (NARX) is well-suited for dynamic systems, leveraging both historical and external inputs to forecast future states.[21] In addition, long short-term memory (LSTM) networks, a type of recurrent neural network known for capturing long-term dependencies, are particularly effective for modeling the time-series dynamics of building energy consumption.[22] Integrating these ML-based forecasting methods with MPC can further enhance the predictive capabilities and overall performance of building automation systems.

Despite the promise of MPC, challenges remain in accurately modeling the complex thermal dynamics of buildings, which are influenced by varying weather conditions, occupancy patterns, and building envelope characteristics. In addition, the computational burden associated with optimization across multiple zones presents further difficulties.[23] To address these challenges, model-free approaches based on RL (such as Q-learning[24] and deep reinforcement learning[25]) have been explored, with several studies demonstrating notable energy savings. For instance, Ding *et al.*[26] reported approximately 14% energy savings using deep RL with inputs from EnergyPlus simulations. However, the present study focuses on MPC due to its structured approach and its ability to directly integrate system constraints into the optimization process.

In this paper, we present a multizone MPC framework for the coordinated control of ACMV systems, aiming to optimize both occupant comfort and energy efficiency. To evaluate performance, we compare the energy consumption and thermal comfort achieved using our coordinated MPC approach with those obtained from a conventional reactive control strategy employing proportional-integral-derivative (PID) control. The key novelties of our MPC system are as follows:

(i) Large-scale, multizone implementation: This study demonstrates a real-world deployment of MPC in an 11-zone building (850 m^2) with actual occupancy. This contrasts with previous studies, which were limited to smaller-scale implementations (e.g., two zones[27]), simulated environments,[28] or unoccupied buildings[29]

(ii) Integration of ML-based weather forecasting: By incorporating ML-based weather forecasting into the MPC framework, our system proactively adjusts control actions in anticipation of weather changes. This represents an advancement over previous studies, which typically relied on existing or historical weather data.[30-32]

This work provides valuable insights into both the challenges and benefits of implementing MPC in complex, real-world settings, particularly within tropical environments. Consequently, it contributes significantly to the advancement of scalable solutions for building energy management.

This paper is organized as follows:

- Section 1 (Introduction): Provides background, motivation, and objectives of the study
- Section 2 (Testbed setup): Describes the heating, ventilation, and air conditioning (HVAC) system and sensor configuration of the testbed
- Section 3 (MPC controller for ACMV): Details the design and implementation of the MPC controller for the ACMV system
- Section 4 (ML model development for MPC): Outlines the development of the ML models used for forecasting and their integration within the MPC framework
- Section 5 (Results and discussion): Presents the performance evaluation, key findings, and a comparative analysis with conventional control strategies and discusses the challenges encountered
- Section 6 (Conclusion): Summarizes the study's contributions, implications, and suggestions for future research.

2. Testbed setup

For the testbed, a commercial building located in Jurong East, Singapore, was selected. The building is served by a central chiller plant consisting of two chillers, each with a capacity of 438 refrigeration tons. The ACMV system in the building operates from 7 AM to 7 PM on weekdays. The experiments were conducted in an 850 m^2 multiuse test space, partitioned into 11 zones, including two office spaces (OS1 – 2), six learning zones (LZ1 – 6), and three common areas (CA1 – 3), as depicted in Figure 1. MPC was implemented to coordinate and control the ACMV systems, as illustrated in Figure 2. The test space includes operable partition walls between LZ1 and LZ2, as well as among LZ4, LZ5, and LZ6. These partitioning walls can be adjusted so that two zones (LZ1 and LZ2) or up to three zones (LZ4, LZ5, and LZ6) can be merged into a single zone. The open/closed state of these walls is determined by proximity sensors, which output "1" if the walls are closed (i.e., zones remain separate, as in LZ1 and LZ2) or "0" if the walls are open (zones merge into one larger room).

The schematic diagram of the ACMV system installed in the test space is shown in Figure 2. The ACMV system serving the test space consists of two primary air-handling units (PAUs) for pre-conditioned fresh air supply and 16 fan coil units (FCUs) for cooling and dehumidification. The cooling coils in the PAUs and FCUs are supplied with chilled water from the building's central chiller plant (not shown in Figure 2). Since the MPC system of this study controls only the airside equipment of the ACMV system within the test space – a fraction of the entire building – the net effect of MPC on the overall chiller plant is not considered. Instead, cooling energy consumption, measured by British thermal unit (BTU) meters (or energy meters) installed at the cooling coils of the PAUs and FCUs, is used to evaluate the cooling energy consumed by the ACMV system in the test space. Each PAU is equipped with an on/off damper, a cooling coil, and a supply fan. The PAUs

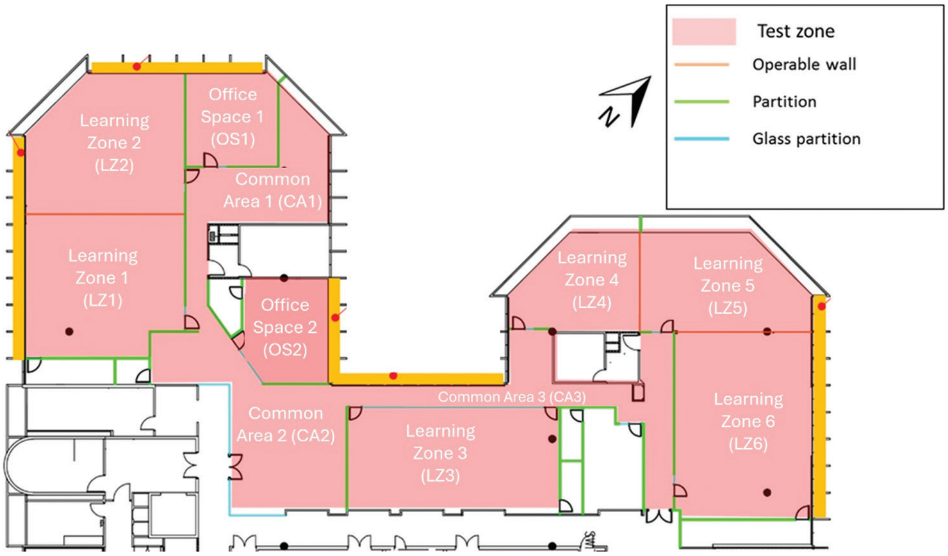


Figure 1. Layout of spaces within the test zone

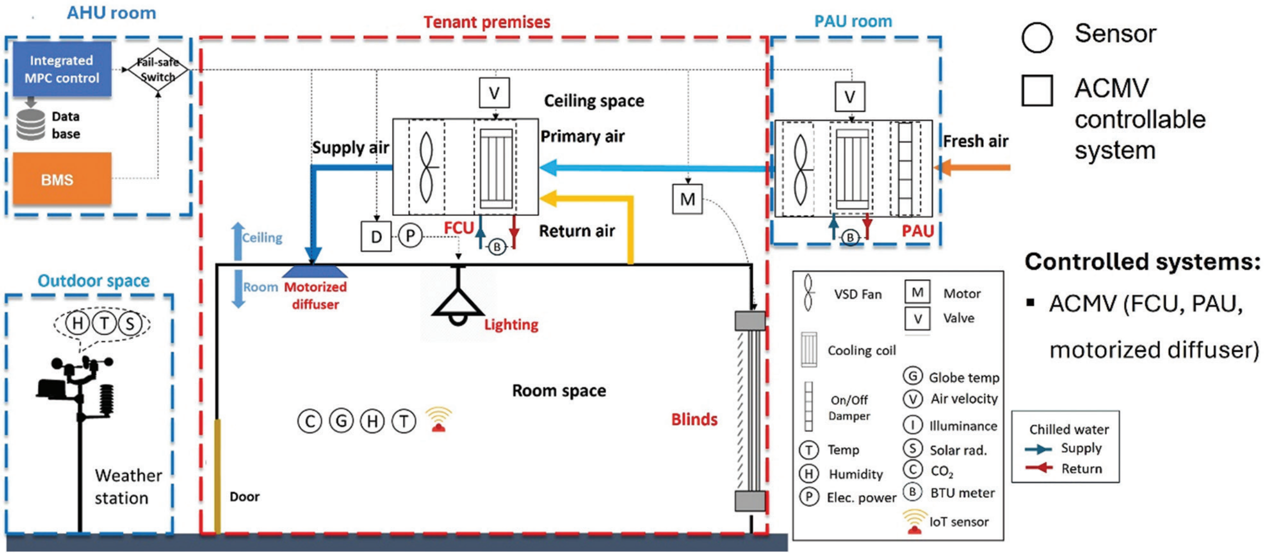


Figure 2. Sensors, actuators, and other hardware installed in the test space for model predictive control
Abbreviations: ACMV: Air conditioning and mechanical ventilation; AHU: Air handling unit; BMS: Building management system; BTU: British thermal unit; FCU: Fan coil unit; IoT: Internet of Things; PAU: Primary air-handling unit; VSD: Variable speed drive.

operate at a constant off-coil temperature setpoint of 24°C and a constant supply airflow rate of 2.2 m^3/s. The 16 FCUs provide terminal air conditioning to the test space, each equipped with a cooling coil and a supply fan. The FCUs draw in a mixture of fresh air from the PAUs and return air from the test zones for cooling and dehumidification, and then supply the conditioned air back to the test zones through motorized air diffusers. Each FCU operates at a constant supply fan speed, with chilled water flowing through its cooling coil regulated by a motorized water valve. The cooling energy consumed by each FCU is determined by converting thermal energy measured using BTU meters to electrical energy consumption. Similarly, the electrical energy consumed by each FCU's fan is measured using an electrical power meter installed at the fan. The total cooling energy consumed by the ACMV system is calculated as the sum of cooling energy consumed by each FCU and electrical energy consumed by their fans.

When the test space is controlled by the existing building management system (referred to as "BMS" in this study), cooling power delivered to the test space is regulated by controlling the chilled water flow rate through each FCU's cooling coil through a motorized water valve using a PID controller. The control is based on a temperature setpoint of 21°C, measured by a thermostat located at the diffuser outlet.

Each zone is equipped with a set of sensors, including globe temperature, ambient temperature, and humidity sensors for measuring thermal comfort, as well as CO_2 sensors for measuring occupancy within each zone. For the evaluation of the predicted mean vote (PMV) as per ASHRAE 55,[33] ambient air velocity, metabolic rate, and clothing insulation factor for each occupant were assumed constant at 0.1 m/s, 1.2 met, and 0.5 CLO, respectively. Duct air temperature, humidity, and flow sensors were installed at both the supply and return sides of the PAUs and FCUs to enable precise measurement of the cooling capacity of supply air for each zone. In addition, duct CO_2 sensors were installed on both the supply and return sides of FCUs. By integrating data from the duct CO_2 and flow sensors in the FCUs and PAUs with the indoor CO_2 sensors in each zone, the occupancy of each zone was determined.

A weather station, equipped with outdoor temperature, humidity, and solar radiation sensors, was installed on the rooftop of the building. The specifications of these sensors, installed in the test space, are summarized in Table 1.

3. MPC controller for ACMV

The MPC controller comprises an ML-based building dynamics predictive model and an optimization solver. The ML model for predicting indoor PMV is developed using a NARX feedback neural network, a recurrent neural network commonly used for time-series modeling. A detailed description of the development of the ML network for MPC is provided in Section 4. For the MPC controller, the cooling power supplied by the FCU cooling coil serves as the manipulated variable and is the control signal to be optimized. The current cooling power is measured as feedback using the BTU meter installed in each FCU within the zone. The number of occupants, representing the internal heat load, is measured using the duct CO_2 and flow sensors at the FCU and PAU, along with the indoor CO_2 sensors in each zone. Outdoor temperature and solar radiation data are obtained from the rooftop weather station. The solar heat gain from the windows (Q_{win}) is derived as a function of solar radiation, as shown in Equation I. This equation accounts for heat gain from both the shaded (by blinds) and unshaded regions of the window. Here, *SR* represents the ratio of the area shaded by blinds to the total window area, A_{win} is the total window area, E_{inc} is the incident solar radiation, *SHGC* is the solar heat gain coefficient, and *IAC* is the indoor attenuation coefficient for blinds, which are made of light translucent fabric. The *SHGC* values of the windows used in the study are obtained from the as-built drawings of the test building. For a given window type and climate, *SHGC* is generally considered a constant parameter[34] and is commonly used in evaluating building energy performance.[35] As per the ASHRAE Handbook, *SHGC* and *IAC* are assumed constant, with values of 0.287 and 0.75 (for the shaded region), respectively.[33]

Table 1. Specifications of the sensors installed in the test space

Location	Sensors	Range	Accuracy
Outdoor	Air temperature	−40 – 120°C	±0.25°C
	Relative humidity	0 – 100%	±1.5%
	Solar radiation	0 – 2000 W/m^2	±10 W/m^2
Indoor room space	Air temperature	0 – 50°C	0.2°C
	Globe temperature	0 – 50°C	0.5°C
	Relative humidity	0 – 100%	±1.7%
	CO_2	0 – 2000 ppm	±30 ppm
ACMV system	BTU meter	0 – 50 kW	±1%
	Elec. power meter	0 – 5 kW	±0.5%
	Duct air temperature	0 – 50°C	±0.2°C
	Duct RH	0 – 100%	±1.5%
	Duct airflow	0 – 16 m/s	±5% of the reading
	Duct CO_2	0 – 2000 ppm	±30 ppm

Abbreviations: ACMV: Air conditioning and mechanical ventilation; BTU: British thermal unit; RH: Relative humidity.

$$Q_{win} = \underbrace{SR * A_{win} * E_{inc} * SHGC * IAC}_{\text{region shaded by blinds}} + \underbrace{(1-SR) * A_{win} * E_{inc} * SHGC}_{\text{unshaded region}} \tag{I}$$

The MPC framework for ACMV has two primary objectives: (i) minimizing the cooling power (Q_{cool}) consumed by the ACMV system and (ii) maximizing thermal comfort by maintaining the indoor PMV as close as possible to thermal neutrality (i.e., $PMV_{ref} = 0$). Feedback from room air temperature, humidity, and globe temperature sensors is used to determine PMV.[33] The objective function for ACMV is formulated as shown in Equation II:

$$J = Minimize(\sum_{k=0}^{N} \frac{W_{cool} * Q_{cool,\, t+k|t}}{COP} + \sum_{k=0}^{N} W_{PMV} * (PMV_{t+k|t} - PMV_{ref})^2 + \sum_{k=0}^{N} W_{\epsilon} * (\epsilon_{t+k|t})^2) \tag{II}$$

This objective function is subject to the following constraints, as expressed in Equations III–IV:

$$Q_{\mathrm{cool,lb}} \leq Q_{\mathrm{cool}} \leq Q_{\mathrm{cool,ub}} \quad \text{(III)}$$

$$-0.5 \leq \mathrm{PMV} \leq 0.5 \quad \text{(IV)}$$

where Q, W, N, and ϵ refer to the cooling power, weighting factor, number of control intervals within one prediction horizon, and slack variable, respectively. The subscripts *cool*, *t*, *k*, *ref*, *lb*, and *ub* refer to cooling, current time, index of the time step, preferred PMV (i.e., PMV = 0), lower bound, and upper bound, respectively. The slack variable (ϵ) is used for constraint relaxation. In addition, the coefficient of performance (*COP*) of the cooling system is assumed to be constant at 3.7 as per the specification of the ACMV system. The three terms on the right-hand side of Equation II correspond to the cost functions related to cooling energy consumption, thermal discomfort, and constraint violation. The variables associated with the multizone MPC are summarized in Table 2. A more detailed explanation of Equations I-IV is provided in the Appendix.

4. ML model development for MPC

ML model training plays a pivotal role in developing predictive models for MPC. This section outlines the approach used to select, train, and evaluate neural network models for predicting PMV. This study frames the prediction of PMV as a regression task, given the continuous nature of this comfort index. An overview of the model training process and its integration into the MPC framework is presented in Figure 3.

Table 2. Summary of variables used in multizone MPC

Variable type in MPC	Variable type in ML models	Variables	Description
Manipulated variable	Inputs	Q_{cool}	Cooling power supplied by FCU
Measured disturbances		Temperature	Outdoor ambient temperature
		Solar heat gain	Measured from solar irradiance
		Adjacent PMV	PMV of adjacent/connected zones
		Occupancy	Occupancy measured from CO_2 concentration in return air
		Mean radiant temperature	Mean radiant temperature
State variable		PMV	Current indoor PMV
Output variable	Output	PMV	Indoor PMV

Abbreviations: FCU: Fan coil unit; ML: Machine learning; MPC: Model predictive control; PMV: Predicted mean vote.

For this study, four ML algorithms – SVM, RF, NARX, and LSTM – were selected for evaluation. Following the testbed setup, historical operational data from the building was collected over 2 months. This dataset was then preprocessed and divided into training, validation, and testing subsets. Each ML model underwent training, prediction, and evaluation processes. The model demonstrating the best prediction accuracy was selected for the construction of the MPC system in this study. Below, we elaborate on dataset creation, model evaluation criteria, model selection (including hyperparameter tuning), and the results of the training process.

4.1. Data collection and model evaluation

4.1.1. Dataset creation

The dataset used for model training was derived from historical data collected from the baseline measurement phase from August to October 2021 as part of the project's test plan. Given Singapore's consistently low month-to-month climatic variations throughout the year,[34] this three-month baseline dataset, inclusive of outdoor conditions, can be considered representative of the entire year. However, it is important to acknowledge that the model should ideally be trained on a dataset capturing a full range of weather conditions and building operational characteristics. In regions experiencing significant seasonal climate variations, a more extensive dataset covering a longer period would be necessary. Furthermore, certain model parameters – such as the lag length for the NARX model or the input depth for the LSTM model, defining the context window for learning temporal patterns – may need adjustment to accommodate these variations. Such adaptation could introduce challenges related to computational costs, overfitting, and vanishing gradients, which require careful consideration. Future research will focus on validating the model's performance across diverse climatic zones and exploring strategies to enhance its generalizability.

In the current study, the input variables include cooling power, outdoor temperature, solar irradiance, occupancy, mean radiant temperature, and the current indoor PMV, whereas the output is future indoor PMV prediction. A summary of the collected data is provided in Table 2.

4.1.2. Handling historical data

The dataset comprises 15,606 time steps, recorded at 6-min intervals. Data preprocessing included dimensionality reduction and Z-score normalization to minimize scale effects and improve computational efficiency. The data were partitioned into training (70%), validation (15%), and testing (15%) subsets to facilitate model development and evaluation. The Z-score standardization formula[36] used is shown in Equation V:

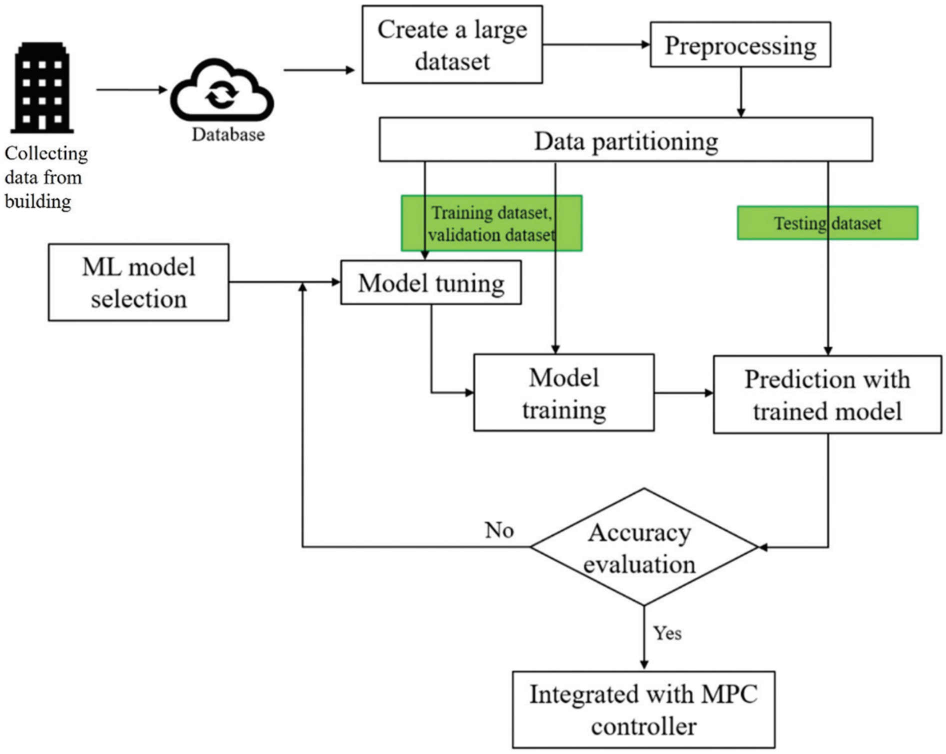


Figure 3. Overview of ML model training and integration loop
Abbreviations: ML: Machine learning; MPC: Model predictive control.

$$z_i = \frac{x_i - \overline{x}}{s} \tag{V}$$

where z_i is the Z-score for the i^{th} data point, x_i is the i^{th} data point, s is the standard deviation, and $\overline{x}$ is the mean.

4.1.3. Model evaluation

The performance of the models was evaluated based on the mean absolute error (MAE) and computational time. MAE is calculated using Equation VI:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \widehat{y_i}\right| \tag{VI}$$

where n is the number of data points, y_i is the actual value of the i^{th} data point, and $\widehat{y_i}$ is the predicted value of the i^{th} data point. MAE is commonly used as a performance metric for ML model training in building control applications and has demonstrated good results in similar studies.[37,38] Therefore, this study adopts this metric following established literature.

4.2. Training results

Before training the models for PMV prediction, their hyperparameters were optimized. Table 3 summarizes the hyperparameters employed and their respective optimization techniques.

Table 4 compares the performance of non-ANN ML models (SVM and RF) and ANN models (NARX and LSTM) using MAE as a measure of model accuracy and maximum prediction error as an indication of generalization capability. The best results are indicated. Computational cost is represented as the sum of training and prediction times.

As seen in Table 4, SVM showed a comparable MAE to RF but had slightly poorer generalization, as indicated by its higher maximum prediction error. However, SVM significantly outperformed RF in terms of computational cost. Nonetheless, both non-ANN models demonstrated inferior MAE compared to ANN models (LSTM and NARX).

For RF, Out-of-Bag feature importance analysis can intuitively demonstrate the influence of different input variables on the output variables. This feature sensitivity analysis, presented in Figure 4, indicated that cooling power and current PMV were the most influential predictors for PMV prediction. The higher MAE observed for RF, compared to the ANN models, could be attributed to RF's lack of a time-memory structure, limiting its

Table 3. Hyperparameters optimization methods for the PMV prediction models

Model name	Hyperparameters tuned
SVM	1. Box constraint (C): Balances model complexity and training error. Higher values lead to better fit but risk overfitting. 2. Kernel scale (σ): Determines the shape of the RBF kernel, controlling the decision boundary's flexibility. 3. Epsilon (ε): Specifies the tolerance level for prediction errors, influencing model robustness. Bayesian optimization identified the optimal hyperparameters, with the kernel scale showing the most significant impact on accuracy.
RF	1. Number of trees: Determines the number of decision trees in the ensemble. More trees generally increase accuracy but also computational cost. 2. Minimum leaf size: Regulates the minimum number of samples in a leaf node to control model complexity and prevent overfitting. 3. Number of predictors for splitting: Adjusted to optimize feature selection at each decision split. Bayesian optimization was used for hyperparameter tuning, revealing that increasing the number of trees enhanced prediction accuracy, while smaller leaf sizes improved granularity but increased training time.
NARX neural network	The NARX model structure involved determining the number of neurons in the hidden layer and lag lengths for inputs and outputs. Bayesian optimization was applied to identify the optimal combination of hyperparameters.
LSTM	LSTM model training involved tuning the number of LSTM units and dropout rates to prevent overfitting and learning rates. Bayesian optimization and trial-and-error methods were used to achieve optimal model configurations.

Abbreviations: LSTM: Long short-term memory; NARX: Non-linear autoregressive network with exogenous inputs; PMV: Predicted mean vote; RBF: Radial basis function; RF: Random forest; SVM: Support vector machine.

Table 4. Summary of ML model training

ML model	MAE (Test dataset)	Maximum prediction error	Computing cost (s)		
			Training	Prediction	Total
SVM	0.1041	0.4810	235	1[a]	236
RF	0.1040	0.4200	18[a]	450	468
NARX	0.0623[a]	0.3140[a]	96	2	98[a]
LSTM	0.0843	0.4830	840	26	866

Note: [a]Best results.
Abbreviations: LSTM: Long short-term memory; MAE: Mean absolute error; ML: Machine learning; NARX: Non-linear autoregressive network with exogenous inputs; RF: Random forest; SVM: Support vector machine.

applicability in scenarios characterized by strong temporal dependencies.[20]

The NARX model exhibited the best accuracy and generalization among all models, along with the lowest computational cost. Potential improvements in NARX's accuracy could be achieved by increasing the number of hidden layers, but this would increase computational time.

The LSTM model significantly outperformed the non-ANN models in PMV prediction, displaying substantially better prediction accuracy. However, it did not show an advantage in generalization, as indicated by its maximum prediction error. The LSTM model required the highest training times due to the computational complexity of LSTM cells. Thus, its superior prediction accuracy came at a higher computational cost.

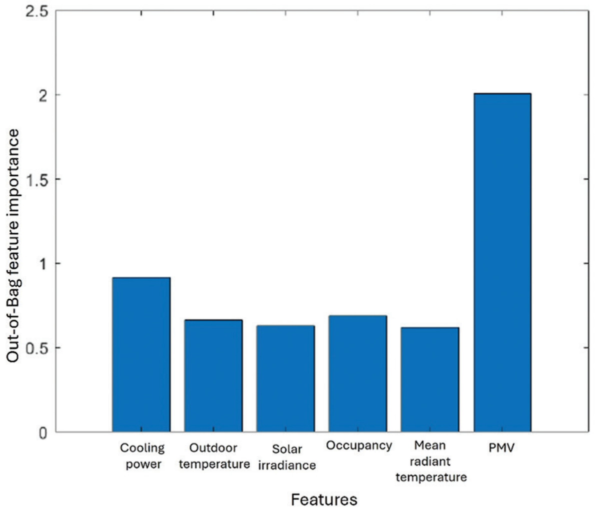


Figure 4. Feature sensitivity analysis of predictors
Abbreviation: PMV: Predicted mean vote.

In summary, ANN models outperformed non-ANN models in terms of accuracy. Among the ANN models, the choice between NARX and LSTM depends on specific application requirements. For real-time implementation with limited computational resources, NARX is preferable. Conversely, for complex systems where accuracy is paramount, LSTM is more appropriate. Given that NARX achieved the lowest test and prediction errors, along with the lowest combined training and prediction cost, it was selected for integration into the MPC framework in this study.

5. Results and discussion

In this section, the performance of the MPC system is compared against the baseline BMS mode of operation. Data collected over a 7-day period for both the BMS and MPC systems indicated that the statistical variations of outdoor temperature and solar radiation were similar.

As shown in Figure 5A and B, the median outdoor air temperatures during the test days were 30°C for the BMS and 30.1°C for the MPC tests, while median solar radiation was approximately 309 and 274 W/m^2, respectively. These comparable outdoor conditions ensure that subsequent performance comparisons are based on similar heat load scenarios.

Figure 6A-C presents the time series data of measured CO$_2$ levels, indoor air temperature, and indoor PMV in the test room for a selected typical day. This day was chosen to illustrate the performance of BMS and MPC, as depicted in Figure 6, through a daily time series graph. The selection criterion for the typical day was based on the daily median outdoor temperature that most closely matched the median outdoor temperature of the entire 7-day measurement period. In addition, the comparison of CO$_2$ levels in Figure 6A indicates similar room occupancy patterns for BMS and MPC, except for the lunch hour between 12 PM and 1 PM.

Figure 6B suggests that under BMS, the room is consistently overcooled throughout the day due to the fixed fan speed settings in the PAUs and FCUs, as well as the indoor air temperature setpoint of 21°C. This occurs regardless of occupancy or outdoor conditions. In contrast, MPC accounts for multiple inputs, including room occupancy and external heat load, to dynamically adjust the cooling power delivered to the room. As a result, MPC maintains a more stable indoor temperature and PMV profile throughout the day compared to the test building's original BMS. The variation in ACMV cooling power, measured using a BTU meter and fan power meter for a typical day under both BMS and MPC, is shown in Figure 6D. Through its predictive capability, MPC effectively regulates cooling power while enhancing indoor thermal comfort, as suggested by Figure 6C.

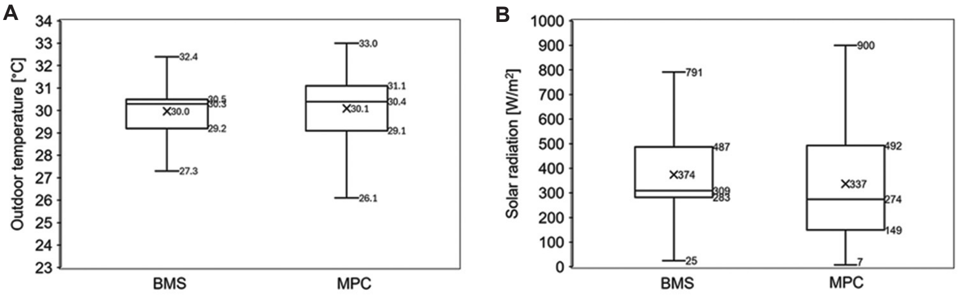


Figure 5. Weather conditions of test days. (A) Variation in outdoor temperature, (B) Variation in solar radiation.

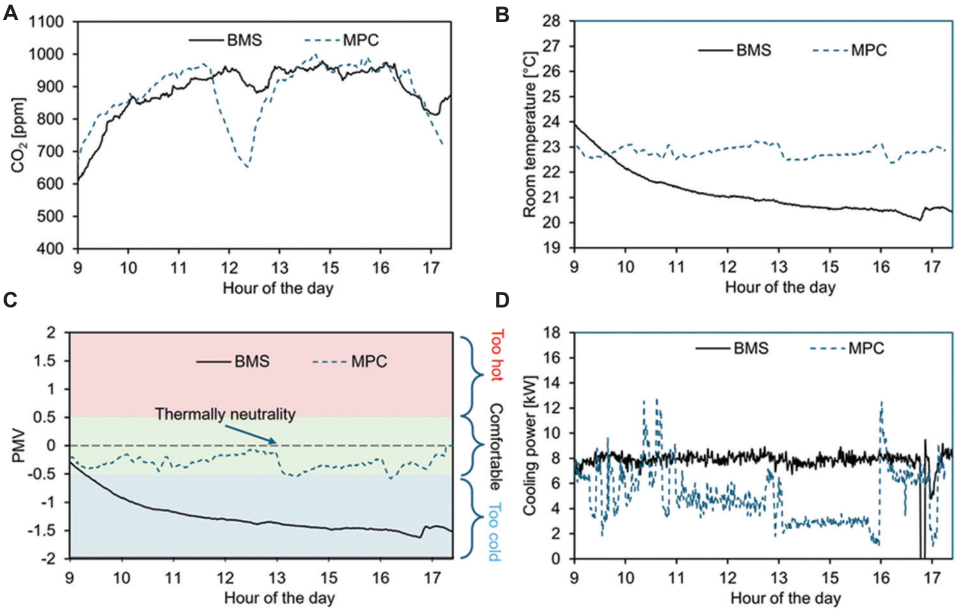


Figure 6. Typical variations of (A) indoor CO$_2$ in a test space, (B) indoor temperature, (C) indoor PMV, and (D) Variation of ACMV Cooling Power for a day

Abbreviations: BMS: Building management system; MPC: Model predictive control; PMV: Predicted mean vote.

Figure 7A-E summarizes the control performance of BMS and MPC over the 1-week measurement period. Figure 7A illustrates the distribution of indoor CO_2 levels for the week under both BMS and MPC. The median CO_2 levels for BMS and MPC were found to be 610.3 and 645.4 ppm, respectively, suggesting comparable occupancy levels. This, along with the outdoor data, establishes similar heat load conditions for the ACMV system, allowing for a fair thermal comfort comparison. Figure 7B presents the temperature distribution over the seven-day test period, where the median temperature under BMS was 20.8°C compared to 22.8°C under MPC. In terms of thermal comfort, Figure 7C clearly demonstrates that BMS overcools the room, resulting in <1% of its operational time within the thermally comfortable PMV ranges (–0.5 – 0.5). Conversely, MPC is designed to maintain PMV within the comfortable zone (as defined by Equation IV), ensuring acceptable thermal comfort for more than 98% of its operation time. In addition, the small interquartile range in PMV, as seen in Figure 7C, indicates that MPC maintains stable indoor thermal comfort for a long period of time with minimal variation. As a consequence, MPC significantly reduces cooling energy waste, which is a common issue with BMS due to overcooling. Figure 7D illustrates the reduction in total cooling power consumption, measured through BTU and fan power meters. Given Singapore's consistently hot and humid climate throughout the year, the daily cooling energy consumption recorded over the 7-day test period was extrapolated to estimate annual consumption. Figure 7E shows that by leveraging predictive capabilities and multiobjective optimization, MPC reduces energy consumption by over 42% compared to BMS. Long-term performance sustainability of MPC can be achieved through periodic online learning, which calibrates MPC to account for changes in occupancy patterns, climatic conditions, and/or building dynamics. In our previous study, we successfully deployed such an adaptive MPC system incorporating online learning.[39]

A further breakdown of the total cooling energy consumption associated with FCUs was conducted. The FCU fans installed in the test space were constant-speed fans, meaning that electrical power consumption by these

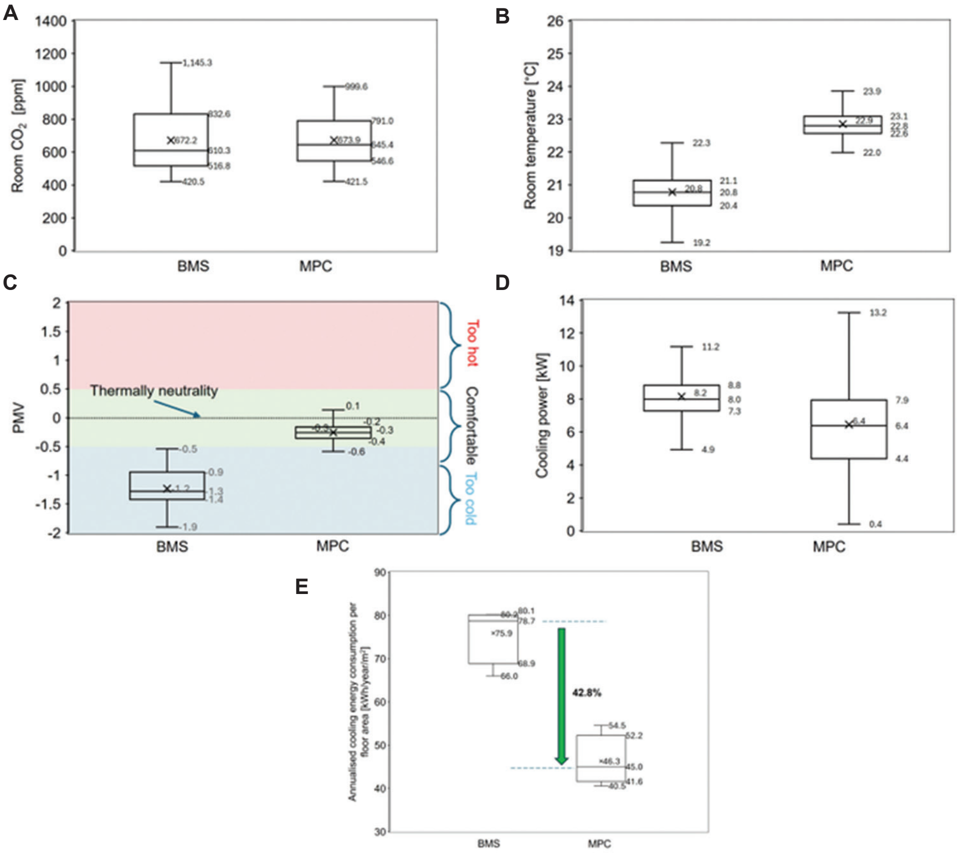


Figure 7. Statistical distributions of indoor conditions on test days: (A) indoor CO_2, (B) indoor temperature, (C) indoor PMV, (D) ACMV cooling cower, and (E) annualized cooling energy consumption per floor area

Abbreviations: ACMV: Air conditioning and mechanical ventilation; BMS: Building management system; MPC: Model predictive control; PMV: Predicted mean vote.

fans remained approximately the same for both baseline BMS and MPC. During MPC duration, it was found that approximately 32% of the total cooling energy consumption was attributable to electric fan power. If FCU fans equipped with variable-speed drives were installed instead of constant-speed fans, MPC would have the opportunity to further optimize fan power consumption. For example, MPC could be configured to reduce FCU fan speed by over 70% compared to constant-speed operation when no occupants are present in the test rooms. By adopting such control logic for optimizing fan operation, a conservative reduction of 10 – 15% in fan power consumption was estimated. This estimation excludes the reduction in cooling energy produced by chilled water (which can be measured by the BTU meter) during the heat exchange process with the reduced airflow in the FCU. Such a detailed impact of reduced fan power on cooling energy consumption will be investigated in future studies. Excluding the potential effect of airflow reduction on chilled water cooling energy, an additional 3 – 5% reduction in total cooling energy consumption was estimated.

A key advantage of our MPC implementation is its modular and adaptable framework, involving: (i) collecting building data; (ii) developing a control-oriented model using ML tools; (iii) designing the optimization algorithm; and (iv) integrating the MPC with the BMS. While this framework is generally applicable to buildings in various climates and with diverse operational profiles, building-specific retraining is necessary to accurately capture unique thermal dynamics. In environments with significant seasonal or regional variations, model parameters (e.g., lag length for NARX or input depth for LSTM) may require adjustment, posing challenges such as increased computational cost and potential overfitting. Future studies will further validate the framework across different climatic zones.

A key challenge in MPC deployment is balancing model complexity against computational cost. In our current implementation, the NARX model provided superior accuracy and efficiency (Table 4). For more complex systems, LSTM architectures – with increased input depth and larger hidden layers – could offer enhanced prediction accuracy, albeit with higher computational demands. In such cases, employing artificial intelligence/ML-optimized edge hardware and multithreaded computing can help manage computational loads. In our previous work,[11] we introduced an approach using instantaneous linearization within ML-based MPC to further address these challenges. Although a detailed exploration of this topic is beyond the scope of the present paper, it represents a promising direction for future research.

6. Conclusion

This study demonstrated the effectiveness of an ML-based MPC system in optimizing energy consumption and occupant comfort in a multizone commercial building. The implemented system achieved energy savings exceeding 42% compared to conventional thermostat-based control while simultaneously improving thermal comfort and stability. The novelty of our approach lies in its real-world, large-scale deployment in an 11-zone building, integration of ML-based weather forecasting for proactive control adjustments, and the use of a NARX neural network for accurate PMV prediction.

While initial implementation costs and model development efforts present challenges for widespread adoption, future research endeavors aim to mitigate these limitations. Such efforts include developing comfort models that rely exclusively on existing building sensors, exploring cost-effective sensor technologies, and examining the integration of additional building systems, such as lighting and shading, for more holistic optimization strategies. The present study contributes to advancing intelligent building automation systems, with significant potential to enhance energy efficiency and sustainability in the built environment.

Acknowledgments

Technical support from Dr. Wai Soong Loh from JTC for this project is much appreciated. We also appreciate the patience, understanding, and logistical support of the administrative team from Civil Service College, Singapore, for this project.

Funding

This research is jointly supported by Jurong Town Corporation (JTC) of Singapore (NTU REF 2019-0607) and Smart Nation and Digital Government Office, SNDGO of Singapore (NRF2016IDM-TRANS001-031).

Conflict of interest

The authors declare they have no competing interests.

Author contributions

Conceptualization: Pradeep Shakya, Man Pun Wan
Data Curation: Shiva Sreenivasan
Formal analysis: Pradeep Shakya, Man Pun Wan
Funding acquisition: Shiyu Yang, Man Pun Wan
Investigation: Shiva Sreenivasan, Pradeep Shakya
Methodology: Pradeep Shakya
Resources: Baskaran Krishnamoorthy, Shiyu Yang
Writing – original draft: Pradeep Shakya, Shiva Sreenivasan
Writing – review & editing: Man Pun Wan, Pradeep Shakya, Shiva Sreenivasan

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

The datasets presented in this article are not readily available due to a confidentiality agreement with the funding agency.

Further disclosure

Part of or the entire set of findings have been presented at 2024 Herrick Conferences in Compressor Engineering, Refrigeration and Air Conditioning and High-Performance Buildings (held on July 14, 2024, at Purdue University, West Lafayette, Indiana, United States of America) hosted by the Ray W. Herrick Laboratories and the Center for High-Performance Buildings in West Lafayette, Indiana, United States of America.

References

1. Building and Construction Authority (BCA). *Super Low Energy Building Technology Roadmap*. Building and Construction Authority; 2018. Available from: https://www1.bca.gov.sg/docs/default-source/docs-corp-buildsg/sustainability/sle-tech-roadmap-report--published-ver1-1.pdf?sfvrsn=f2df22ed_0 [Last accessed on 2024 Dec 05].
2. Singapore Green Building Council (SGBC). *Singapore Green Building Masterplan*. Singapore Green Building Council; 2020. Available from: https://www.sgbc.sg/about-green-building/sgbmp [Last accessed on 2024 Dec 05].
3. Building and Construction Authority (BCA). *Green Building Masterplan*. Building and Construction Authority; 2021. Available from: https://www1.bca.gov.sg/docs/default-source/docs-corp-buildsg/sustainability/sgbmp-80-80-80-in-2030-infographic.pdf?sfvrsn=57172d48_2 [Last accessed on 2024 Dec 05].
4. Das HP, Lin YW, Agwan U, *et al*. Machine learning for smart and energy-efficient buildings. *Environ Data Sci*. 2024;3:e1.

   doi: 10.1017/eds.2023.43
5. Qiang G, Tang S, Hao J, Di Sarno L, Wu G, Ren S. Building automation systems for energy and comfort management in green buildings: A critical review and future directions. *Renew Sustain Energy Rev*. 2023;179:113301.

   doi: 10.1016/j.rser.2023.113301
6. Pinheiro S, Wimmer R, O'Donnell J, *et al*. MVD based information exchange between BIM and building energy performance simulation. *Autom Constr*. 2018;90:91-103.

   doi: 10.1016/j.autcon.2018.02.009
7. Mauri L, Vallati A, Ocłoń P. Low impact energy saving strategies for individual heating systems in a modern residential building: A case study in Rome. *J Clean Prod*. 2019;214:791-802.

   doi: 10.1016/j.jclepro.2018.12.320
8. Mathews Roy A, Prasanna Venkatesan R, Shanmugapriya T. Simulation and analysis of a factory building's energy consumption using eQuest software. *Chem Eng Technol*. 2021;44(5):928-933.

   doi: 10.1002/ceat.202000489
9. Andriamamonjy A, Saelens D, Klein R. An automated IFC-based workflow for building energy performance simulation with Modelica. *Autom Constr*. 2018;91:166-181.

   doi: 10.1016/j.autcon.2018.03.019
10. Yang S, Wan MP, Ng BF, *et al*. A state-space thermal model incorporating humidity and thermal comfort for model predictive control in buildings. *Energy Build*. 2018;170:25-39.

    doi: 10.1016/j.enbuild.2018.03.082
11. Yang S, Wan MP. Machine-learning-based model predictive control with instantaneous linearization-A case study on an air-conditioning and mechanical ventilation system. *Appl Energy*. 2022;306:118041.

    doi: 10.1016/j.apenergy.2021.118041
12. Široký J, Oldewurtel F, Cigler J, Prívara, S. Experimental analysis of model predictive control for an energy efficient building heating system. *Appl Energy*. 2011;88:3079-3087.

    doi: 10.1016/j.apenergy.2011.03.009
13. Ma Y, Borrelli F, Hencey B, Coffey B, Bengea S, Haves P. Model predictive control for the operation of building cooling systems. *IEEE Trans Control Syst Technol*. 2012;20:796-803.

    doi: 10.1109/TCST.2011.2124461
14. Pang X, Duarte C, Haves P, Chuang F. Testing and demonstration of model predictive control applied to a radiant slab cooling system in a building test facility. *Energy Build*. 2018;172:432-441.

    doi: 10.1016/j.enbuild.2018.05.013
15. Yang Y, Srinivasan S, Hu G, Spanos CJ. Distributed control of multizone HVAC systems considering indoor air quality. *IEEE Trans Control Syst Technol*. 2021;29:2586-2597.

    doi: 10.1109/TCST.2020.3047407
16. Yang S, Chen W, Wan MP. A machine-learning-based event-triggered model predictive control for building energy management. *Build Environ*. 2023;233:110101.

    doi: 10.1016/j.buildenv.2023.110101
17. Fathi S, Srinivasan R, Fenner A, Fathi S. Machine learning applications in urban building energy performance forecasting: A systematic review. *Renew Sustain Energy Rev*. 2020;133:110287.

    doi: 10.1016/j.rser.2020.110287
18. Seyedzadeh S, Rahimian FP, Rastogi P, Glesk I. Tuning machine learning models for prediction of building energy loads. *Sustain Cities Soc*. 2019;47:101484.

doi: 10.1016/j.scs.2019.101484

19. Zhou X, Xu L, Zhang J, *et al*. Data-driven thermal comfort model via support vector machine algorithms: Insights from ASHRAE RP-884 database. *Energy Build*. 2020;211:109795.

    doi: 10.1016/j.enbuild.2020.109795

20. Chaudhuri T, Zhai D, Soh YC, Li H, Xie L. Random forest based thermal comfort prediction from gender-specific physiological parameters using wearable sensing technology. *Energy Build*. 2018;166:391-406.

    doi: 10.1016/j.enbuild.2018.02.035

21. Koschwitz D, Frisch J, Van Treeck C. Data-driven heating and cooling load predictions for non-residential buildings based on support vector machine regression and NARX Recurrent Neural Network: A comparative study on district scale. *Energy*. 2018;165:134-142.

    doi: 10.1016/j.energy.2018.09.068

22. Zhang C, Li J, Zhao Y, Li T, Chen Q, Zhang X. A hybrid deep learning-based method for short-term building energy load prediction combined with an interpretation process. *Energy Build*. 2020;225:110301.

    doi: 10.1016/j.enbuild.2020.110301

23. Beltran A and Cerpa AE. Optimal HVAC Building Control with Occupancy Prediction. In: *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-efficient Buildings*; 2014. p. 168-171.

    doi: 10.1145/2674061.2674072

24. Li B, Xia L. A Multi-grid Reinforcement Learning Method for Energy Conservation and Comfort of HVAC in Buildings. In: *2015 IEEE International Conference on Automation Science and Engineering* (*CASE*); 2015. p. 444-449.

    doi: 10.1109/CoASE.2015.7294119

25. Zhang Z and Lam KP. Practical Implementation and Evaluation of Deep Reinforcement Learning Control for a Radiant Heating System. In: *Proceedings of the 5th Conference on Systems for Built Environments*; 2018. p. 148-157.

    doi: 10.1145/3276774.3276775

26. Ding X, Cerpa A, Du W. Exploring deep reinforcement learning for holistic smart building control. *ACM Trans Sens Netw*. 2024;20(3):1-28.

    doi: 10.1145/3656043

27. Shamachurn H, Seebaruth M, Kowlessur NS, Hassen SS. Real-time model predictive control of air-conditioners through IoT-results from an experimental setup in a tropical climate. *Adv Control Appl Eng Ind Syst*. 2024;6:e232.

    doi: 10.1002/adc2.232

28. Hu G, You F. Multi-zone building control with thermal comfort constraints under disjunctive uncertainty using data-driven robust model predictive control. *Adv Appl Energy*. 2023;9:100124.

    doi: 10.1016/j.adapen.2023.100124

29. Joe J, Im P, Cui B, Dong J. Model-based predictive control of multi-zone commercial building with a lumped building modelling approach. *Energy*. 2023;263:125494.

    doi: 10.1016/j.energy.2022.125494

30. Oldewurtel F, Parisio A, Jones CN, *et al*. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy Build*. 2012;45:15-27.

    doi: 10.1016/j.enbuild.2011.09.022

31. Hou J, Li H, Nord N, Huang G. Model predictive control under weather forecast uncertainty for HVAC systems in university buildings. *Energy Build*. 2022;257:111793.

    doi: 10.1016/j.enbuild.2021.111793

32. Mazar MM, Rezaeizadeh A. Adaptive model predictive climate control of multi-unit buildings using weather forecast data. *J Build Eng*. 2020;32:101449.

    doi: 10.1016/j.jobe.2020.101449

33. ASHRAE Handbook. *Heating, Ventilating, and Air-Conditioning Systems and Equipment*. Vol. 39. Atlanta, GA, USA: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.; 1996.

34. Meteorological Service Singapore (MSS). *Climate of Singapore*. Meteorological Service Singapore; 2020. Available from: https://www.weather.gov.sg/climate-climate-of-singapore [Last accessed on 2024 Dec 05].

35. Yang S, Wan MP, Ng BF, *et al*. Model predictive control for integrated control of air-conditioning and mechanical ventilation, lighting and shading systems. *Appl Energy*. 2021;297:117112.

    doi: 10.1016/j.apenergy.2021.117112

36. Freedman DA. *Statistical Models: Theory and Practice*. United Kingdom: Cambridge University Press; 2009.

37. Chifu VR, Pop CB, Chifu ES, Barleanu H. Deep Learning for Forecasting the Energy Consumption in Public Buildings. In: *2021 20th RoEduNet Conference: Networking in Education and Research* (*RoEduNet*); 2021. p. 1-6.

    doi: 10.48550/arXiv.2207.11953

38. Yang R, Hao J, Jiang H, Jin X. *Machine-Learning-Driven, 2020, Site-Specific Weather Forecasting for Grid-Interactive Efficient Buildings*. Golden, CO, United States: National Renewable Energy Lab. (NREL); 2020. Available from: https://www.osti.gov/biblio/1669587 [Last accessed on 2025 Feb 06].

39. Yang S, Wan MP, Chen W, Ng BF, Dubey S. Model predictive control with adaptive machine-learning-based model for building energy efficiency and comfort optimization. *Appl Energy*. 2020;271:115147.

    doi: 10.1016/j.apenergy.2020.115147

40. Zhao S, Cajo R, De Keyser R, Liu S, Ionescu CM. Nonlinear predictive control applied to steam/water loop in large scale ships. *IFAC PapersOnLine*. 2019;52(1):868-873.

    doi: 10.1016/j.ifacol.2019.06.171

Appendix

1. Elaboration on MPC Equations

This appendix provides a comprehensive explanation of the equations presented in Section 3, which govern the MPC controller for the ACMV system.

Equation 1: Solar heat gain

Solar heat gain through windows is a critical component of building thermal load modeling. The formulation accounts for dynamic shading effects, which modulate solar radiation entering space. The total solar heat gain, Q_{win}, is expressed as:

$$Q_{win} = \underbrace{SR * A_{win} * E_{inc} * SHGC * IAC}_{region\ shaded\ by\ blinds} + \underbrace{\left(1-SR\right) * A_{win} * E_{inc} * SHGC}_{unshaded\ region} \tag{I}$$

Description of key variables in Equation I:

- Shading ratio (SR): Represents the fraction of the window area covered by blinds. SR can vary dynamically based on occupant preferences or automated shading control strategies
- Solar heat gain coefficient (SHGC): Typically ranges between 0.3 and 0.9 depending on glazing properties (e.g., low-emissivity coatings, double/triple glazing). SHGC values are often derived from standardized tests (ASHRAE 90.1) or manufacturer data
- Indoor attenuation coefficient (IAC): Quantifies the reduction in solar radiation due to blinds. For example, horizontal slat blinds with high reflectivity may have an IAC of 0.5 – 0.7, while blackout shades could reduce IAC to near zero. For this study, as per ASHRAE Handbook,[33] IAC of 0.75 was assumed for blinds made of light translucent fabric
- Incident radiation (E_{inc}): Includes direct, diffuse, and reflected solar radiation, which depends on window orientation, time of day, and geographic location.

Expanded discussion for Equation I:

The equation assumes uniform shading distribution across the window, which may simplify real-world scenarios where partial shading or non-uniform blind deployment occurs. The model does not explicitly account for spectral properties of solar radiation or transient thermal storage in glazing materials, though these effects are often negligible for hourly or sub-hourly MPC timescales. For further validation, SHGC and IAC values should align with empirical measurements or established databases (e.g., Lawrence Berkeley National Laboratory's Window Module).

Equation II: Objective function

The MPC controller's objective function, *JJ*, balances energy efficiency, thermal comfort, and operational feasibility. Building on the framework from,[39] the cost function is defined as:

$$J = Minimize(\sum_{k=0}^{N} \frac{W_{cool} * Q_{cool,\ t+k|t}}{COP} + \sum_{k=0}^{N} W_{PMV} * (PMV_{t+k|t} - PMV_{ref})^2 + \sum_{k=0}^{N} W_{\epsilon} * (\epsilon_{t+k|t})^2) \tag{II}$$

Description of key variables in Equation II:

- Normalized cooling power (Q_{cool}): The first term uses cooling power normalized by the system's maximum capacity to ensure scalability across different HVAC systems
- Normalized PMV (PMV): The second term penalizes deviations from the neutral setpoint ($PMV_{ref} = 0$), normalized by the acceptable comfort range (±0.5\pm 0.5). This normalization ensures equitable weighting between thermal comfort and energy use.
- Weighting factors:
  - $W_{cool} = 1/10$: Prioritizes thermal comfort over energy savings, reflecting occupant-centric design principles
  - $W_{PMV} = 4$: Prioritizes thermal comfort over energy savings, reflecting occupant-centric design principles
  - $W_{\epsilon} = 10{,}000$: Strongly penalizes constraint violations to enforce strict adherence to comfort and equipment limits.
- Prediction horizon: Set to 12 control intervals (60 minutes), chosen to match the thermal response time of the building (~40 minutes for PMV stabilization during morning start-up). A longer horizon would increase computational complexity without improving performance.[39,40]

Practical implications for Equation II:

The weighting scheme reflects a design philosophy where occupant comfort is prioritized, but energy efficiency remains a secondary goal. The slack variable ϵ ensures feasibility under unexpected disturbances (e.g., occupancy spikes), though its high penalty weight (W_ϵ) minimizes its use in practice.

Equations III and IV

The optimization is bounded by operational and comfort limits:

$$Q_{\mathrm{cool,lb}} \leq Q_{\mathrm{cool}} \leq Q_{\mathrm{cool,ub}} \tag{III}$$

$$-0.5 \leq \mathrm{PMV} \leq 0.5 \tag{IV}$$

Contextual details for Equations III and IV:

- Cooling power bounds ($Q_{cool,\, lb}$, $Q_{cool,\, ub}$) are determined by the HVAC system's capacity
- The PMV range (±0.5) corresponds to the ASHRAE Standard 55 “neutral” comfort zone, ensuring 80 – 90% occupant satisfaction.

Additional notes:

- Control interval: The MPC operates at 5-min intervals, balancing responsiveness with computational tractability
- Slack V=variable (ϵ): Introduced as a soft constraint to avoid infeasible solutions during transient disturbances (e.g., abrupt weather changes)
- Thermal comfort: The PMV model assumes steady-state conditions and standardized clothing/metabolic rates; dynamic adjustments for occupant activity are not modeled here.

ACCSCIENCE PUBLISHING

International Journal of AI for Materials and Design

ORIGINAL RESEARCH ARTICLE

Improvement of multiaxial fatigue life prediction performance based on contrastive learning feature extraction

Ziyu Cui[1], **Xingyue Sun**[2*], **and Xu Chen**[1,3]

[1]Department of Process Equipment & Control Engineering, School of Chemical Engineering and Technology, Tianjin University, Tianjin, China

[2]Department of Aeronautical Structure Engineering, School of Aeronautics, Northwestern Polytechnical University, Xi'an, Shaanxi, China

[3]Zhejiang Institute of Tianjin University, Ningbo, Zhejiang, China

(This article belongs to the *Special Issue: AI for Multiscale Analysis and Defect Identification in Packaging Structures and Semiconductor Chips*)

Abstract

Accurate prediction of multiaxial fatigue life was crucial for structural integrity assessment, yet the variability in material responses under complex loading paths made it challenging for both classical and data-driven models to achieve high accuracy. To address this issue, a contrastive learning-based framework was proposed in this study, enabling the construction of more generalized low-dimensional feature representations across different loading paths. This framework enhanced the robustness of fatigue life prediction without relying on mechanical assumptions. Experimental validation demonstrated that, compared to existing methods, the contrastive learning model learned more suitable feature encodings, significantly improving prediction performance. This framework provided a reference solution for engineering applications requiring reliability assessment under multiaxial stress conditions.

Keywords: Contrastive learning; Deep learning; Feature engineering; Life prediction; Multiaxial fatigue

***Corresponding author:**
Xingyue Sun
(xysun@nwpu.edu.cn)

Citation: Cui Z, Sun X, Chen X. Improvement of multiaxial fatigue life prediction performance based on contrastive learning feature extraction. *Int J AI Mater Design*. 2025;2(1):54-72. doi: 10.36922/IJAMD025040004

Received: January 22, 2025

Revised: March 6, 2025

Accepted: March 14, 2025

1. Introduction

In modern high-tech industries such as electronic packaging, aerospace, and nuclear power generation, the role of multiaxial fatigue analysis has become increasingly critical.[1-5] Electronic packaging materials are subjected to complex thermal and mechanical loads, which can precipitate premature material and structural failures, thereby severely damaging structural integrity.[6,7] Consequently, comprehensive research into multiaxial fatigue is essential for enhancing the reliability and service life of electronic devices at the design and serving stages. In addition, precise assessments of fatigue behavior under complex stress environments are crucial for ensuring the safe operation of aerospace vehicles, nuclear power stations, and other fields.[8-10] Such studies not only facilitate a better understanding of material responses under multiaxial stresses but also advance the design of materials and the evaluation of structural integrity, pivotal for the development of safer and more efficient technological solutions.

Currently, with the development of advanced intelligent algorithms, data-driven methods have been widely developed and applied in the field of material fatigue life prediction.[11-18] From classical shallow machine learning algorithms to deep learning models based on neural networks, these methods have demonstrated excellent performance in various fatigue-related problems.[19-24] Jiang *et al.*[25] proposed a physics-informed multilayer nested neural network framework, using stacking fault energy, strain amplitude, strain rate, and temperature as input features. Physical constraints were embedded in the loss function to ensure the model adhered to known physical laws. The performance of the model was validated on fatigue data of 316 stainless steel. Zhu *et al.*[26] developed a deep learning model called Multi-GAT (Multi-Graph Attention Network) for predicting the high-cycle fatigue (HCF) life of titanium alloys. This model is integrated with an attention mechanism and uses a graph structure as the data structure, allowing the full consideration of relationships between samples. This approach enables the accurate prediction of HCF life for various titanium alloys using a limited number of sample data. Liao *et al.*[27] proposed a path-dependent adaptive physics-informed neural network to address the non-proportionality caused by phase differences. The model embedded multiple critical plane models into the loss function and achieved optimization through dynamic weight adjustment. Genetic algorithms and a meta-learning framework were used to optimize the weight hyperparameters. The meta-learning framework enabled the weights of different physical terms in the loss function to dynamically adapt based on the load path information. The explored meta-learning framework, applied through transfer learning to predict the fatigue life of 316L and 304 stainless steels, demonstrated strong generalization capabilities. Chen *et al.*[28] proposed a multiview neural network model incorporating frequency domain analysis. This model integrates convolutional neural networks (CNN), long short-term memory networks, and FNet with frequency domain analysis in a parallel structure, extracting effective features from the material loading path to predict fatigue life. Through ablation experiments, the extrapolation capability of the model was verified using specific test datasets. Zhang *et al.*[29] used a SHapley Additive exPlanations-informed recursive feature elimination method to identify key features in a multiaxial fatigue experiment dataset. Symbolic regression was employed to extract and encapsulate expressions predicting fatigue life based on these salient features, which were then integrated into the traditional mean squared error (MSE). This significantly improved the predictive accuracy of the model on the existing database.

In fatigue life prediction, constructing appropriate input features is crucial for improving the model's predictive performance.[30-33] Simply using raw experimental features as inputs may not lead to optimal results. To achieve effective feature engineering, researchers have drawn on various traditional empirical models, such as critical plane models,[34,35] damage mechanics models,[36,37] and fracture mechanics models.[38,39] These empirical insights help in selecting and designing more relevant features, which significantly enhance the model's ability to predict fatigue life accurately. Figure 1 provides an intuitive comparison between traditional physics-based models and data-driven approaches. In traditional methods, domain experts manually extract key features using physics-

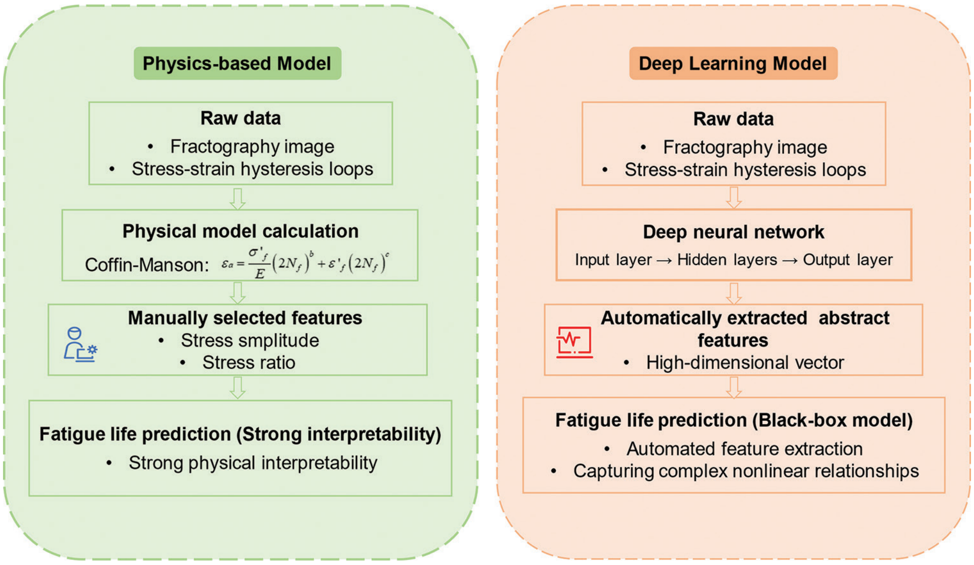


Figure 1. A comparison of fatigue life prediction frameworks: physics-based model versus deep learning model

based, whereas deep learning methods can automatically learn high-dimensional abstract representations from raw experimental data, capturing complex nonlinear relationships within the data. This comparison highlights the strengths and limitations of both approaches and underscores the importance of selecting appropriate input features for accurate fatigue life prediction. Wang *et al.*[40] proposed two physics-guided machine learning frameworks based on the Paris law and machine learning models. The first framework transforms the original features into new features using the Paris law, which are then combined with the original features and input into the machine learning model to predict fatigue life. The second framework integrates the Paris law and machine learning models using the Kalman filter, leveraging the advantages of both approaches and integrating information from different models. This enables the fusion of physical information and machine learning, allowing the model to account for factors ignored by physics-based models while ensuring consistency with physical results. Dong *et al.*[41] unified the influence of different defects through equivalent damage area representation using the M-integral fatigue model. By taking cyclic loading and equivalent damage area as inputs and fatigue life as output, the approach effectively improved the generalization ability and prediction accuracy of incomplete fatigue data. Fan *et al.*[42] expanded the original dataset using the Z-parameter model by restricting key parameters such as the size of critical defects, the relative depth of critical micro-defects, and stress levels within certain ranges. These extended datasets were then used to build machine learning models for ultra-high-cycle fatigue life prediction. With an appropriate increase in the size of the training set, the model demonstrated significantly improved accuracy and exhibited higher accuracy and stronger generalization compared to physical models. Gan *et al.*[43] used the Ye-Wang damage theory to derive theoretical estimates of material behavior, constructing additional features closely related to the desired outputs. This approach integrated original data information with domain knowledge. In addition, output standards were set to provide information for the data-driven process of model training and prediction, highlighting its potential in addressing small-sample problems. Wang *et al.* [44] characterized the fatigue life of additive manufacturing (AM) parts under different stress amplitudes using predictions from continuous damage mechanics associated with AM. These predictions, along with initial features such as experimental conditions, mechanical properties, porosity analysis, and surface morphology, were used as inputs. By learning the dependence on physical principles, the model can better map the nonlinear relationships between inputs and outputs.

The essence of feature engineering lies in the extraction and dimensionality reduction of complex, high-dimensional features, which brings samples closer together in high-dimensional space. Traditional models, based on physical laws and phenomenon analysis, focus on extracting and analyzing key values. However, many of these models are empirical, summarizing patterns that may not truly reflect the relationships between data samples, and they may not have clear physical models to describe the underlying dynamics. From the perspective of data relationships, dimensionality reduction or clustering techniques, such as K-means,[45] variational autoencoder,[46] and contrastive learning,[47-50] can also be employed with effective results. Among these, contrastive learning stands out by maximizing the consistency between similar samples and the disparity between different ones, enabling the learning of more robust and generalized feature representations. This approach does not require prior knowledge of label information, which reduces model usage costs. Furthermore, its adversarial sample pair learning strategy enhances the model's generalization ability. By optimizing the relative distances between samples, contrastive learning provides a powerful learning mechanism for complex datasets.[51] This is especially beneficial in unsupervised and self-supervised learning scenarios, where it outperforms traditional algorithms. These advantages have led to contrastive learning demonstrating outstanding application performance in a variety of fields, such as visual recognition,[52,53] natural language processing,[54,55] and sound analysis.[56,57]

This paper addresses the issue of multiaxial fatigue life prediction in materials by applying contrastive learning to effectively extract sample features across different multiaxial loading paths. The paper compares the feature representation performance of different model frameworks and tests various downstream task models. Compared to other clustering or dimensionality reduction algorithms, contrastive learning consistently achieves good prediction results, providing a new feature engineering strategy for multiaxial fatigue life prediction.

2. Data

2.1. Experimental data

In this study, 20 multiaxial fatigue experimental data of 316L stainless steel were used for training and validation of the contrastive learning model and downstream fatigue life prediction model. The experimental details can be found in the referenced published literature.[16,17] As shown in Figure 2, the experimental data included four typical uniaxial and multiaxial loading paths, each with five different amplitudes, with a stress ratio of $R = -1$. Table 1

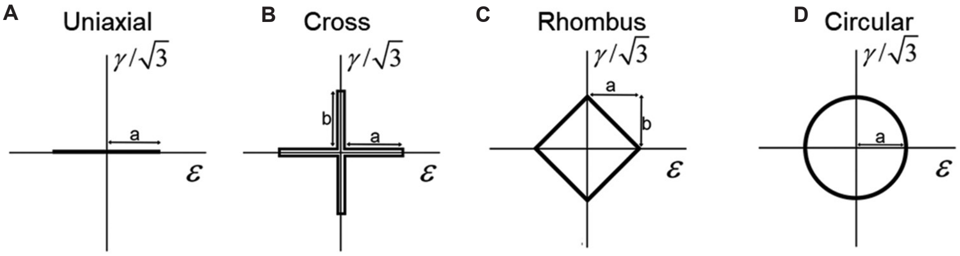


Figure 2. The diagram of multiaxial loading paths: (A) uniaxial, (B) cross, (C) rhombus, and (D) circular

Table 1. The detailed information of specimen for each loading path type

Loading path	ε_a	γ_a	N_f
Uniaxial	0.2	-	161000
	0.4	-	11865
	0.6	-	3339
	0.8	-	1719
	1.0	-	906
Cross	0.3	0.3	10023
	0.35	0.35	5831
	0.4	0.4	2038
	0.5	0.5	1378
	0.6	0.6	1311
Rhombus	0.3	0.3	7193
	0.35	0.35	4041
	0.4	0.4	3944
	0.5	0.5	1031
	0.6	0.6	795
Circular	0.3	0.3	5045
	0.35	0.35	2963
	0.4	0.4	1802
	0.5	0.5	744
	0.6	0.6	396

provides the amplitude and life information for each experimental specimen.

2.2. Data augmentation

To ensure the effectiveness of contrastive learning, data augmentation was also applied to enhance the training samples. The generative adversarial network (GAN)-based augmentation method used was derived from our previous research.[16] By integrating CNN with the Fourier transform, the proposed model achieves synergistic augmentation of stress-strain hysteresis loops and fatigue life prediction. Based on the data augmentation capability of this model, the 20 experimental samples were expanded to over 1000 samples for each loading path. For each multiaxial loading path, the strain amplitude of the augmented samples was distributed between the maximum and minimum values of the experimental data. Based on these augmented samples, 100 samples were uniformly selected for each loading path. A total of 400 augmented samples were selected for the training of the contrast learning model. Data augmentation effectively increases the training sample size and improves the performance of the contrastive learning model. In addition, only stress-strain hysteresis loops are required, without the need for fatigue life data, which reduces the difficulty of using augmented samples. In the subsequent downstream fatigue life prediction task, there are only the original 20 experimental datasets used for the training. In addition, in future work, constitutive material models or finite element simulations can also be employed as alternative methods for data augmentation.

2.3. Preprocessing

According to the previous introduction, the stress-strain hysteresis loop will serve as input features for the training of following contrastive learning and machine learning model. The output feature of the downstream supervised model is the logarithmic fatigue life. It can avoid the effect of its large magnitude range. Furthermore, all input and output features are normalized using the z-score method to ensure that the model captures the relative relationships between features rather than being influenced by their absolute values. The *z*-score normalization is described as Equation I:

$$\bar{x} = \frac{x - \mu_x}{\sigma_x} \tag{I}$$

Where $\bar{x}$ represents the normalized data, x represents the raw data, μ_x is the mean of the raw data, and σ_x represents the standard deviation of the entire sample space.

3. Algorithm

3.1. Contrastive learning architecture

Contrastive learning is a self-supervised learning strategy that enables the model to learn to distinguish between similar and dissimilar samples to extract features, without relying on manually labeled data[58] In the contrastive learning framework, a pair of samples (positive and negative pairs) is typically constructed, where positive pairs consist of similar or related samples and negative pairs consist of dissimilar samples. The model's objective is to minimize the distance between positive pairs and maximize the distance between negative pairs. This learning approach effectively utilizes unlabeled data and enhances the model's generalization ability on real-world data. Contrastive learning has been widely applied to various tasks, such as image recognition,[59,60] natural language processing,[61,62] and audio analysis,[63,64] demonstrating outstanding performance. Especially in the field of image processing, contrastive learning learns powerful visual feature representations by leveraging samples generated from different perspectives, sizes, or other transformations of images.

Among all contrastive learning models, SimCLR[65] learns representations by applying contrastive loss in the feature space to maximize the consistency between different augmented samples of the same data sample. The model framework, as shown in Figure 3, consists of four components: the data augmentation module, where each sample is randomly augmented to generate two related samples x_i and x_j, which are then passed through an encoder to extract high-dimensional features. After passing through a projection layer, the features are mapped to a lower-dimensional space, where the contrastive loss between the samples is computed to achieve maximum consistency.

In this work, contrastive learning was applied to the multiaxial fatigue life prediction of materials, using stress-strain hysteresis data as the input. Considering that contrastive learning can reduce the distance between similar samples in the feature space, this framework aims to extract general, discriminative features that are shared by samples with the same amplitude but different loading paths. These features were then used to construct a downstream fatigue life prediction model. The core idea of contrastive learning was implemented by constructing positive and negative sample pairs. The construction of these pairs was done by the data augmentation module, which randomly selected a small batch of data containing N samples. Negative samples were not specifically sampled, and after data augmentation, $2N$ data samples were generated. In this case, for each positive sample pair, there were $2(N–1)$ corresponding negative pairs. In this study, data augmentation involved adding Gaussian noise and applying time masking. Each sample was augmented with Gaussian noise, and with a 50% probability, time masking was applied, which randomly masked a certain number of time steps in the stress-strain data. After feature extraction through the encoder and projection layer, the distance between the features of positive and negative samples was computed. The loss function was minimized to learn a good feature space representation. Since SimCLR does not impose specific constraints on the network architecture, its performance improves after adding a projection layer. The model framework used in this study is shown in Figure 4, which adopts a synchronized symmetric architecture.

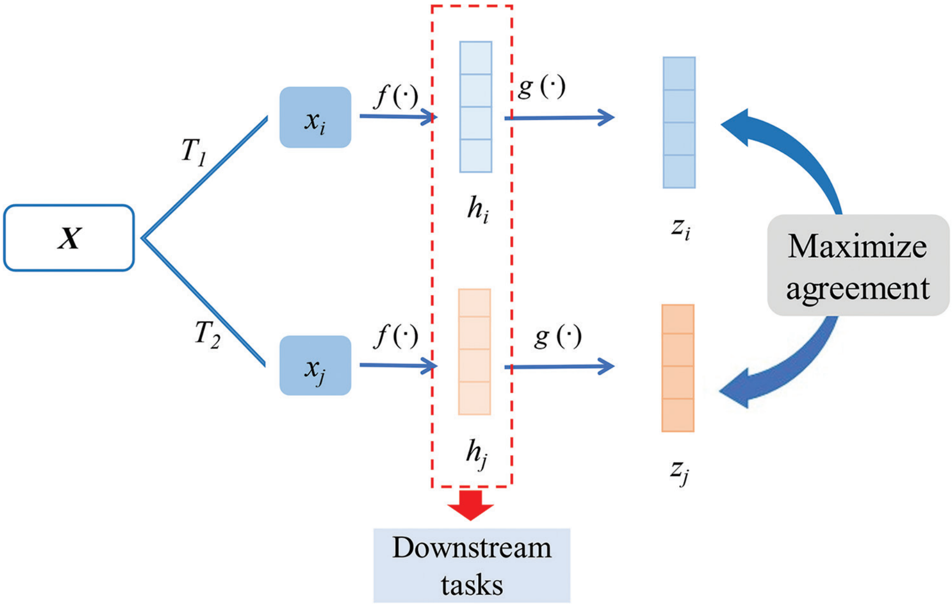


Figure 3. Schematic diagram of contrastive learning algorithm

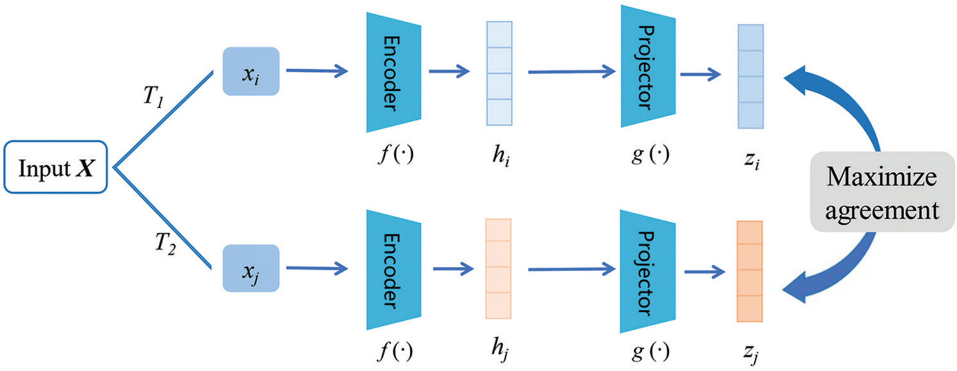


Figure 4. The framework of the proposed contrastive learning model

The core of contrastive learning is to optimize the model through a set of loss functions, enabling it to distinguish between similar and dissimilar samples. Below are some key formulas commonly used in contrastive learning. Among them, Equations II and IV are the formulas for the first sample generated by data augmentation, passed through the encoder and projection layer, respectively. Equation III and V are the formulas for the second sample x_j generated by data augmentation, after being processed by the encoder and projection layer. Equation VI is the formula for the calculation of the loss function. The Normalized Temperature-scaled Cross Entropy (NT-Xent) loss function enhances sample utilization and feature extraction quality during the learning process by effectively leveraging the relative information between samples. As a result, it has been widely applied in self-supervised learning tasks across various domains, including image processing,[66] text processing,[67] and more.T.

$$h_i = f\left(T_1\left(X\right);\Theta\right) = ReLU(\mathrm{W}_2^f\ f_1\left(\mathrm{T}_1\left(X\right);\ \Theta_1 + b_2^f\right) \quad \text{(II)}$$

$$h_i = f\left(T_2\left(X\right);\Theta\right) = ReLU(\mathrm{W}_2^f f_1\left(\mathrm{T}_2\left(X\right);\ \Theta_1 + b_2^f\right) \quad \text{(III)}$$

$$z_i = g\left(h_i\right) = W_2^g\ \sigma\left(W_1 h_i + b_1\right) + b_2 \quad \text{(IV)}$$

$$z_j = g\left(h_i\right) = W_2^g\ \sigma\left(W_1 h_j + b_1\right) + b_2 \quad \text{(V)}$$

$$l_{i,j} = -\log \frac{\exp(smi(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} 1[k \neq 1]\exp(smi(z_i, z_j)/\tau)} \quad \text{(VI)}$$

where $\Theta = \{\Theta_1, \Theta_2\}$ represents the set of all learnable parameters of the encoder layers; W^f, W^g and b^f, b^g represent the weight matrices and biases of the encoder and projection layers, respectively; $1[k \neq 1]$ is an indicator function, which takes the value 1 when $k \neq i$; and τ represents the temperature parameter.

3.2. Downstream life prediction model

After contrastive learning, deep feature representations can be extracted from the material stress-strain hysteresis loop, and the similarity between samples from different loading paths can be enhanced. This greatly benefits subsequent multiaxial fatigue life prediction tasks. As shown in Figure 5, after obtaining these features, this study uses them as input for various supervised learning models, such as linear regression, support vector machines (SVM), eXtreme Gradient Boosting (XGBoost), or Artificial Neural Network (ANN), for fatigue life prediction. This approach not only improves the predictive accuracy of the model but also enhances its generalization ability when confronted with unseen complex loading conditions.

3.3. Evaluation criteria

In this study, the root MSE (RMSE) was used to describe the deviation between the predicted logarithmic fatigue life values and the experimental logarithmic fatigue life, as shown in Equation VII:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{i,pre} - y_{i,exp})^2} \quad \text{(VII)}$$

where $y_{i,\mathrm{pre}}$ represents the model-predicted logarithmic fatigue life value and $y_{\mathrm{i,exp}}$ represents the experimental logarithmic fatigue life value. In addition, the model's prediction performance was also evaluated from other aspects, including the distribution of the predicted fatigue life values and the experimental values.

4. Results and discussion

In this section, the proposed contrastive learning framework was evaluated by comparing different network architectures to explore models with feature representation capabilities. It also compares these models with other unsupervised learning clustering models in terms of their ability to represent features effectively. Finally, based on the learned feature representations, a performance comparison is conducted for downstream life prediction tasks, examining the predictive effectiveness across various machine learning models.

4.1. Performance with different encoder network architectures

In this section, the experiment first determined two hyperparameters of the proposed contrastive learning framework: the number of layers in the encoder and the output feature dimension. The framework of the contrastive learning model is shown in Figure 6, where the encoder was set to two layers, and the output dimension was uniformly set to 128, which preliminarily validates the model's effectiveness. From a time-series perspective, the stress-strain hysteresis data recorded the change of stress and strain over time during one complete cycle. Therefore, to further determine which encoder structure was more suitable for the original data, four structures—One-Dimensional Convolutional Neural Network (1D-CNN), Two-Dimensional Convolutional Neural Network (2D-CNN), Gated Recurrent Unit (GRU), and ANN—were used. The high-dimensional features extracted by the encoder were passed through two fully connected layers and uniformly reduced to 64 dimensions, and training was conducted on the stress-strain hysteresis data. Table 2 provides the detailed hyperparameters of the contrastive learning models with different network frameworks.

In this study, to ensure the effectiveness of model training, the original dataset (with 20 samples in total) was randomly split into training and testing sets at a 6:4 ratio. Although the original samples had an equal number of samples for each type, random sampling without stratified control led to some distribution bias between different

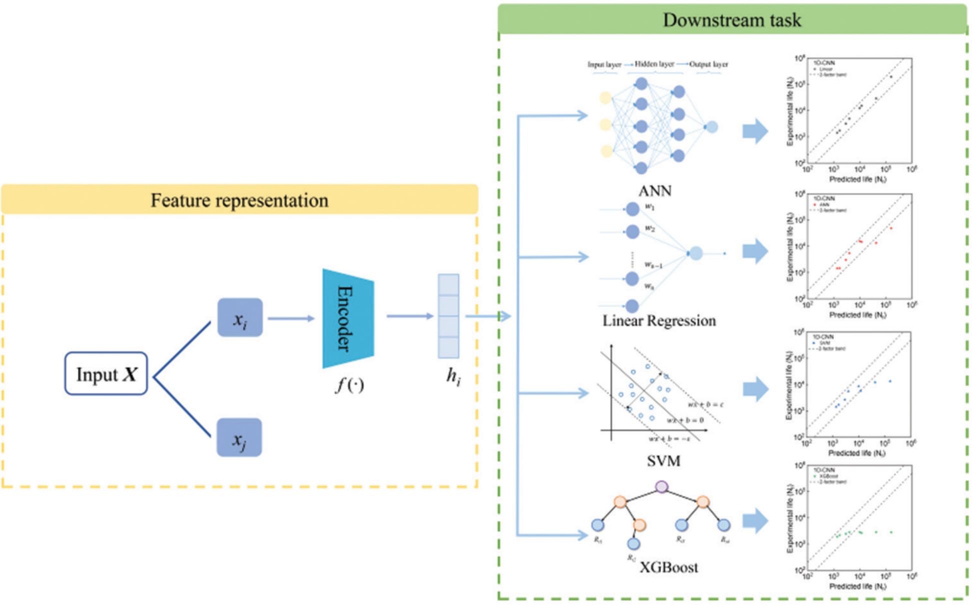


Figure 5. The utilization of deep features provided by contrastive learning model for downstream fatigue life prediction
Abbreviations: ANN: Artificial neural network; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

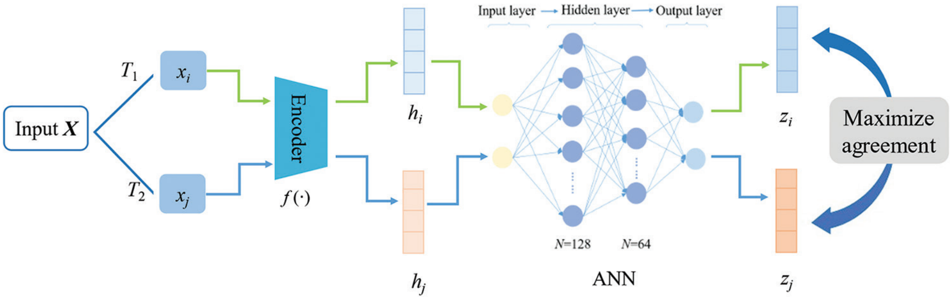


Figure 6. Hyperparameters and schematic diagrams of contrastive learning model with different network architectures
Abbreviation: ANN: Artificial neural network.

types of data in the training and testing sets, simulating the uneven distribution of data in real-world conditions. To further mitigate the impact of this distribution imbalance, additional sample data generated by a GAN model was included in the training set to augment the samples and improve the model's generalization ability. Each sample underwent two different data augmentation methods. The training loss of each contrastive learning model framework is shown in Figure 7. After 150 epochs, all models converged well, validating the effectiveness of the proposed framework.

Table 2. The detailed hyperparameters of contrastive learning models with different network architectures

Encoder	Projector	Hyperparameters			
		Configuration	Stride	Padding	τ
1D-CNN	ANN	Convolutional Layers: 2, Kernel Size: (3,3), Filters: 64, 128	1	1	0.3
2D-CNN		Convolutional Layers: 2, Kernel Size: (3, 3), Filters: 64, 128	1	1	
GRU		GRU Units: 128 Number of layers: 2	\	\	
ANN		Neurons: 256, 128	\	\	

Abbreviations: 1D-CNN: One-Dimensional Convolutional Neural Network; 2D-CNN: Two-Dimensional Convolutional Neural Network; ANN: Artificial Neural Network; GRU: Gated Recurrent Unit.

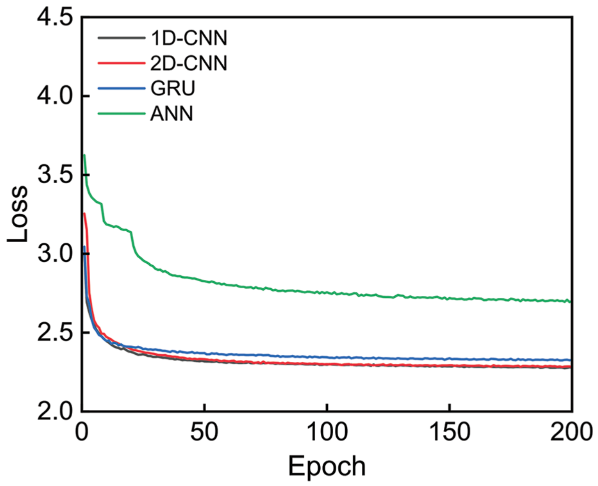


Figure 7. The training process of contrastive learning model between different architecture models

Abbreviations: 1D-CNN: One-dimensional convolutional neural network; 2D-CNN: Two-dimensional convolutional neural network; ANN: Artificial neural network; GRU: Gated recurrent unit.

To investigate whether the model successfully learned the distribution of data features during training, a visual analysis of the input and output features before and after training was performed. Specifically, in the visualization process, original data was used as the visualization samples, excluding the augmented samples generated by GAN. The t-distributed stochastic neighbor embedding (t-SNE)[68] method was employed for dimensionality reduction and visualization of the features, with the pre- and post-training visualization results shown in Figures 8 and 9. In these figures, data points with the same color represented features of samples with the same amplitude but different loading paths, while different colors corresponded to samples with different amplitudes. The visualization results indicated that in the original data, samples from the same loading path were clustered together in the feature space. However, for features generated by each encoder from the initial samples, the distribution of features for the same amplitude data was quite disordered, and no obvious clustering or separation of samples was observed in the reduced-dimensional space. It was worth noting that, although there was no clear separation of feature types in the visualization space, the learned features still showed significant application effects in downstream tasks. This may have been because t-SNE's low-dimensional visualization lost important information from the high-dimensional space, and the visualization results can not accurately reflect the intrinsic structure of the high-dimensional feature space.

4.2. The effectiveness of downstream fatigue life prediction

To further find the optimal downstream regression model, the linear regression model was considered, as it is simple,

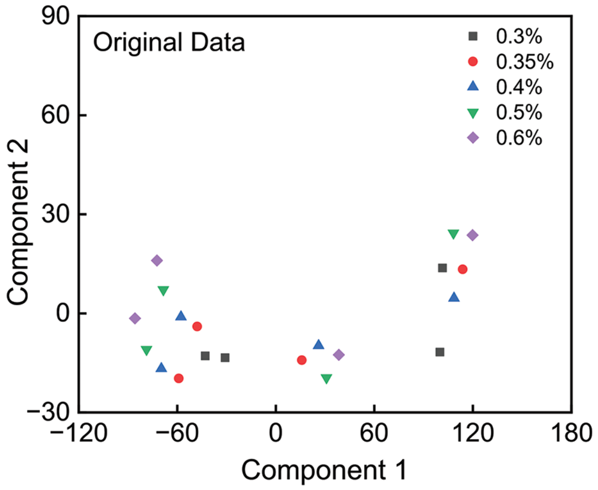


Figure 8. Visualization results of the original data

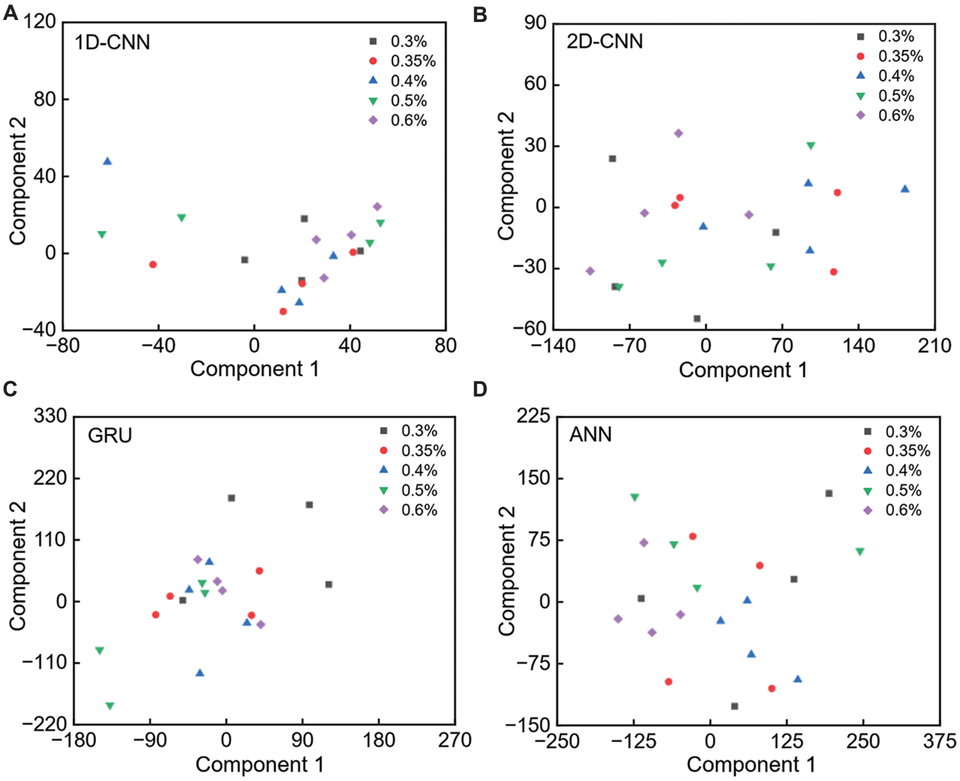


Figure 9. Visualization results of the feature representations obtained through contrastive learning with different encoder architectures: (A) 1D-CNN, (B) 2D-CNN, (C) GRU, and (D) ANN.

Abbreviations: 1D-CNN: One-dimensional convolutional neural network; 2D-CNN: Two-dimensional convolutional neural network; ANN: Artificial neural network; GRU: Gated recurrent unit.

efficient to train, and suitable as a baseline model to provide a reference for more complex models. In addition, since the downstream training set only contained a small amount of data randomly split 60:40 from the original dataset, and given the robust performance of SVM and XGBoost in small sample scenarios, as well as their capability in handling nonlinear problems, three models—linear regression, SVM, and XGBoost—were chosen for training. As a widely used machine learning model, the ANN model was also selected as one of the models for downstream tasks. The features learned by contrastive learning from different network architectures were used as input for the downstream models, and the models were trained. The RMSE of the experimental results on the test set is shown in Figure 10. In the figure, the *x*-axis represents the contrastive learning models with different network architectures, and each bar color represents a different downstream model. The *y*-axis represents the RMSE value, with lower RMSE indicating better prediction performance. The results showed that, for all contrastive learning models regardless of the framework, the extracted features had relatively lower RMSE values on the linear regression model in the downstream task. For all downstream linear regression models, whether using 1D-CNN or 2D-CNN frameworks in the contrastive learning model, the extracted features achieved lower RMSE values. In particular, the contrastive learning model with 1D-CNN as the encoder achieved the lowest RMSE value on the downstream task. This suggests that the features extracted by 1D-CNN are sufficiently simple, have high linear separability, and can be effectively utilized by linear models.

In addition, the predicted fatigue life on the test set and the experimental fatigue life are shown in Figure 11. It can be observed that the best performance was achieved when 1D-CNN was used as the encoder. In contrast to 1D-CNN, the contrastive learning model with 2D-CNN as the encoder had one test point lying outside the 2-factor band of the linear regression model. However, for the XGBoost model, the features extracted by contrastive learning performed the worst on the downstream model, with the RMSE significantly higher than that of other models. This might be because the XGBoost model was more suited to handle high-dimensional complex features, and it was unable to fully leverage the advantages of the features learned by contrastive learning. In addition, during computation, it might have introduced extra noise or information loss, severely impacting fatigue life prediction performance.

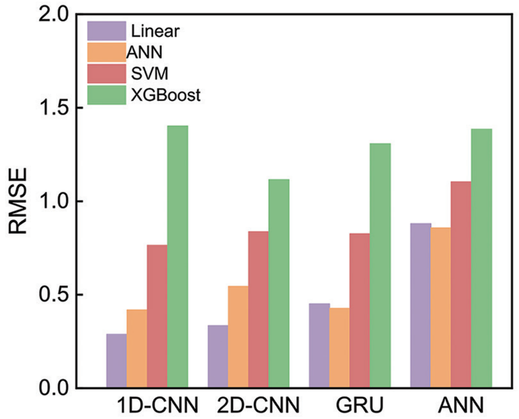


Figure 10. The RMSE performances of life prediction models.
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; 2D-CNN: Two-dimensional convolutional neural network; ANN: Artificial neural network; GRU: Gated recurrent unit; Linear: Linear regression; RMSE: Root mean squared error; SVM: Support vector machine; XGBoost: eXtreme Gradient boosting.

Based on these results, CNN1D was selected as the encoder for the contrastive learning framework, and the linear regression layer was used as the downstream regression model for fatigue life prediction. This combination not only achieved the lowest RMSE value but also provided acceptable prediction results on the test set, fully validating the superiority of this combination.

To further validate the effectiveness of data augmentation in contrastive learning on the training results, the CNN1D-based contrastive learning encoder was chosen, and the performance of the downstream model was compared under two conditions: with and without data augmentation. The training loss is shown in Figure 12. It can be observed that with data augmentation, the model achieved a sufficiently small loss after fewer epochs and began to converge quickly. In contrast, the model without data augmentation had a relatively large initial loss and required more epochs to converge.

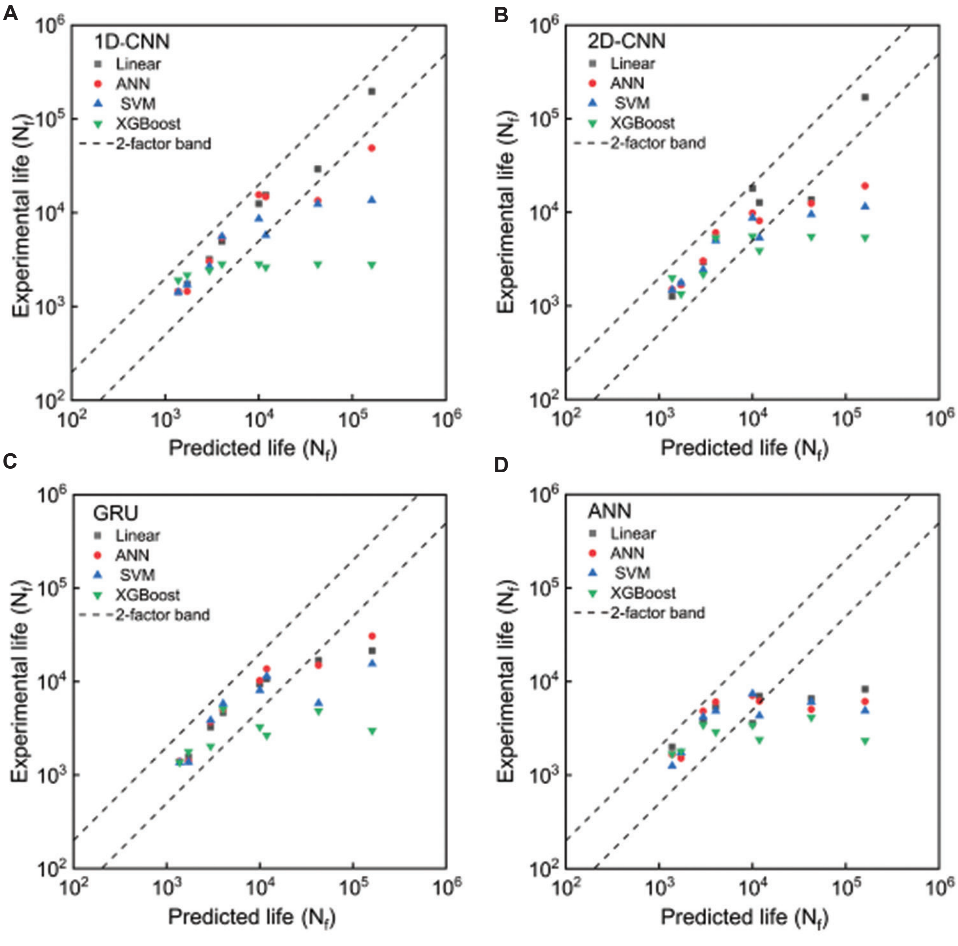


Figure 11. The detailed predicted results of contrastive learning and downstream models: (A) 1D-CNN, (B) 2D-CNN, (C) GRU, and (D) ANN.
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; 2D-CNN: Two-dimensional convolutional neural network; ANN: Artificial neural network; GRU: Gated recurrent unit; Linear: Linear regression; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

The predicted fatigue life on the test set and the comparison with experimental results, as well as the RMSE, are shown in Figures 13 and 14. From the figures, the results indicated that while the model without data augmentation had only two data points outside the 2-factor band, the model with data augmentation produced better prediction results and had a lower RMSE. Specifically, the RMSE for the model with data augmentation was reduced by 16.35% compared to the model without data augmentation.

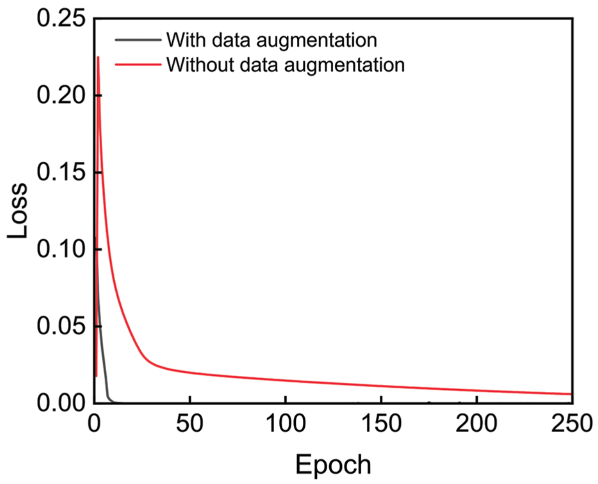


Figure 12. The evolution of loss function during training loss of contrastive learning models with and without data augmentation

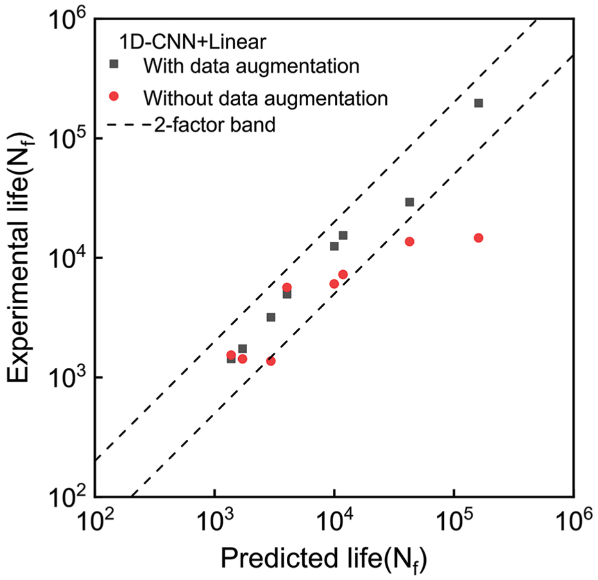


Figure 13. The predicted results of 1D-CNN contrastive learning encoder and linear regression downstream model with and without data augmentation.
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; Linear: Linear regression.

In addition, to investigate the effectiveness of the contrastive learning framework, this experiment explored the scenario where no contrastive learning was used, and the stress-strain data was directly input into the downstream regression model, with data augmentation applied. The data augmentation mainly involved using GAN-generated data to expand the dataset, which, especially when the sample size was small, can increase the diversity and generalization ability of the data, thereby enhancing the model's predictive performance. The choice of downstream model was consistent with previous experiments, and the model's performance was compared using RMSE and prediction results, as shown in Figure 15.

From the experimental results, it was be observed that when contrastive learning was not used but data augmentation was applied, the performance of the four models was poor. The minimum RMSE value was close to 0.7, and in the comparison of the model's predictions with experimental results, the best performance still had four data points outside the 2-factor band. Whether in terms of RMSE or the comparison between predicted and experimental values, the model performance was worse than that of the model using deep features extracted by the contrastive learning framework. Especially for the linear regression model, it showed the best performance when contrastive learning was applied, while its performance was the worst when contrastive learning was not used.

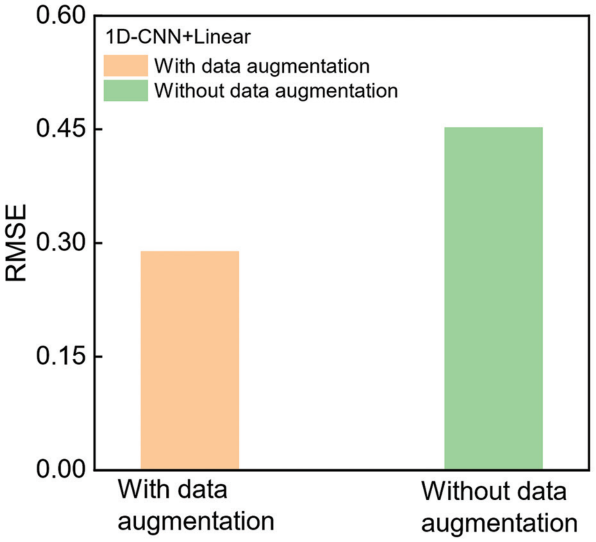


Figure 14. The RMSE performances of 1D-CNN contrastive learning encoder and linear regression downstream model with and without data augmentation.
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; Linear: Linear regression; RMSE: Root mean squared error.

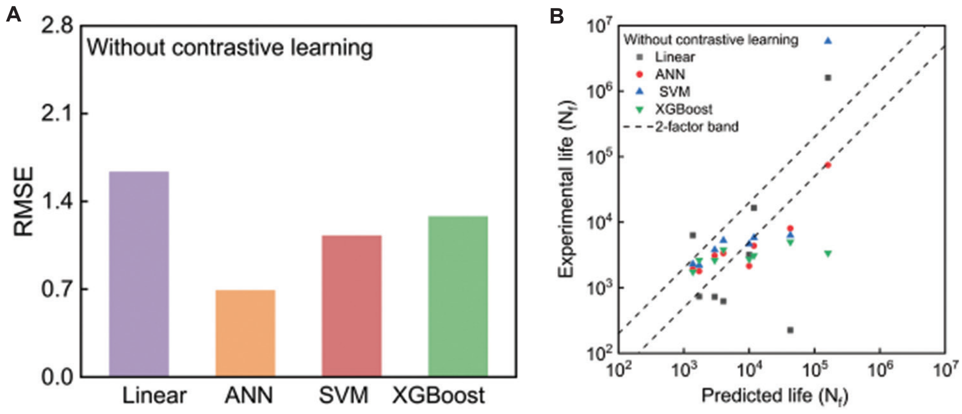


Figure 15. The performance of the downstream model in the case of data augmentation without contrastive learning. (A) The RMSE of the downstream models. (B) The prediction results of the downstream model.

Abbreviations: ANN: Artificial neural network; Linear: Linear regression; RMSE: Root mean squared error; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

This indicated that although data augmentation can provide more training data, the augmented data might only be generated based on surface-level features. The four downstream models were unable to solely rely on augmented data to understand the underlying structure and complex patterns in the data. In contrast, the contrastive learning model, by maximizing the similarity between similar samples and maximizing the distance between dissimilar samples, continuously optimized the representation space of the data. The large amount of data provided by augmentation can help the model learn in a broader sample space, enabling the model to learn more universal and representative features. Therefore, although pure data augmentation did not provide sufficient structural information, the training strategy of contrastive learning allowed the model to better uncover the inherent relationships within the data, thereby improving the performance of the downstream model.

To further verify the superiority of the combination of contrastive learning framework and data augmentation, the performance of the downstream regression model was investigated in this experiment without the contrastive learning framework and data augmentation. The input to the downstream model did not undergo any form of data augmentation, nor was it trained with features learned through the contrastive learning framework. Instead, the raw stress-strain data was used as input, and the logarithmic fatigue life was used as the output. As a result, the model's performance was directly constrained by the data volume and sample diversity. The model's RMSE and prediction results were compared with the experimental results, as shown in Figure 16.

From the figure, it was evident that the RMSE of the models was above 0.5, and in the comparison between predicted and experimental fatigue life, although Linear Regression models only had one point outside the 2-factor band, most of the data within the error margin was distributed near its edges. This indicated that while the models could, to some extent, capture some patterns in the data through simple training methods, the performance remained limited, and the internal relationships were not fully exploited. In contrast, under the contrastive learning framework, the linear regression model had the smallest RMSE and its predicted values were well-distributed along the diagonal, outperforming all other downstream models.

When compared with the second experiment, despite the absence of the contrastive learning framework, the model still relied on the limited information from the raw data for training. The model depended on the data's quality and complexity to learn some effective features. This proved that simply relying on data augmentation did not necessarily contribute positively to model performance. Although data augmentation could increase the training sample size, the augmented data did not add meaningful information. Without an effective training strategy, the augmented data could introduce significant noise and negatively affect the model's performance, diminishing the effectiveness of data augmentation.

Through these three experiments, the effects of data augmentation, contrastive learning framework, and their combination on downstream models were explored. The experimental results not only showed the effects of each factor individually but also demonstrated the synergistic effect when they were combined. Ultimately, the experiments confirmed the superiority of the combination of contrastive learning framework and data augmentation. While data augmentation could effectively increase the sample size, it might introduce noise and did not necessarily contribute positively to the model's training process, and

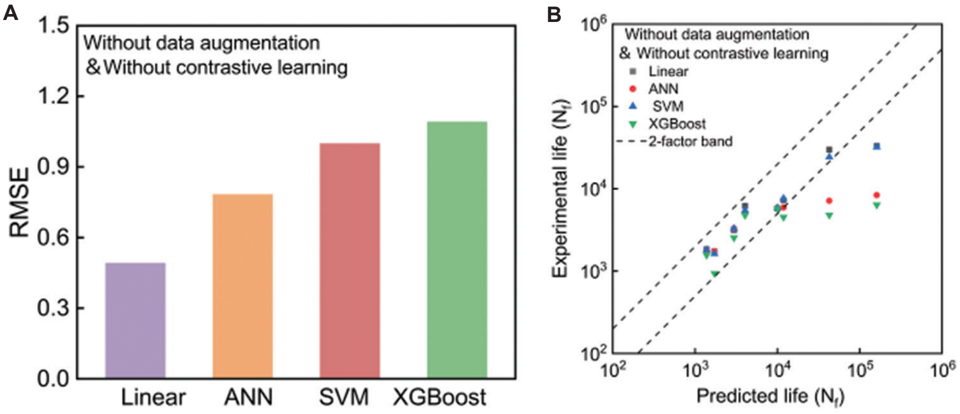


Figure 16. The performance of the downstream models without data augmentation without contrastive learning. (A) The RMSE of the models. (B) The prediction results of the models.

Abbreviations: ANN: Artificial neural network; Linear: Linear regression; RMSE: Root mean squared error; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

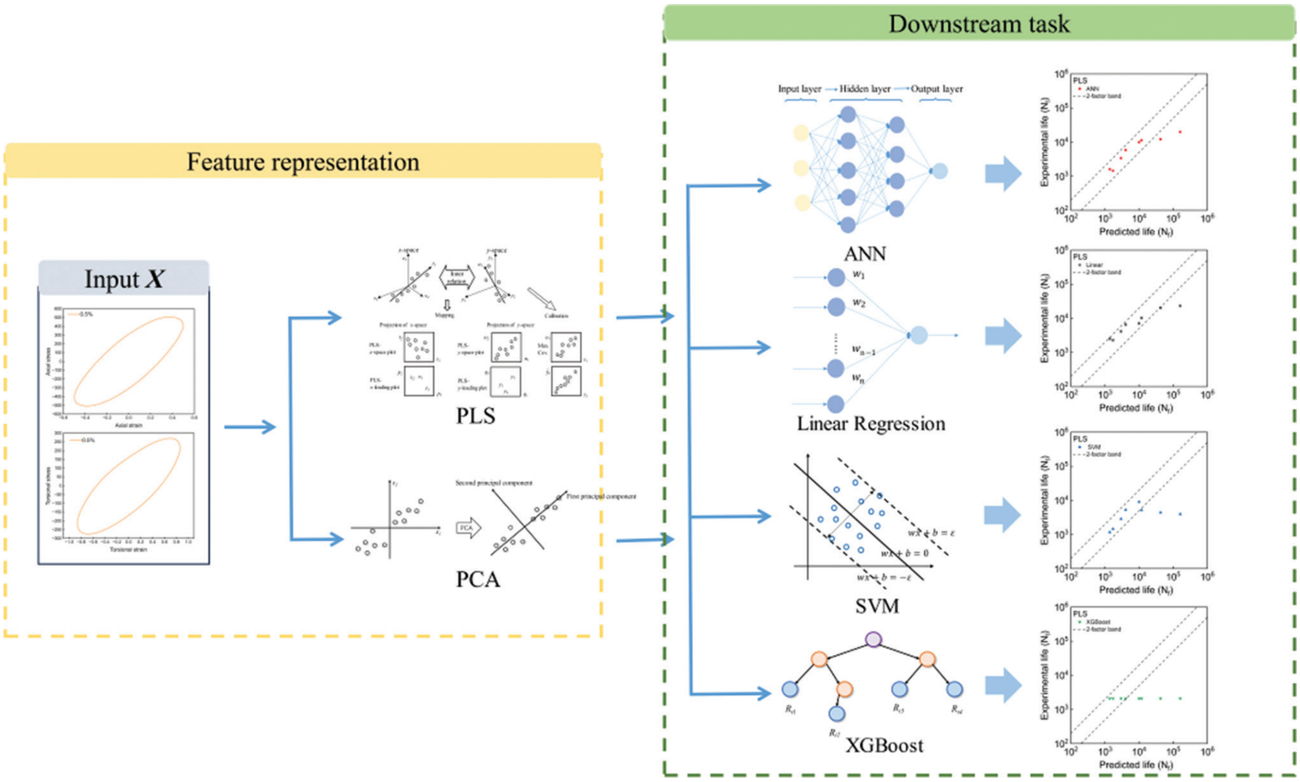


Figure 17. Schematic diagrams of classical unsupervised clustering learning algorithms

Abbreviations: ANN: Artificial neural network; PCA: Principal component analysis; PLS: Partial least squares; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

might even interfere with the model's training. In contrast, the combination of data augmentation and the contrastive learning framework's training strategy maximized the utilization of data samples, extracted deep features from the stress-strain data, and applied them to the downstream model training, achieving the best prediction results.

4.3. Comparison of the effectiveness between different clustering methods

In this section, two unsupervised learning algorithms, partial least squares (PLS) and principal component analysis (PCA), were applied to perform dimensionality reduction on the

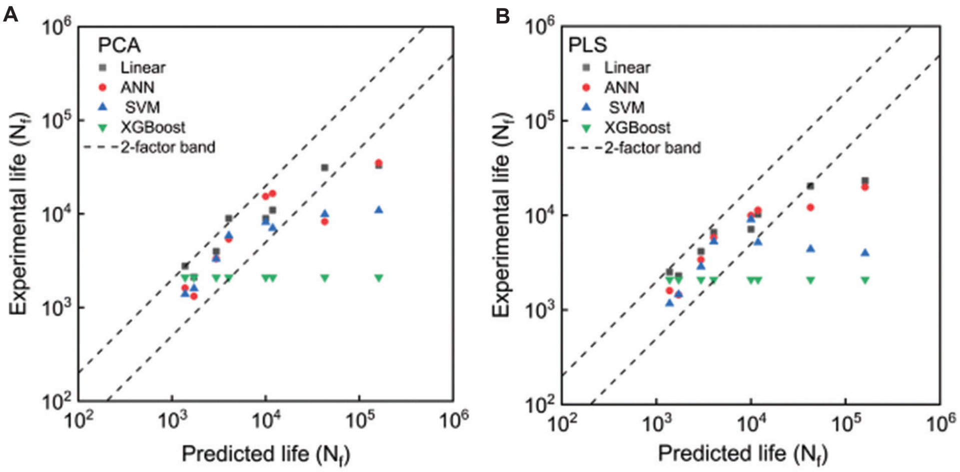


Figure 18. Downstream prediction results of other unsupervised algorithms: (A) PCA, and (B) PLS
Abbreviations: ANN: Artificial neural network; Linear: Linear regression; PCA: Principal component analysis; PLS: Partial least squares; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

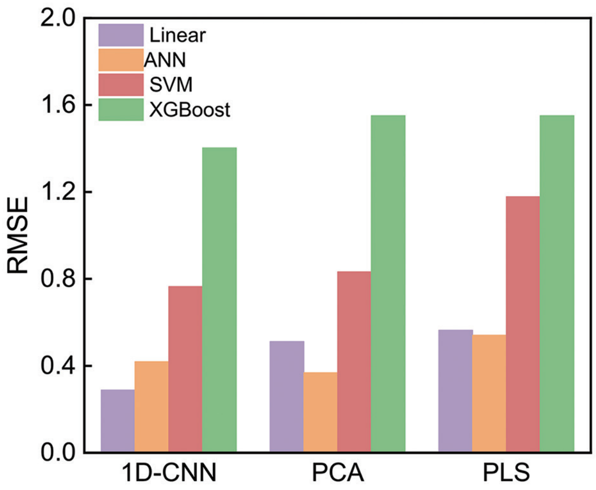


Figure 19. Comparison of downstream RMSE between other unsupervised learning algorithms and contrastive learning
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; ANN: Artificial neural network; Linear: Linear regression; RMSE: Root mean squared error; SVM: Support vector machine; XGBoost: eXtreme gradient boosting.

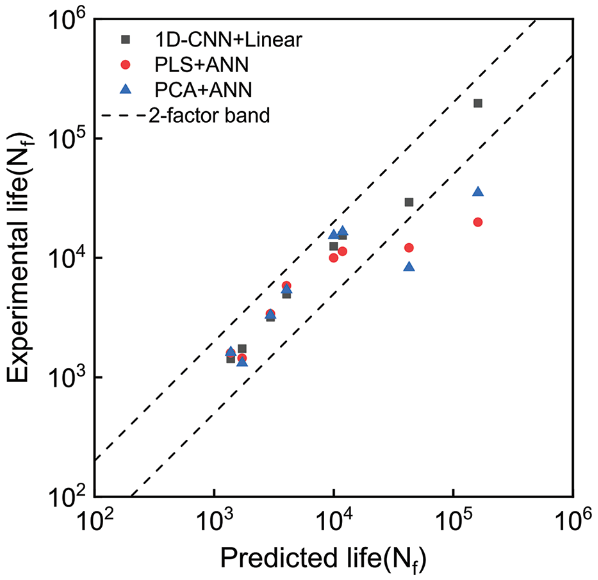


Figure 20. Comparison with the optimal performance prediction results of the contrastive learning framework
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; ANN: Artificial neural network; Linear: Linear regression; PCA: Principal component analysis; PLS: Partial least squares.

original data. The reduced features were then fed into four regression models for training. The overall process of the unsupervised learning algorithms is illustrated in Figure 17. Using these two methods, the original data was reduced to a certain dimension, and the extracted features were directly applied to the construction of regression models. ANN, linear regression, SVM, and XGBoost were still used as the regression models. The comparison between the predicted results from the test set and the experimental values is shown in Figure 18, and the RMSE is shown in Figure 19. From the results, it can be seen that the features reduced by both PLS and PCA performed relatively well on linear models and ANN, with two points falling outside the 2-factor band. However, even the most optimal PCA and PLS frameworks still did not outperform the performance of contrastive learning, where the contrastive learning framework remained the best, as shown in Figure 20. This indicated that, compared to the features extracted by unsupervised learning algorithms, the contrastive learning framework can improve prediction performance to some extent.

Table 3. The detailed number of each specimen and defects for each type

Feature extraction method	RMSE of the downstream prediction model			
	Linear	ANN	SVM	XGBoost
1D-CNN	**0.2889**	0.41946	0.76503	1.40253
2D-CNN	**0.33526**	0.54549	0.8378	1.11583
GRU	0.45183	**0.42774**	0.82675	1.30814
ANN	0.88021	**0.85771**	1.10393	1.3852
PLS	0.56382	**0.54084**	1.17852	1.55079
PCA	0.51162	**0.36833**	0.83287	1.55079

Note: The values in boldface represent the lowest RMSE among the four downstream models under the same conditions.
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; 2D-CNN: Two-dimensional convolutional neural network; ANN: Artificial neural network; GRU: Gated recurrent unit; Linear: Linear regression; PCA: Principal component analysis; PLS: Partial least squares; RMSE: Root mean squared error; SVM: Support Vector Machines.

Table 4. RMSE results of ablation experiments on 1D-CNN with contrastive learning

Ablation setting	RMSE of the downstream prediction model			
	Linear	ANN	SVM	XGBoost
Without data augmentation	**0.45243**	-	-	-
Without contrastive learning	1.63522	**0.69097**	1.12755	1.28143
Without data augmentation & without contrastive learning	**0.49226**	0.78342	0.50171	1.09144
With data augmentation & with contrastive learning	**0.2889**	0.41946	0.76503	1.40253

Note: The values in boldface represent the lowest RMSE among the four downstream models under the same conditions.
Abbreviations: 1D-CNN: One-dimensional convolutional neural network; ANN: Artificial neural network; Linear: Linear regression; RMSE: Root mean squared error; SVM: Support vector machines; XGBoost: eXtreme gradient boosting.

Table 3 summarizes the RMSE values of different feature extraction methods across various downstream prediction models. It was observed that the features extracted by the contrastive learning framework with 1D-CNN as the encoder achieved the best performance on the downstream linear regression model. To further validate the contributions of data augmentation and contrastive learning, ablation experiments were conducted. Table 4 presents the results of removing data augmentation, removing contrastive learning, and removing both, illustrating their impact on model performance.

5. Conclusion

In this study, a multiaxial fatigue hysteresis feature extraction method based on contrastive learning is proposed. This method effectively extracts deep feature representations from complex multiaxial fatigue stress-strain responses, which are then utilized in downstream fatigue life prediction tasks to enhance prediction accuracy. The specific conclusions are as follows:

(i) Compared to network architectures such as ANN, GRU, and 2D-CNN, the 1D-CNN network achieves the best contrastive learning performance and is more stable during training. The deep feature representations extracted through contrastive learning, when visualized by t-SNE, show a chaotic distribution in the reduced-dimensional space, with no obvious clustering or separation of categories. However, the extracted features enable the downstream fatigue life prediction model to more easily learn from samples with different loading paths, achieving excellent performance even with a simple linear regression model.

(ii) Compared to other unsupervised learning algorithms, the features extracted using contrastive learning show better similarity and consistency. Through comparative experiments, contrastive learning is found to be more effective in extracting features related to fatigue life prediction from stress-strain hysteresis data, thereby helping downstream models better uncover the underlying patterns in the data. Compared to traditional unsupervised learning algorithms, contrastive learning demonstrates stronger robustness and effectiveness when handling multiaxial fatigue data.

(iii) The feature representations learned through contrastive learning exhibit superior predictive performance in downstream tasks. In multiple machine learning models, contrastive learning consistently achieves better prediction results. Compared to scenarios without contrastive learning, the maximum reduction in RMSE in models such as SVM, ANN, and others can reach 86.26%. In addition, the prediction stability is improved, as evidenced by a reduction in the standard deviation of repeated experiments.

(iv) Contrastive learning has the potential to be further extended for applications in multiaxial fatigue life prediction and similar domains. Leveraging the benefits of contrastive learning, it can help achieve few-shot or even zero-shot learning for downstream tasks. This approach can also contribute to addressing challenges in fields such as electronic packaging and multiscale structural integrity, where data scarcity and the need for robust predictive models are key concerns.

(v) To further enhance the effectiveness and applicability of the proposed framework, several key directions warrant exploration. One important area is integrating contrastive learning with traditional physics-based models to bridge data-driven insights with mechanical principles, thereby improving model interpretability. In addition, extending this framework to materials with heterogeneous microstructures can help evaluate its performance under different fatigue mechanisms and further validate its generalization capability.

Acknowledgments

None.

Funding

The authors gratefully acknowledge financial support for this work from the National Natural Science Foundation of China (No. 12302098).

Conflict of interest

The authors declared that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Author contributions

Conceptualization: Xingyue Sun
Formal analysis: Ziyu Cui
Investigation: Xingyue Sun, Ziyu Cui
Methodology: Xingyue Sun
Writing – original draft: Ziyu Cui
Writing – review & editing: Xingyue Sun, Xu Chen

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Availability of data

The data are available from the corresponding author upon reasonable request.

References

1. Towashiraporn P, Gall K, Subbarayan G, *et al*. Power cycling thermal fatigue of Sn-Pb solder joints on a chip scale package. *Int J Fatigue*. 2024;26(5):497-510.

   doi: 10.1016/j.ijfatigue.2003.09.004

2. Donnerbauer K, Bill T, Starke P, *et al*. Fatigue life evaluation of metastable austenitic stainless steel AISI347 based on nondestructive testing methods for different environmental conditions. *Int J Fatigue*. 2024;179:108056.

   doi: 10.1016/j.ijfatigue.2023.108056

3. Zhao B, Xie L, Song J, *et al*. Fatigue life prediction of aero-engine compressor disk based on a new stress field intensity approach. *Int J Mech Sci*. 2020;165:105190.

   doi: 10.1016/j.ijmecsci.2019.105190

4. Liang Q, Peng C, Li X. A multi-state Semi-Markov model for nuclear power plants piping systems subject to fatigue damage and random shocks under dynamic environments. *Int J Fatigue*. 2023;168:107448.

   doi: 10.1016/j.ijfatigue.2022.107448

5. Yu Z, Sun X, Xing R, Chen X. Unified prediction of uniaxial ratcheting deformation at elevated temperatures with physics-informed multimodal network. *Int J Plast*. 2025;187:104275.

   doi: 10.1016/j.ijplas.2025.104275

6. Maniar Y, Konstantin G, Sharma A, *et al*. Solder joint lifetime modeling under random vibrational load collectives. *JOM*. 2020;72(2):898-905.

   doi: 10.1007/s11837-019-03947-1

7. Wijker JJ. *Spacecraft Structures*. Berlin: Springer Science & Business Media; 2008.

8. Min KD, Lee BS, Kim SJ. Effects of oxide on fatigue crack growth behaviour of type 347 stainless steel in PWR water conditions. *Fatigue Fract Eng Mater Struct*. 2015;38(8):960-969.

   doi: 10.1111/ffe.12290

9. Agency IAE. *Assessment and Management of Ageing of Major Nuclear Power Plant Components Important to Safety: PWR Pressure*, IAEA-TECDOC-1556, IAEA, Vienna; 2007.

10. Guo C, Yu D, Sun X, *et al*. Fatigue failure mechanism and life prediction of a cast duplex stainless steel after thermal aging. *Int J Fatigue*. 2021;146:106161.

    doi: 10.1016/j.ijfatigue.2021.106161

11. Kishore P, Mondal A, Trivedi A, *et al*. A microstructure sensitive machine learning-based approach for predicting fatigue life of additively manufactured parts. *Int J Fatigue*. 2025;192:108724.

    doi: 10.1016/j.ijfatigue.2024.108724

12. Yang J, Kang G, Liu Y, Kan Q. A novel method of multiaxial fatigue life prediction based on deep learning. *Int J Fatigue*. 2021;151:106356.

    doi: 10.1016/j.ijfatigue.2021.106356

13. Zhang M, Sun CN, Zhang X, *et al*. High cycle fatigue life prediction of laser additive manufactured stainless steel: A machine learning approach. *Int J Fatigue*. 2019;128:105194.

    doi: 10.1016/j.ijfatigue.2019.105194

14. Li J, Yang Z, Qian G, Berto F. Machine learning based very-high-cycle fatigue life prediction of Ti-6Al-4V alloy fabricated by selective laser melting. *Int J Fatigue*. 2022;158:106764.

    doi: 10.1016/j.ijfatigue.2022.106764

15. Jing G, Ma T, Wang Z, *et al*. Physical hierarchical neural

network for low cycle fatigue life prediction of compacted graphite cast iron based on small data. *Int J Fatigue*. 2024;188:108509.

doi: 10.1016/j.ijfatigue.2024.108509

16. Sun X, Zhou K, Shi S, Song K, Chen X. A new cyclical generative adversarial network based data augmentation method for multiaxial fatigue life prediction. *Int J Fatigue*. 2022;162:106996.

    doi: 10.1016/j.ijfatigue.2022.106996

17. Sun X, Zhou T, Song K, Chen X. An image recognition based multiaxial low-cycle fatigue life prediction method with CNN model. *Int J Fatigue*. 2023;167:107324.

    doi: 10.1016/j.ijfatigue.2022.107324

18. Zhou T, Sun X, Yu Z, Chen X. A generalization ability-enhanced image recognition based multiaxial fatigue life prediction method for complex loading conditions. *Eng Fract Mech*. 2024;295:109802.

    doi: 10.1016/j.engfracmech.2023.109802

19. Wang H, Zhang J, Li B, *et al*. Machine learning-based fatigue life prediction of laser powder bed fusion additively manufactured Hastelloy X via nondestructively detected defects. *Int J Struct Integr*. 2024;16(1):104-126.

    doi: 10.1108/IJSI-09-2024-0161

20. Zhao F, Cui J, Yuan M, Zhao J. A weakly supervised pairwise comparison learning approach for bearing health quantitative evaluation and remaining useful life prediction. *Eng Computat*. 2023;40(7/8):1593-1616.

    doi: 10.1108/EC-12-2022-0747

21. Mao M, Wang W, Lu C, Jia F, Long X. Machine learning for board-level drop response of BGA packaging structure. *Microelectron Reliabil*. 2022;134:114553.

    doi: 10.1016/j.microrel.2022.114553

22. Long X, Lu C, Shen Z, Su Y. Identification of Mechanical Properties of Thin-Film Elastoplastic Materials by Machine Learning. *Acta Mechanica Solida Sinica*. 2023;36(1):13-21.

    doi: 10.1007/s10338-022-00340-5

23. Long X, Mao M, Lu C, Li C, Li R. Modeling of heterogeneous materials at high strain rates with machine learning algorithms trained by finite element simulations. *J Micromech Mol Phys*. 2021;6(1):2150001.

    doi: 10.1142/S2424913021500016

24. Cao W, Sun X, Li Y, *et al*. Multiaxial damage parameter evaluation by neural network-based symbolic regression. *Eng Fract Mechan*. 2025;315:110809.

    doi: 10.1016/j.engfracmech.2025.110809

25. Jiang L, Hu Y, Liu Y, *et al*. Physics-informed machine learning for low-cycle fatigue life prediction of 316 stainless steels. *Int J Fatigue*. 2024;182:108187.

    doi: 10.1016/j.ijfatigue.2024.108187

26. Zhu S, Zhang Y, Zhu B, *et al*. High cycle fatigue life prediction of titanium alloys based on a novel deep learning approach. *Int J Fatigue*. 2024;182:108206.

    doi: 10.1016/j.ijfatigue.2024.108206

27. Liao H, Pan J, Su X, Sun X, Chen X. A path-dependent adaptive physics-informed neural network for multiaxial fatigue life prediction. *Int J Fatigue*. 2025;193:108799.

    doi: 10.1016/j.ijfatigue.2024.108799

28. Chen S, Zhou X, Bai Y. A frequency domain enhanced multi-view neural network approach to multiaxial fatigue life prediction for various metal materials. *Int J Fatigue*. 2025;190:108620.

    doi: 10.1016/j.ijfatigue.2024.108620

29. Zhang P, Tang K, Wang A, Wu H, Zhong Z. Neural network integrated with symbolic regression for multiaxial fatigue life prediction. *Int J Fatigue*. 2024;188:108535.

    doi: 10.1016/j.ijfatigue.2024.108535

30. Long X, Li H, Iyela PM, Kang SB. Predicting the bond stress-slip behavior of steel reinforcement in concrete under static and dynamic loadings by finite element, deep learning and analytical methods. *Eng Fail Anal*. 2024;161:108312.

    doi: 10.1016/j.engfailanal.2024.108312

31. Zhou K, Sun X, Shi S, Song K, Chen X. Machine learning-based genetic feature identification and fatigue life prediction. *Fatigue Fract Eng Mater Struct*. 2021;44(9):2524-2537.

    doi: 10.1111/ffe.13532

32. Zhou T, Sun X, Chen X. A multiaxial low-cycle fatigue prediction method under irregular loading by ANN model with knowledge-based features. *Int J Fatigue*. 2023;176:107868.

    doi: 10.1016/j.ijfatigue.2023.107868

33. Long X, Ding X, Li J, Dong R, Su Y, Chang C. Indentation reverse algorithm of mechanical response for elastoplastic coatings based on LSTM deep learning. *Materials*. 2023;16(7):2617.

    doi: 10.3390/ma16072617

34. Wang CH, Brown MW. A path-independent parameter for fatigue under proportional and non-proportional loading. *Fatigue Fract Eng Mater Struct*. 1993;16(12):1285-1297.

    doi: 10.1111/j.1460-2695.1993.tb00739.x

35. Fatemi A, Socie DF. A critical plane approach to multiaxial fatigue damage including out-of-phase loading. *Fatigue Fract Eng Mater Struct*. 1988;11(3):149-165.

    doi: 10.1111/j.1460-2695.1988.tb01169.x

36. Zhan Z, Hu W, Li B, Zhang Y, Meng Q, Guan Z. Continuum damage mechanics combined with the extended finite element method for the total life prediction of a metallic

component. *Int J Mech Sci*. 2017;124-125:48-58.

doi: 10.1016/j.ijmecsci.2017.03.002

37. Yang S, Hu W, Meng Q, Zhao B. A new continuum damage mechanics-based two-scale model for high-cycle fatigue life prediction considering the two-segment characteristic in S-N curves. *Fatigue Fract Eng Mater Struct*. 2020;43(2):387-402.

    doi: 10.1111/ffe.13161

38. Brown MW, Miller KJ. A theory for fatigue failure under multiaxial stress-strain conditions. *Proc Inst Mech Eng*. 1973;187(1):745-755.

    doi: 10.1243/pime_proc_1973_187_161_02

39. Carraro PA, Quaresimin M. A damage based model for crack initiation in unidirectional composites under multiaxial cyclic loading. *Compos Sci Technol*. 2014;99:154-163.

    doi: 10.1016/j.compscitech.2014.05.012

40. Wang L, Zhu S, Luo C. Physics-guided machine learning frameworks for fatigue life prediction of AM materials. *Int J Fatigue*. 2023;172:107658.

    doi: 10.1016/j.ijfatigue.2023.107658

41. Dong Y, Yang X, Chang D, Li Q. Predicting fatigue life of multi-defect materials using the fracture mechanics-based physics-informed neural network framework. *Int J Fatigue*. 2025;190:108626.

    doi: 10.1016/j.ijfatigue.2024.108626

42. Fan JL, Zhu G, Zhu ML, Xuan FZ. A data-physics integrated approach to life prediction in very high cycle fatigue regime. *Int J Fatigue*. 2023;176:107917.

    doi: 10.1016/j.ijfatigue.2023.107917

43. Gan L, Wu H, Zhong Z. On the use of data-driven machine learning for remaining life estimation of metallic materials based on Ye-Wang damage theory. *Int J Fatigue*. 2022;156:106666.

    doi: 10.1016/j.ijfatigue.2021.106666

44. Wang H, Li B, Xuan FZ. Fatigue-life prediction of additively manufactured metals by continuous damage mechanics (CDM)-informed machine learning with sensitive features. *Int J Fatigue*. 2022;164:107147.

    doi: 10.1016/j.ijfatigue.2022.107147

45. Cam LML, Neyman J. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. United States: University of California Press; 1967.

46. Kingma DP, Welling M. Auto-Encoding Variational Bayes. *In Proceedings of the 2nd International Conference on Learning Representations (ICLR2014)*; 2014.

    doi: 10.48550/arXiv.1312.6114

47. Zhang CS, Chen J, Li QL, Deng BQ, Wang J, Chen CC. Deep contrastive learning: A survey. *Acta Automatica Sin*. 2023;49(1):15-39.

    doi: 10.16383/j.aas.c220421

48. Jaiswal A, Babu AR, Zadeh MZ, Banerjee D, Makedon F. A survey on contrastive self-supervised learning. *Technologies*. 2021;9(1):2.

    doi: 10.3390/technologies9010002

49. Le-Khac PH, Healy G, Smeaton AF. Contrastive representation learning: A framework and review. *IEEE Access*. 2020;8:193907-193934.

    doi: 10.1109/ACCESS.2020.3031549

50. Liu X, Zhang F, Hou Z, *et al*. Self-supervised Learning: Generative or Contrastive. In: *IEEE Transactions on Knowledge and Data Engineering*; 2021. p. 1-1.

    doi: 10.1109/TKDE.2021.3090866

51. van den Oord A, Li Y, Vinyals O. *Representation Learning with Contrastive Predictive Coding*; 2019. *arXiv*. Available from: https://arxiv.org/abs/1807.03748 [Last accessed on 2025 Jan 08].

52. He K, Fan H, Wu Y, *et al*. Momentum Contrast for Unsupervised Visual Representation Learning. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*; 2020. p. 9726-9735. Available from: https://ieeexplore.ieee.org/document/9157636 [Last accessed on 2025 Jan 08].

53. Grill JB, Strub F, Altché F, *et al*. Bootstrap your own latent a new approach to self-supervised learning. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc.; 2020. p. 21271-21284.

54. Radford A, Kim JW, Hallacy C, *et al*. Learning Transferable Visual Models from Natural Language Supervision. In: *International Conference on Machine Learning*. PMLR; 2021. p. 8748-8763.

55. Yang C, An Z, Zhou H, *et al*. Online knowledge distillation via mutual contrastive learning for visual recognition. *IEEE Trans Pattern Anal Mach Intell*. 2023;45(8):10212-10227.

    doi: 10.1109/TPAMI.2023.3257878

56. Baevski A, Zhou H, Mohamed A, Auli M. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS 2020)*; 2020. p. 12449-12460.

57. Hershey JR, Chen Z, Le Roux J, Watanabe S. Deep clustering: Discriminative embeddings for segmentation and separation. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; 2016. p. 31-35. Available from: https://ieeexplore.ieee.org/abstract/document/7471631 [Last accessed on 2025 Jan 13].

58. Hu H, Wang X, Zhang Y, Chen Q, Guan Q. A comprehensive survey on contrastive learning. *Neurocomputing*. 2024;610:128645.

doi: 10.1016/j.neucom.2024.128645

59. Gutmann M, Hyvärinen A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. JMLR Workshop and Conference Proceedings*; 2010. p. 297-304.

60. Caron M, Misra I, Mairal J, Goyal P, Bojanowski P, Joulin A. Unsupervised learning of visual features by contrasting cluster assignments. In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2020. p. 9912-9924.

61. Reimers N, Gurevych I. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics; 2019. p. 3982-3992.

62. Rusak E, Reizinger P, Juhos A, Bringmann O, Zimmermann RS, Brendel W. *InfoNCE: Identifying the Gap between Theory and Practice. arXiv*; 2024. Available from: https://arxiv.org/abs/2407.00143 [Last accessed on 2025 Jan 08].

63. Luo Y, Chen Z, Hershey JR, Le Roux J, Mesgarani N. Deep clustering and conventional networks for music separation: Stronger together. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*; 2017. p. 61-65. Available from: https://ieeexplore.ieee.org/abstract/document/7952118 [Last accessed on 2025 Jan 08].

64. Wu S, Wang Y, Jiang Y, Zhang Q, Liu J. CRATI: Contrastive representation-based multimodal sound event localization and detection. *Knowl Based Syst*. 2024;305:112692.

    doi: 10.1016/j.knosys.2024.112692

65. Chen T, Kornblith S, Norouzi M, Hinton G. A simple framework for contrastive learning of visual representations. In: *Proceedings of the 37th International Conference on Machine Learning*. Vol. 119. JMLR; 2020.

66. Guo Q, Wang C, Xiao D, Huang Q. A lightweight open-world pest image classifier using ResNet8-based matching network and NT-Xent loss function. *Expert Syst Appl*. 2024;237:121395.

    doi: 10.1016/j.eswa.2023.121395

67. Kim T, Yoo KM, Lee SG. Self-Guided Contrastive Learning for BERT Sentence Representations. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*; 2021. p. 2528-2540.

    doi: 10.18653/v1/2021.acl-long.197

68. van der Maaten L, Hinton G. Visualizing data using t-SNE. *J Mach Learn Res*. 2008;9(86):2579-2605.

ACCSCIENCE
PUBLISHING

OUR JOURNALS

Tumor Discovery is a peer-reviewed and open-access journal that aims to present new cancer research with strong emphasis on fundamental and translational studies. *Tumor Discovery* covers topics, including but not limited to the following:

- Etiology and pathogenesis of cancer
- Mechanisms and molecular pathways underlying cancer initiation and progression
- Tumor metastasis
- Tumor evolution and heterogeneity
- Tumor microenvironment and tumor-host interactions
- Cancer genetics and genomics
- Cancer characterization using omics approaches
- Discovery and validation of cancer biomarker
- Discovery of new therapeutic targets
- New approaches of diagnostic and treatment modalities
- Statistical methods in cancer research

Global Translational Medicine is a quarterly journal that focuses on medicine, biological sciences, and biomaterials engineering. The goal of *Global Translational Medicine* is to provide a platform to researchers for showcasing their latest research works in translational medicine so as to advance the field towards the betterment of human health. Despite the advancement of omics and new technologies, the process of transforming these technologies and scientific research results into effective therapies and putting them into clinical use still has a long way to go. *Global Translational Medicine* provides a platform to fill the gaps in preclinical and inter-disciplinary research, to promote clinical translation of scientific research results, and to contribute to the conception of new and improved preventive measures as well as diagnostic and therapeutic techniques of diseases.

Global Translational Medicine covers the following themes: cardiovascular disease, metabolism/diabetes/obesity, neuroscience/neurology, cancer, biomaterials and their applications in medicine, proteomics/metabolomics, pharmacogenomics, biomarkers, bioinformatics and data mining, animal and clinical research, and medical methods arising from interdisciplinary crossover.

Start a new journal

Write to us via email if you are interested to start a new journal with AccScience Publishing. Please attach your CV, professional profile page and a brief pitch proposal in your email. We shall inform you of our decision whether we are interested to collaborate in starting a new journal.

Contact: info@accscience.com

https://accscience.com/journal/IJAMD

ACCSCIENCE
PUBLISHING

Contact
www.accscience.com
8 Burn Road, #15-03 Trivex, Singapore 369977
E-mail: editorial@accscience.com
Phone: +65 8182 1586