Journal of MATERIALS RESEARCH

Article
DOI:10.1557/s43578-025-01571-1

EARLY CAREER SCHOLARS IN MATERIALS SCIENCE, 2026

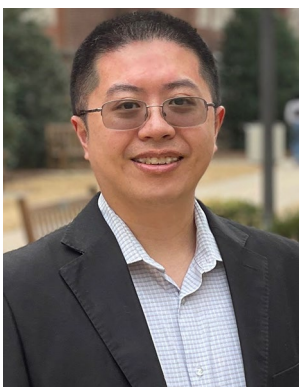
Constructing surrogates for atomistic simulations via deep learning and generative large language models

Mahshad Fani[1], William Chadwell[2], Nishad Tasnim[3], Xin Wang[4], Mohammad Younes Araghi[1], Kun Lu[4], Zejian Zhou[3], Tang Gu[5], Shuozhi Xu[1,a)]

[1] School of Aerospace and Mechanical Engineering, University of Oklahoma, Norman, OK 73019, USA
[2] Engineering Physics Program, University of Oklahoma, Norman, OK 73019, USA
[3] Electrical Engineering and Computer Science Department, University of Wyoming, Laramie, WY 82071, USA
[4] School of Library and Information Studies, University of Oklahoma, Norman, OK 73019, USA
[5] Institute of Polytechnic Science and Aeronautics (IPSA), 94200 Ivry-Sur-Seine, France
[a)] Address all correspondence to this author. e-mail: shuozhixu@ou.edu

Received: 4 July 2024; accepted: 24 March 2025; published online: 7 April 2025

Atomistic simulations offer insights into material behavior at the atomic level. However, they can be computationally intensive. In this paper, two deep learning models, deep symbolic optimization (DSO) and deep neural networks (DNN), and one generative large language model, GPT-4o, are employed to construct surrogates for atomistic simulations. Specifically, atomistic simulations are first performed to investigate the collapse of a nanovoid under hydrostatic pressure. We focus on the role of initial void radius and material characteristics, such as intrinsic and unstable stacking fault energies and surface energy (SE). We find that the critical pressure required for void collapse spans from 17.05 to 19.62 GPa, with the highest values corresponding to the maximum USFE. Additionally, an intermediate SE value (1068.13 mJ/m^2) minimizes the critical pressure. Based on the simulation results, surrogate models based on DSO, DNN, and GPT-4o are constructed, concluding that the SE affects the critical pressure the most.



Shuozhi Xu

Shuozhi Xu received his Ph.D. degree in Mechanical Engineering from the Georgia Institute of Technology in 2016. After spending several years at the University of California, Santa Barbara as a Postdoctoral Scholar, he joined the University of Oklahoma (OU) in 2023 as a tenure-track Assistant Professor in the School of Aerospace and Mechanical Engineering. He leads the OU Computational Materials, Mechanics, and Manufacturing Lab, where he advances multi-physics and data-driven materials modeling methods from subatomic to macro scale and applies them to understanding processing-structure-property relations in chemically and structurally complex materials. For the processing-structure relation, he focuses on melt-based additive manufacturing of metals and ceramics. For the structure-property relation, he mainly works with two multiscale modeling frameworks. The first framework involves concurrent multiscale modeling based on the concurrent atomistic-continuum (CAC) method. He has advanced CAC for robust modeling of defect dynamics and used CAC to address micro-scale mechanical and thermal problems in materials. To help the scientific community run CAC simulations, he developed a software package named PyCAC and built a website (www.pycac.org) to host its user's manual. The second framework concerns sequential multiscale modeling, which links density functional theory, atomistic simulation methods, phase-field models, and crystal plasticity models. He has used it to investigate submicron-scale deformation of structural materials such as multi-principal element alloys, nano-laminated materials, and additively manufactured alloys. He has won prestigious awards such as the Materials Research Society Graduate Student Award, Georgia Tech Sigma Xi Best Ph.D. Thesis Award, UC Santa Barbara Elings Prize Fellowship in Science, and Finalist for the Rising Stars in Computational Materials Science.

Introduction

Atomistic simulations model materials at the atomic level, capturing detailed interactions between atoms, making them powerful tools for predicting material properties and mechanisms that are challenging to observe experimentally [1]. However, they are often slow and computationally demanding, especially for large systems or long timescales [2]. Surrogate models help address this limitation by approximating simulation outcomes with significantly reduced computational costs [3]. Machine learning (ML) approaches, such as neural networks (NNs) and Gaussian processes, are commonly used to build these surrogates [4]. For example, Kadupitiya et al. [5] developed NN models as surrogates for atomistic simulations of soft materials, while Ruiz et al. [6] employed multivariate Gaussian process regression as surrogates to predict basic structural parameters in non-dilute random alloys. Despite the growing number of surrogate models for atomistic simulations, new research remains valuable. For instance, while highly effective and easy to implement, shallow learning models were recently found to underperform in extrapolation on materials datasets compared to deep learning models [7].

In this paper, two deep learning techniques are utilized: deep symbolic optimization (DSO) and deep neural networks (DNN). DSO is an advanced technique that uncovers mathematical expressions from data using deep learning methods [8]. It generates potential mathematical expressions, which are then evaluated based on their fit to the given dataset. The DSO method optimizes these expressions through a novel risk-seeking policy gradient algorithm, enhancing the best-performing expressions [9]. This method surpasses traditional symbolic regression tools, like the Eureqa algorithm, by efficiently navigating the search space of potential expressions [10]. The flexibility and accuracy of DSO make it a powerful tool for identifying interpretable and precise mathematical models from complex datasets [11]. On the other hand, an NN is an artificial intelligence (AI) method that processes data inspired by the human brain, using artificial neurons that work together [12]. NNs learn by adjusting connection weights through backpropagation, minimizing the loss function and improving performance. Equipped with multiple layers of neurons, DNNs outperform traditional ML methods in terms of generalization and accuracy [13]. The most important difference between DSO and DNN is output representation in that the former generates interpretable mathematical expressions or symbolic formulas that explicitly describe the relationships within the data while the latter produces outputs based on learned patterns within the network, typically resulting in a "black box" model that lacks explicit interpretability [14].

Another emerging AI tool is the generative large language models (LLMs), which originate in the subfield of natural language process (NLP). They are pre-trained on massive amount of text data using the transformer architecture [15]. With additional supervised fine-tuning and reinforcement learning from human feedback, the resulting LLMs have shown impressive capabilities in language understanding and generation. Since the advent of ChatGPT in November 2022 [16], generative LLMs have attracted wide attention from many fields due to their user-friendly interfaces and ability to respond reasonably to a variety of questions. Materials scientists have applied generative LLMs to tasks such as extracting data from unstructured text in scientific literature [17, 18]. In addition to NLP tasks, generative LLMs have also been used to answer materials questions to assist scientific discoveries in materials science owing to their ability to understand human language and generate relevant responses [19, 20]. More recently, Hao et al. [21] used LLMs as surrogate models in evolutionary algorithms. To our best knowledge, however, there hasn't been any work where LLMs were used as surrogate models for atomistic simulations. Here, a popular LLM, GPT-4o, will be applied to atomistic datasets to assess its performance.

The specific materials problem to be studied in this work concerns the collapse of a void subject to hydrostatic compression of a Cu single crystal. In metals, voids are ubiquitous. They can be formed during melting and solidification processes due to shrinkage and gas entrapment [22, 23] or as a result of stress concentration [24] at heterogeneities such as inclusions, precipitates, and grain boundaries [25]. When the metal undergoes plastic deformation, voids can grow and coalesce, ultimately resulting in crack propagation and failure of the material [26]. The voids may also collapse, leading to local densification and altering the mechanical properties of the metal [27]. Therefore, it is crucial to understand the deformation of void-containing metals to predict their behavior under stress and to develop alloys that are more resilient to deformation [28–33]. The mechanical response of metals is influenced by the change in void geometry. A smaller void inhibits dislocation motion more effectively, requiring a higher resolved shear stress for dislocations to bypass it compared to a larger void [33, 34]. Factors such as void ellipticity and orientation can also influence stress distribution and defect nucleation patterns [35]. Recently, Chen et al. [36] developed a convolutional NN model that automatically detects voids in Cu-Sn solder joints. Combining this model with finite element analyses helped identify stress concentration zones in solder joints. Similarly, Kong et al. [37] used ML algorithms along with hybrid metrology techniques to identify voids in copper lines, while Saleh et al. [38] used a combination of ML and computational simulations to improve traditional models for predicting void nucleation and growth.

Atomistic simulations have been applied to nanovoids in single crystals to analyze ductile fracture [39–43]. A single crystalline face-centered cubic (FCC) metallic material containing

spherical voids exhibits dislocation emission during yield, which is caused by the absence of alternative nucleation sites [39, 44]. It was revealed that the loading mode influences void growth and coalescence [45, 46] and that variations in specimen size altered dislocation patterns, void aspect ratios, and stress–strain responses [41, 47]. The interactions between nanovoids and neighboring voids frequently lead to coalescence or collapse of the nanovoids, which can eventually lead to the formation of cracks or shear bands [48]. The emission of dislocation is the primary mechanism driving the growth of voids in single crystals using discrete dislocation dynamics modeling technique [47]. It was determined that the elastic moduli of materials with spherical voids are linearly related to the void volume fraction, while the atomic stress concentration factor is affected by the void geometry [49].

Much work has been devoted to gaining valuable insights into the physics of void growth and collapse influenced by porous structures (e.g., void shape/size/porosity) [50–53], properties of matrix materials (e.g., Young's modulus) [54], plastic anisotropy [55], non-local effects that incorporate material size scales [56], strain rate [57], and crystallographic orientations [58–61]. Among material properties are intrinsic stacking fault energy (ISFE) and unstable stacking fault energy (USFE), which are essential in defining the mechanisms of plastic deformation in metals, particularly those of FCC materials. USFE is the highest energy obstacle that must be overcome for the leading partial dislocation to form a stacking fault [62]. ISFE affects dislocation nucleation, which is essential for understanding the mechanical properties of these materials [63]. In addition, the surface energy (SE) of a metal can influence the stability and evolution of voids, thereby affecting the yield strength. It has been demonstrated that variations in SE resulting from lattice orientation can result in voids exhibiting faceted morphologies with rounded corners in both single crystals and nanocrystalline structures, demonstrating the critical role that SE plays in void evolution across grain orientations [64].

Most previous work on void-containing metals and alloys focused on either varying porous structures (e.g., initial void size and shape) within the same material or on different material properties (e.g., ISFE and SE) across different metals. However, there is a lack of systematic studies of both sets of factors in a unified manner. This task is difficult because different materials usually differ in dozens of properties. In this work, we utilize a set of eleven interatomic potentials that differ only in ISFE, USFE, and SE while all other properties are those of Cu. This way, our work analyzes different factors within a single framework. Atomistic simulations using these potentials are carried out to investigate the plastic deformation process of a void-containing Cu single crystal under hydrostatic compression. Despite that real materials frequently contain multiple voids and other defects, we focus on the case of a single void to isolate and understand the fundamental mechanisms that govern the void collapse process. Another reason to choose a relatively simple atomistic system is that at least half of this paper is focused on constructing surrogate models. Based on the simulation data, we build surrogate models to establish the linkage among initial void radius, ISFE, USFE, SE, and critical pressure. All surrogate models conclude that SE is the most important factor affecting the critical pressure. This piece of information would allow us to design stronger, more reliable materials that are less likely to fail due to voids.

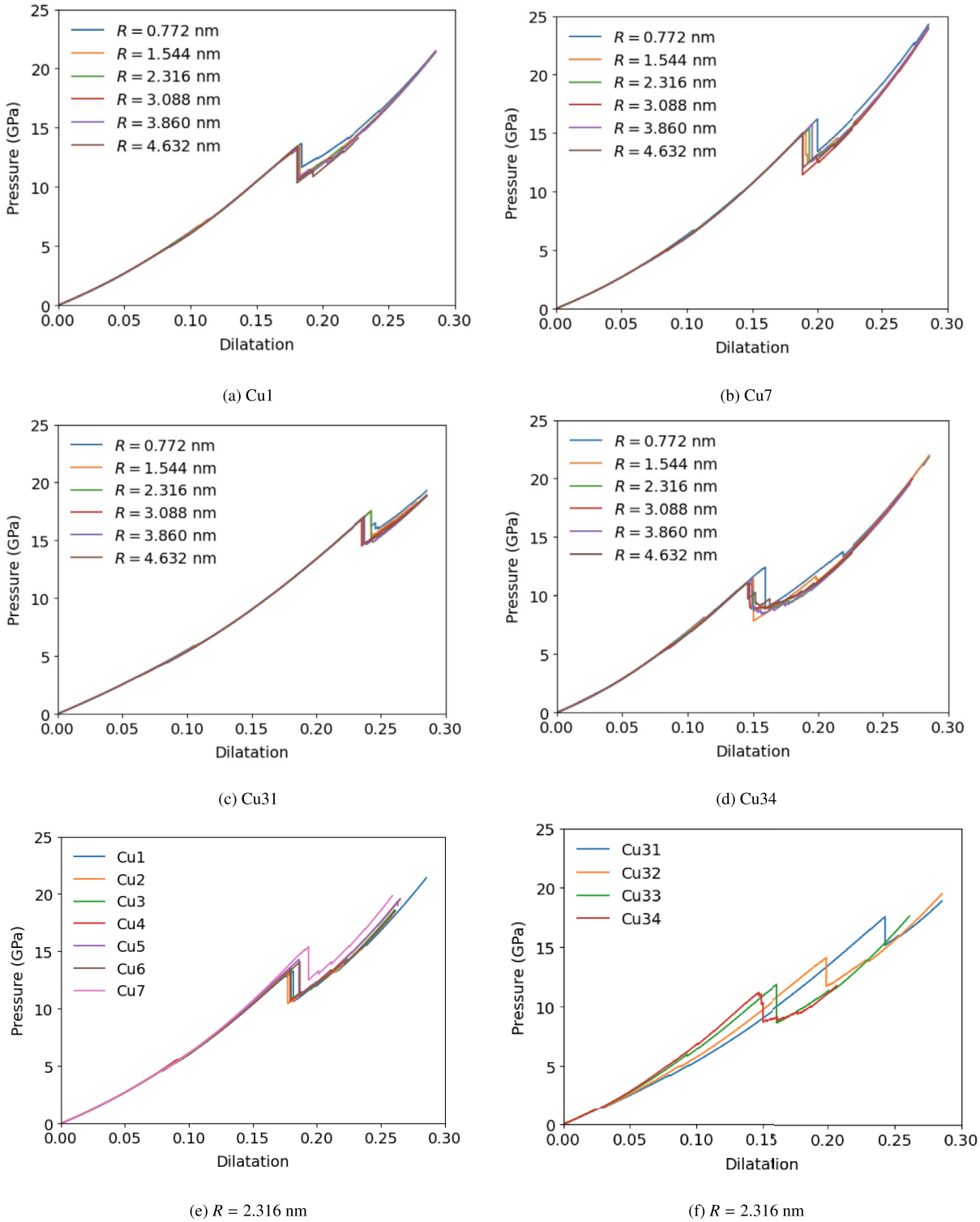
Results and discussion

Pressure-dilatation response

We first investigate the dilatation-pressure responses. Figure 1 shows some representative cases, including the minimum ISFE (i.e., the Cu1 potential), maximum ISFE (i.e., Cu7), minimum USFE (i.e., Cu31), and maximum USFE (i.e., Cu34). Note that Cu7 and Cu31, respectively, also lead to the minimum and maximum SEs. In all cases, when the dilatation is small, the pressure increases smoothly (but not linearly). However, when the dilatation is sufficiently large, the pressure experiences a significant drop. At that point, the void collapses completely and the corresponding pressure is termed the critical pressure. It is consistent with the findings in Ref. [46], where similar trends in void collapse were identified under different loading conditions. The portion of the dilatation-pressure curve prior to the yield point is approximately the same for different void radii for the same potential. Across potentials, Fig. 1(e) and (f) demonstrate that the initial portion of the dilatation-pressure curve does not change much as the ISFE varies (i.e., from Cu1 to Cu7) while its slope increases with the USFE (i.e., from Cu31 to Cu34). There is evidence to support this behavior in Ref. [65], which showed that an increase in USFE can affect dislocation motion. We also observe that, the critical pressure occurs at approximately the same dilatation in Cu1 to Cu7 which have differing ISFE. For example, when $R = 2.316$ nm, the critical dilatation is 0.1815 for Cu1, while it is at 0.1932 for Cu7. Our finding that the ISFE variations did not significantly affect the collapse thresholds is in line with Ref. [39]. In contrast, a significant difference in the critical dilatation is observed between Cu31 and Cu34. For example, when $R = 0.772$ nm, the critical dilatation is 0.2427 for Cu31, whereas it is at 0.1628 for Cu34. Our findings are in accordance with Ref. [64] which indicated that the SE is related to the stabilization of the void structure, thereby affecting the critical pressures.

Porosity-dilatation response

We then analyze the porosity-dilatation responses. Figure 2 presents results for two different initial void sizes with all eleven interatomic potentials. It is observed that the porosity



(a) Cu1

(b) Cu7

(c) Cu31

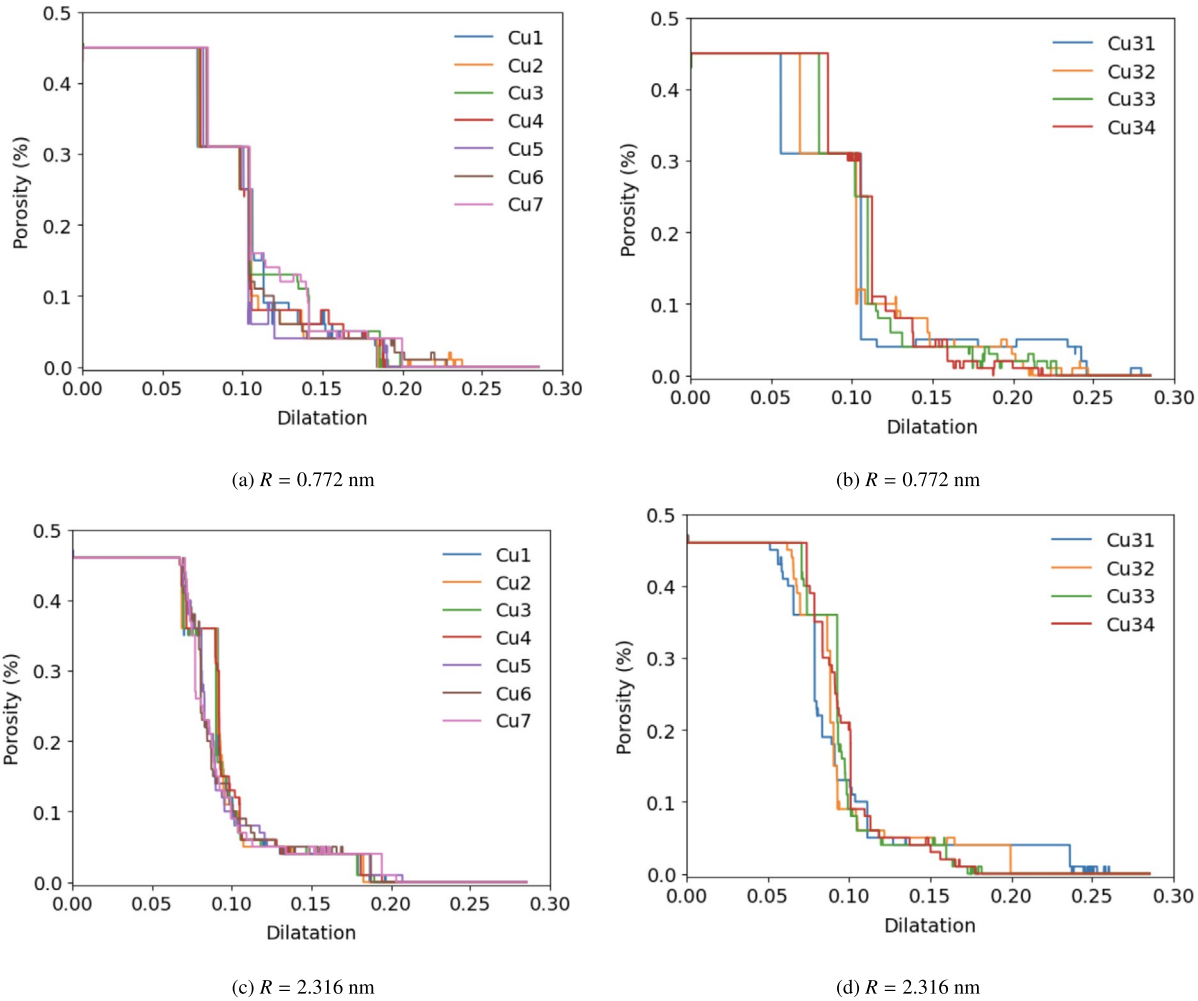
(d) Cu34

(e) R = 2.316 nm

(f) R = 2.316 nm

Figure 1: Pressure-dilatation curves based on different interatomic potentials and/or different initial void radius R.

experiences two sharp drops. The first drop occurs at $e \approx 0.08$, corresponding to the yield point where dislocations are nucleated from the void surface. The second drop occurs at $e \approx 0.18$, corresponding to the complete closure of the void, which is also where the critical pressure occurs. From Cu1 to Cu7, a larger ISFE is found to delay the void closure, in line with a higher critical pressure [Fig. 1(e)]. From Cu31 to Cu34, dislocations start to nucleate at an increasingly larger dilation while the void



(a) $R = 0.772$ nm

(b) $R = 0.772$ nm

(c) $R = 2.316$ nm

(d) $R = 2.316$ nm

Figure 2: Porosity-dilatation curves for (a, b) $R = 0.772$ nm and (c, d) $R = 2.316$ nm based on eleven interatomic potentials.

collapses at an increasingly smaller dilatation, the latter of which is aligned with smaller critical dilatation in Fig. 1(f). The same trends are found for different initial void sizes.

Selected atomistic structures for the evolution of void with $R = 2.316$ nm are presented in Fig. 3. It is shown that the void is first transformed into a stacking fault tetrahedra (SFT), which is in agreement with a previous finding in Cu [66], followed by that the SFT is completely closed while more dislocations are emitted into the system.

Critical pressure

Figure 4 visualizes the relationship among void size, ISFE, USFE, SE, and critical pressure. Qualitatively, it is shown that the critical pressure increases as the ISFE increases or the USFE decreases which aligns with observations in Ref. [67]. Additionally, there exists an intermediate SE that corresponds to the minimum critical pressure. To provide a more quantitative understanding, we employ several surrogate models in the remainder of this section.

DSO

In what follows, we define the normalized initial void radius as the $\tilde{x}_1$, the three normalized energies as $\tilde{x}_2$, $\tilde{x}_3$, and $\tilde{x}_4$, while the normalized critical pressure as $\tilde{y}$. The normalization process is described in Sect. "DSO".

Our DSO regression finds that the following equation establishes the relationship between $\tilde{x}_1$ and $\tilde{y}$, with the help of five constants (C_1 to C_5), i.e.,

$$\begin{aligned}\tilde{y}_{\mathrm{DSO}} =& C_1 \log(\tilde{x}_1) - 0.08\tilde{x}_1^{2.75} + C_2\tilde{x}_1^{1.75} + C_3\tilde{x}_1^{0.75} \\ & - \tilde{x}_1^{0.5} - \tilde{x}_1^{0.25} + C_4\tilde{x}_1^{-0.25} + C_5\end{aligned} \tag{1}$$

These five constants are related to the three energy terms, i.e., normalized ISFE, normalized USFE, and normalized SE, as follows,

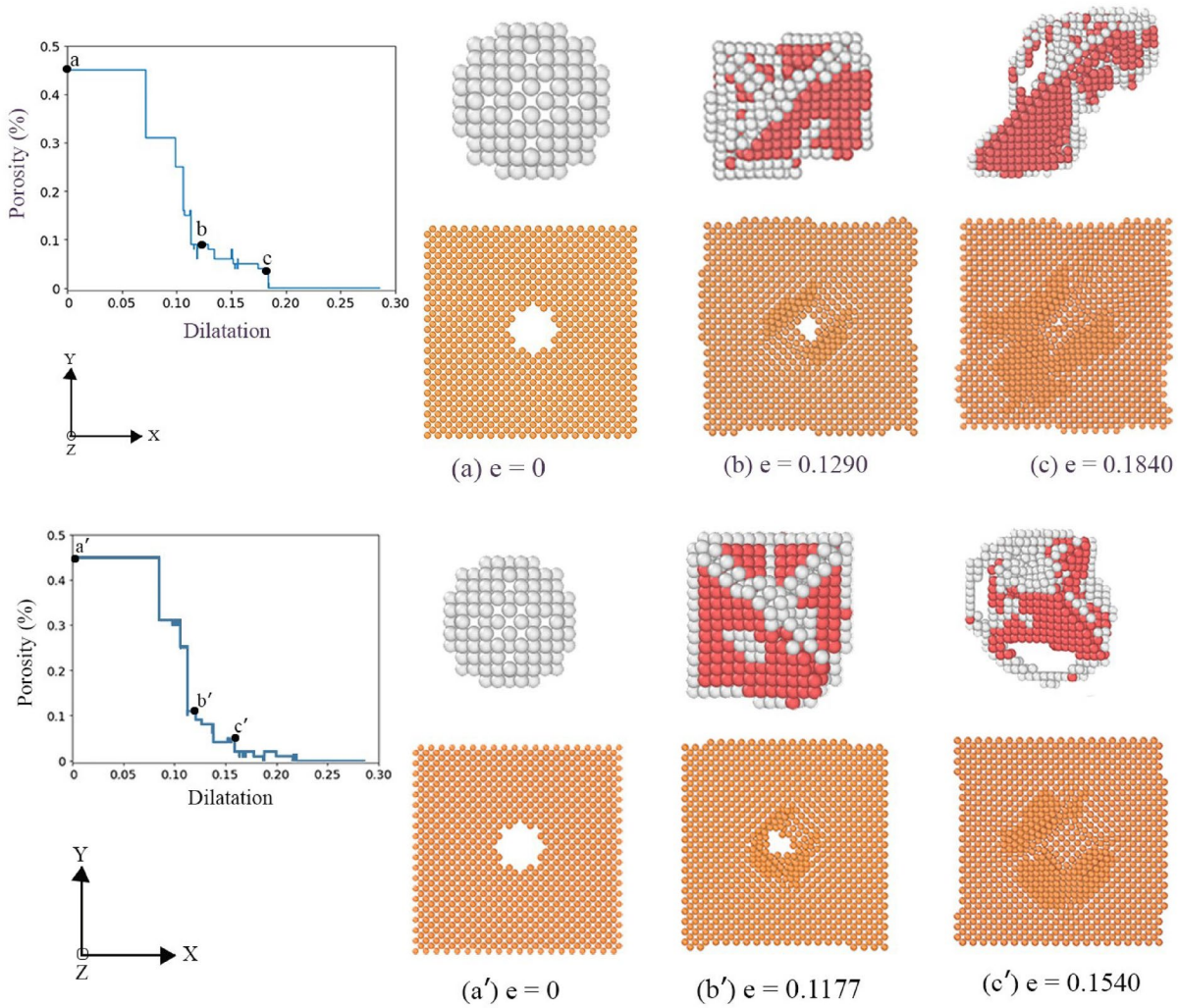


Figure 3: Evolution of the void ($R = 0.772$ nm) as a function of the dilatation e based on the (a–c) Cu1 and (a′–c′) Cu34 potentials, respectively. Top row: FCC and body-centered cubic atoms are deleted, while hexagonal close-packed atoms indicate dislocations. Bottom row shows the change in void shape and size.

$$C_1 = \frac{1}{\sqrt{\tilde{x}_3}} \tag{2}$$

$$C_2 = 0.23\tilde{x}_2 + 0.29\tilde{x}_3 + 1.17\tilde{x}_4 - 0.91 \tag{3}$$

$$\begin{aligned} C_3 = {} & 8.68\tilde{x}_2^2 - 5.64\tilde{x}_2 x_3 + 44.81\tilde{x}_2\tilde{x}_4 - 40.25\tilde{x}_2 - 14.42\tilde{x}_3^2 \\ & - 14.05\tilde{x}_3\tilde{x}_4 + 37.18\tilde{x}_3 + 56.35\tilde{x}_4^2 - 102.83\tilde{x}_4 + 36.62 \end{aligned} \tag{4}$$

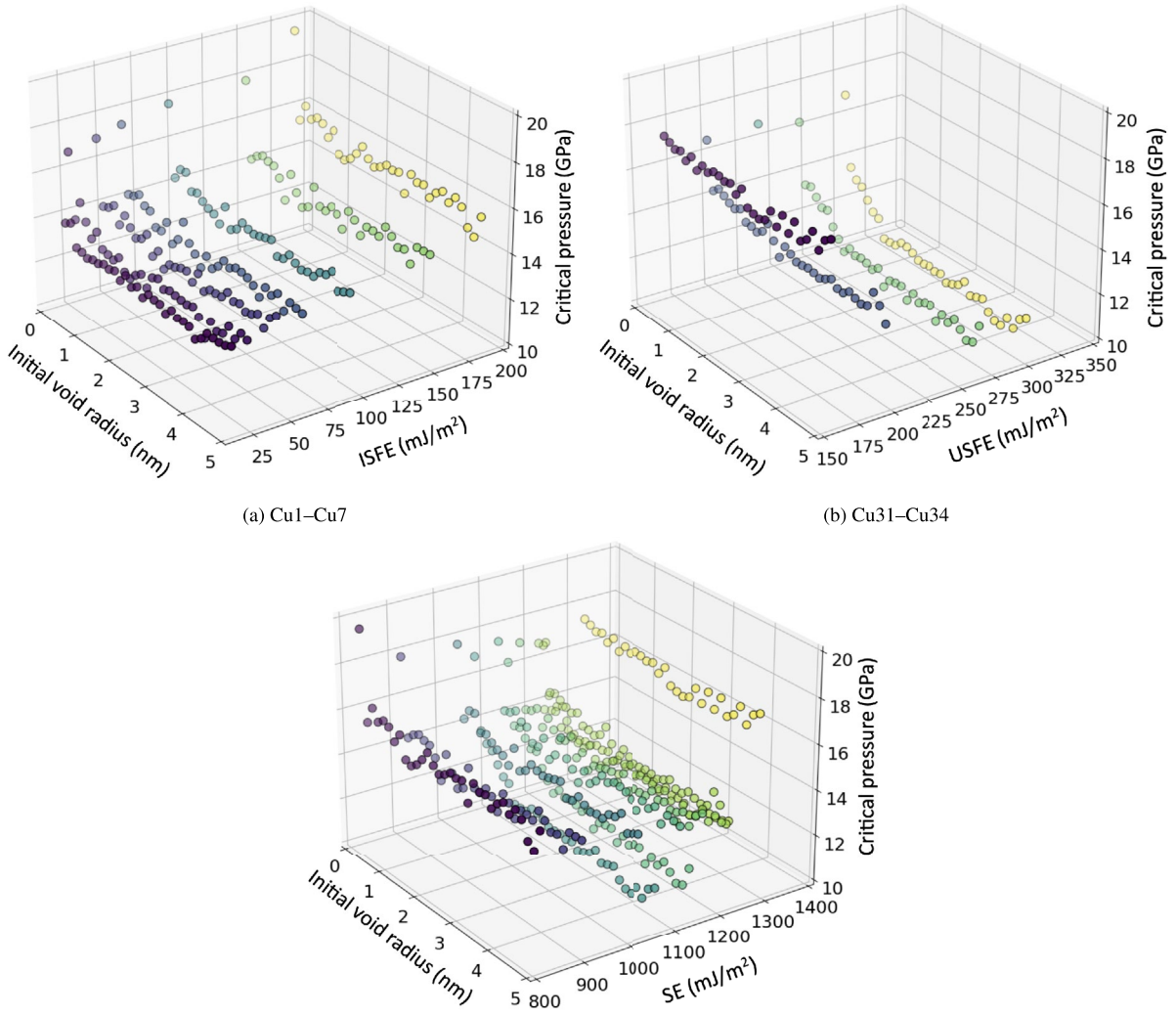
$$\begin{aligned} C_4 = {} & 0.28\tilde{x}_2^3 + 0.58\tilde{x}_2^2\tilde{x}_3 - 0.36\tilde{x}_2^2\tilde{x}_4 - 1.25\tilde{x}_2^2 \\ & - 0.10\tilde{x}_2\tilde{x}_3^2 - 1.61\tilde{x}_2\tilde{x}_3\tilde{x}_4 \\ & - 0.02\tilde{x}_2\tilde{x}_3 - 0.31\tilde{x}_2\tilde{x}_4^2 + 0.63\tilde{x}_2\tilde{x}_4 + 1.21\tilde{x}_2 \\ & - 3.23\tilde{x}_3^3 + 3.43\tilde{x}_3^2\tilde{x}_4 + 3.38\tilde{x}_3^2 \\ & - 4.83\tilde{x}_3\tilde{x}_4^2 - 2.18\tilde{x}_3\tilde{x}_4 + 0.86\tilde{x}_3 \\ & + 2.42\tilde{x}_4^3 + 0.03\tilde{x}_4^2 - 0.03\tilde{x}_4 + 1.61 \end{aligned} \tag{5}$$

$$C_5 = \sqrt{\tilde{x}_4} \tag{6}$$

This way, the porous structure (i.e., void radius) is separated from the material properties (i.e., ISFE, USFE, and SE) in Eq. 1. In the final model, the mean squared error (MSE) and R-squared score are calculated as 0.17 and 0.85, respectively, as shown in Table 1.

DNN

Each performance metric is averaged over 10 folds to provide a reliable estimate of model performance. It is found that the average MSE and average R-squared score are 0.124 and 0.956, respectively, as shown in Table 1, indicating that it makes more accurate predictions than the DSO model. Specifically after 100 epochs, the R-squared scores for training data, validation data,



(a) Cu1–Cu7

(b) Cu31–Cu34

(c) All interatomic potentials

Figure 4: Critical pressure as a function of initial void radius and one material property (ISFE or USFE or SE). In each subfigure, different values of material property are colored differently.

TABLE 1: MSE and R-squared score in DSO, DNN, and three GPT-4o models. Outputs (in GPa) of the reference point $x_1 = 2.6248$ nm, $x_2 = 44.1$, $x_3 = 232.01$, $x_4 = 1169.61$ mJ/m^2 from the five models are presented in the last row. The atomistic simulation-based value for that reference point is 13.75 GPa.

	DSO	DNN	GPT-4o-original	GPT-4o-generic	GPT-4o-random
MSE	0.17	0.124	0.441	0.903	0.788
R-squared	0.85	0.956	0.872	0.738	0.771
Output	13.62	13.73	13.63	13.62	13.78

and testing data are 0.94, 0.91, and 0.8652, respectively, as shown in Fig. 5(a). The parity plots of DNN predictions are shown in Fig. 5(b).

GPT-4o

As shown in Table 1, the GPT-4o-original model demonstrates the highest performance among the three fine-tuned GPT models, achieving an MSE of 0.441 and an R-squared score of 0.872. It is also the fine-tuned GPT model that most closely matches DNN predictions. This indicates that GPT-4o-original, equipped with full contextual information, can effectively capture the relationships among variables. The high R-squared score suggests a strong explanatory power that the model is able to take advantage of the labels and units of the domain-specific variables to understand complex dependencies within the data. The relatively low MSE value also shows that GPT-4o-original is beneficial in retaining explicit contextual cues in the data.

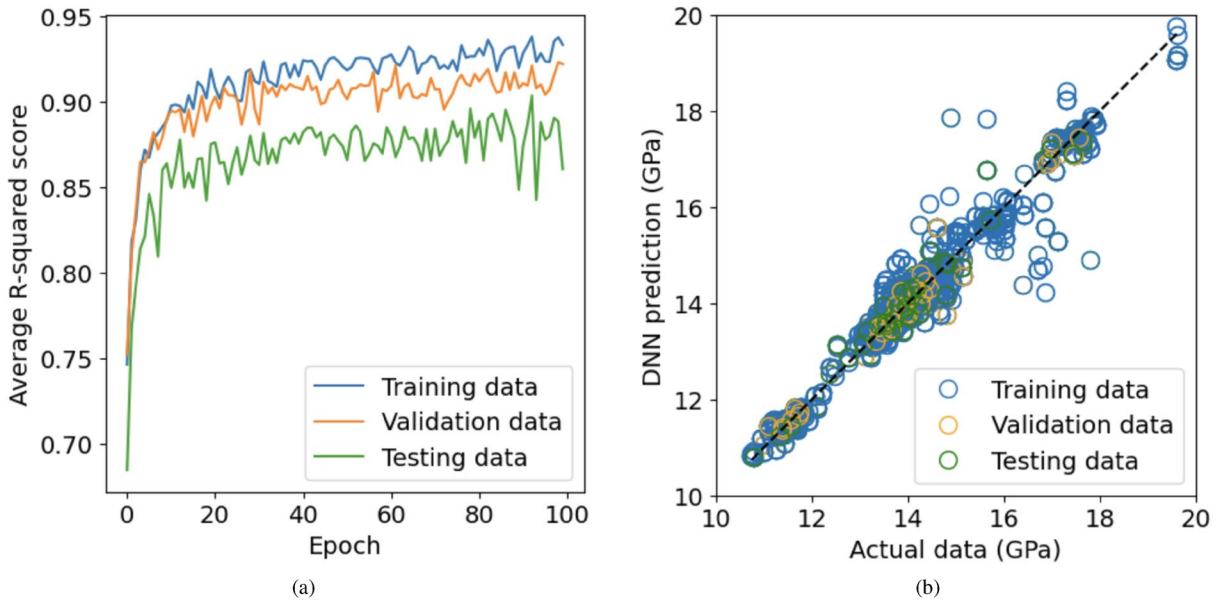


Figure 5: (a) Average R-squared scores for different datasets as a function of epoch. (b) Parity plots of the DNN predictions versus actual data for the critical pressures.

Among the three fine-tuned GPT-4o models, the GPT-4o-generic model performs the worst, with a high MSE of 0.903 and a low R-squared score of 0.738. The lack of explicit variable names and units limits the model's ability to capture the full complexity of the data. GPT-4o-generic only can rely on numerical relationships by using variables 1–4 instead of domain-specific labels. This suggests that while the model can still learn patterns in the absence of explicit context, the loss of descriptive variable labels reduces performance.

The last model, GPT-4o-random, shows an intermediate level of performance, with an MSE of 0.788 and an R-squared score of 0.771. The variable names in GPT-4o-random are completely random strings, with no inherent order or meaning. This extreme abstraction requires the model to explore relationships between numerical values, without being misled by generic labels like "variable 1." However, the lack of semantic information probably prevents it from fully optimizing its prediction.

A comparison among the five surrogate models

Based on the MSE and R-squared score, the performance among the five surrogate models is ranked as: DNN > DSO > GPT-4o-original > GPT-4o-random > GPT-4o-generic. Their relative performance can also be assessed by predicting output based on inputs at a reference point, i.e., 2.6248 nm, 44.1, 232.01, 1169.61 mJ/m^2, for which the atomistic simulation output is 13.75 GPa. Results, summarized in the last row of Table 1, confirm that the DNN model has the best performance.

Factors affecting the critical pressure

To assess how each input (initial void radius R, ISFE, USFE, and SE) affects the output (critical pressure), we apply a perturbation method, i.e., we individually change each input by ±10%. As shown in Fig. 6, the predicted critical pressure either decreases or increases due to the perturbation. The average changes in the critical pressure from the three models (whereas only the mean value of the three fine-tuned GPT-4o models are shown) are summarized in Table 2. The most accurate model, DNN, shows that SE has the greatest impact on the critical pressure, followed by USFE and ISFE, while R has the least impact on the critical pressure. The importance of the four input parameters is ranked the same according to DSO, but not GPT-4o. For example, the three fine-tuned GPT-4o models, on average, predict USFE as the least important input parameter. Nevertheless, all models highlight the primary role of SE in determining the critical pressure. The finding is physically intuitive because the critical pressure corresponds to the complete collapse of the voids, which is strongly associated with the SE. Specifically, a void with a high SE requires a large energy penalty to close, while a void with a small SE requires a higher pressure for the dislocation nucleation, postponing the yield event and delaying the void collapse.

Conclusion

In this paper, a combined atomistic simulation and surrogate model approach is employed to investigate the effects of porous structure and material properties in a void-containing

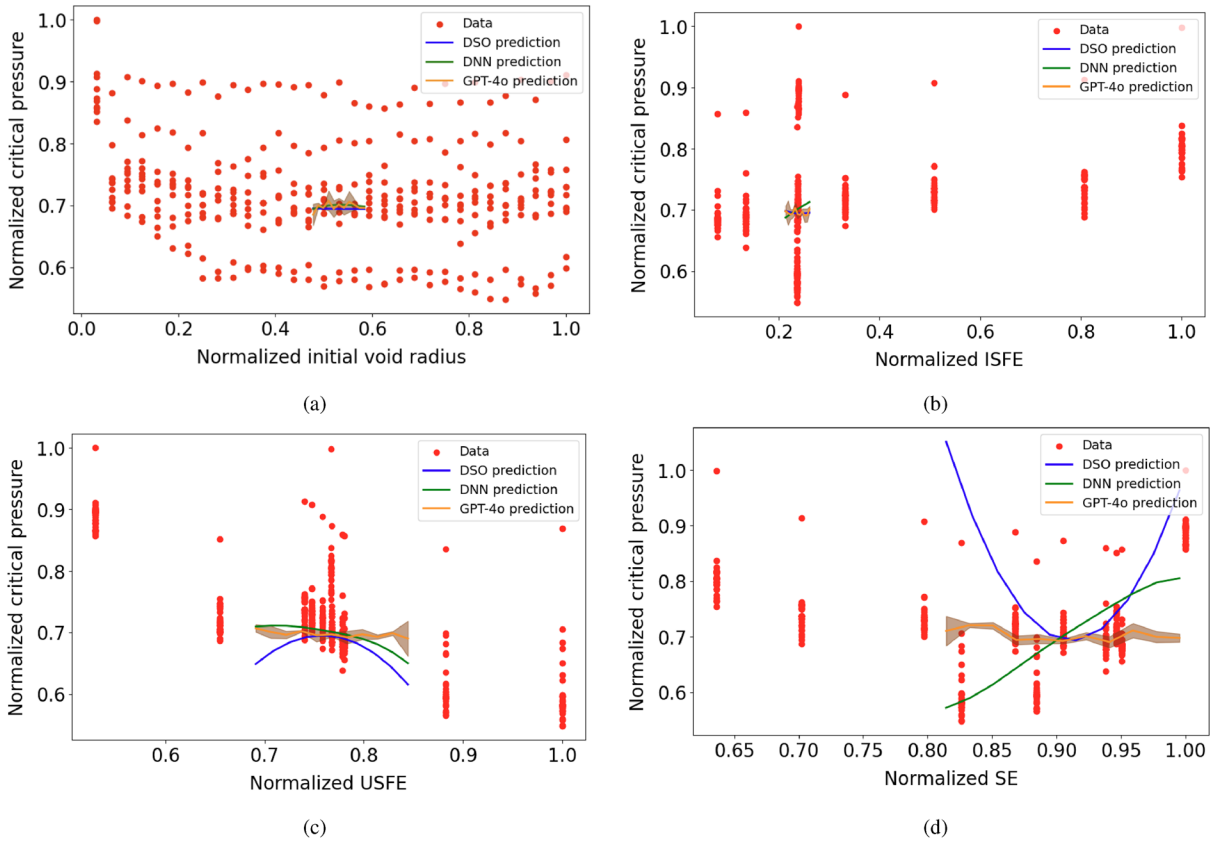


Figure 6: Red dots represent actual data points projected on 2D graphs. Blue, green, and orange lines indicate predictions by DSO, DNN, and GPT-4o models, respectively, when each input parameter (normalized initial void radius, normalized ISFE, normalized USFE, and normalized SE) is perturbed by ±10%. The shaded region indicates the standard deviation around three fine-tuned GPT-4o models.

TABLE 2: Average change in the critical pressure (in GPa) as a result of adding a ±10% perturbation to each factor — initial void radius *R*, ISFE, USFE, and SE—with respect to the reference point: 2.6248 nm, 44.1, 232.01, 1169.61 mJ/m^2. GPT-4o data are average among three fine-tuned models.

Model	*R* (±10%)	ISFE (±10%)	USFE (±10%)	SE (±10%)
DSO	0.008	0.048	1.222	6.141
DNN	0.093	0.25	0.598	2.29
GPT-4o	0.219	0.17	0.164	0.467

Cu single crystal. By systematically varying the initial void radius, ISFE, USFE, and SE by over one order of magnitude, atomistic simulations provide 352 sets of data. Subsequently, two deep learning models (DSO and DNN) and one generative LLM (GPT-4o) are applied to provide a quantitative analysis of the data. It is found that (i) a higher ISFE and/or a lower USFE lead to a higher critical pressure corresponding to complete void closure, (ii) there exists an SE (1068.13 mJ/m^2) that is associated with the minimum critical pressure (13.25 GPa), and (iii) among the four factors (i.e., initial void radius, ISFE, USFE, and SE), SE is the most important factor in determining the critical pressure. We also found that the DNN model outperforms DSO and GPT-4o, with an MSE of 0.124 and an R-squared score of 0.956, suggesting that DNN is able to capture complex, nonlinear relationships among input parameters used in our atomistic simulations. The novelty of our work lies in systematically combining atomistic simulations with surrogate models to elucidate the role of multiple factors on void behavior, demonstrating the significant impact of integrating physics-based and data-driven approaches.

The current work is not without limitations. First, it is important to note that the use of quasi-static simulations at 0 K neglects the influence of thermal vibrations and dynamic effects on the behavior of voids under real-world conditions. Second, the controlled variation of only ISFE, USFE, and SE, while maintaining other material properties constant, may not be sufficient to capture the full complexity of real materials where multiple properties vary simultaneously. Third, the semi-empirical interatomic potentials employed here, while serving the purpose of isolating certain material properties, may not be sufficiently accurate compared with ML-based interatomic potentials. These limitations may be addressed in future research by incorporating dynamic simulations to model multiple voids,

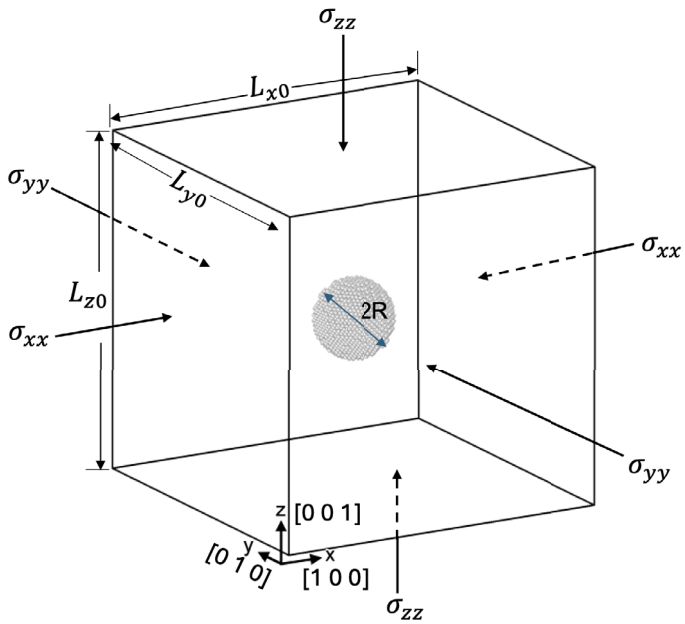
and by expanding the variation of material properties based on more accurate interatomic potentials to enhance generalizability.

Materials and methods

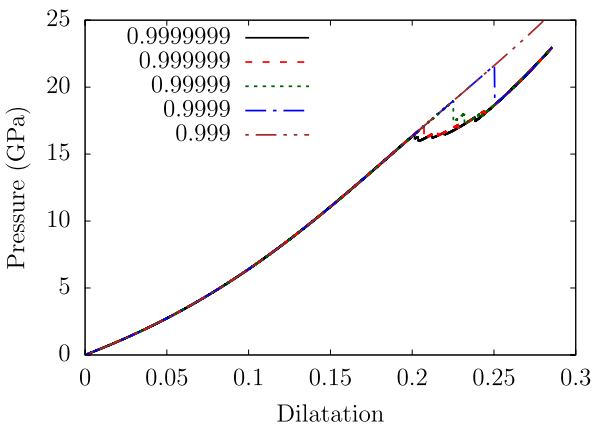
Atomistic simulations

Atomistic simulations based on the molecular statics (MS) method [68] are employed here. MS is advantageous to molecular dynamics (MD) that was used in some void-related work because it can minimize the energy of the voided structure at each strain [69]. Thus, MS results provide critical atomic-scale insights into the mechanisms driving void evolution and how it affects the critical pressure, without reaching any overdriven state which is common in MD modeling [70]. The open-source software package LAMMPS [71] is used. All atomic configurations are visualized using OVITO [72]. The adaptive common neighbor analysis method [73] is used to highlight defects in the atomistic structure.

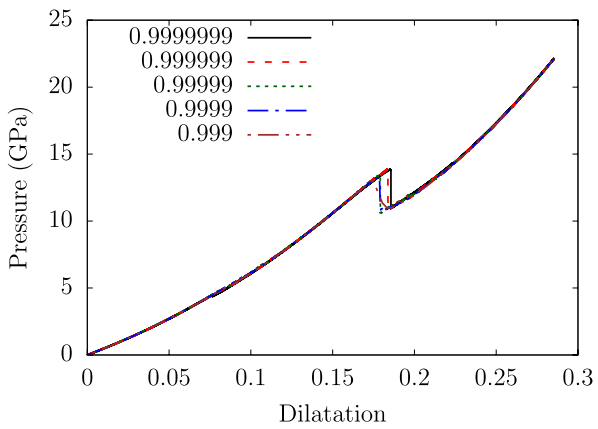
Figure 7(a) illustrates a cubic simulation cell of an FCC single crystal containing a nanovoid. The crystallographic orientations are x[100], y[010], and z[001], with periodic boundary conditions applied in all directions. L_{x0}, L_{y0}, and L_{z0}, which all



(a)



(b) R = 0.1544 nm



(c) R = 4.632 nm

Figure 7: (a) Simulation cell of a Cu single crystal containing a spherical nanovoid subject to hydrostatic compression. (b, c) Pressure-dilatation curves for various scaling factors in two simulation cells with the Cu3 potential.

equal L_0, are the initial edge lengths of the cell along the x, y, and z directions, respectively. Within the cell, a spherical void is created by removing all atoms within a specified radius (R) from the centroid. A series of 32 cells are created, with L_0 varying from L, $2L$,..., to $32L$, where $L = 1.456$ nm. In each cell, the initial void radius R is set such that the initial porosity stays the same, i.e.,

$$R = \left(\frac{L_0^3 \times 0.005}{\frac{4}{3} \times \pi} \right)^{\frac{1}{3}} \tag{7}$$

where an initial porosity of 0.5% is chosen as a typical value in Cu because Kumar et al. [74] found that as-built Cu has a porosity ranging from 0.26 to 1.29%, while heat-treated Cu possesses a reduced porosity of 0.09–0.1%. In another work, Prithivirajan et al. [75] found that a porosity of 1% would likely lead to the formation of fatigue cracks. As a result of the 0.5% initial porosity in this work, the initial void radius R ranged from 0.1544 to 4.9408 nm.

The material considered in this paper is Cu, which is modeled using eleven embedded-atom method interatomic potentials [77]. The first seven potentials, denoted as Cu1, Cu2,..., Cu7, possess differing ISFE and SE while the USFE remains approximately constant at 230 mJ/m^2 [78]. The last four potentials, denoted as Cu31, Cu32, Cu33, and Cu34, were built to predict different USFE and SE while ISFE is constant at about 44 mJ/m^2 [67]. Detailed values of the ISFEs, USFEs, and SEs for these potentials are presented in Table 3. Prior density functional theory calculations in Cu found that its ISFE, USFE [79], and mean SE [80] values are 41.83, 160.52, and 1456.67 mJ/m^2, respectively. This suggests that the Cu31 potential is the best in terms of ISFE and USFE, while no potential is associated with a good mean SE.

All other material properties, such as lattice parameters, elastic constants, and vacancy migration/formation energies, are about the same among the eleven potentials. For example, a uniform lattice parameter $a_0 = 3.639$ Å is used to build the atomistic structures. In total, 32 void/cell sizes and eleven potentials are considered, resulting in 352 simulations. In all surrogate models employed in this paper, the initial void radius R, ISFE, USFE, and SE are treated as input parameters x_1, x_2, x_3, and x_4, respectively, while the critical pressure is the only output parameter y.

After creating the void by removing atoms within a specified sphere, the remaining atomic structure may experience local stresses due to the sudden absence of neighboring atoms. To stabilize the structure, we apply the conjugate gradient algorithm to minimize the system's energy. As a result, the atoms surrounding the void can adjust their positions before any external deformation is applied. No structural collapse was observed in any case at this time. It follows that, in each simulation, a hydrostatic compressive loading is applied to the cell. A constant scaling factor δ is used such that the three edge lengths are multiplied

TABLE 3: ISFE, USFE, and SE calculated by eleven interatomic potentials. Values of ISFE and USFE were taken from Ref. [76] while those of SE were mean values among {100}, {110}, and {111} planes [67].

Potentials	ISFE (mJ/m^2)	USFE (mJ/m^2)	SE (mJ/m^2)
Cu1	14.63	235.8	1228.35
Cu2	24.89	235.14	1212.33
Cu3	44.1	232.01	1169.61
Cu4	61.54	229.17	1121.54
Cu5	94.41	225.84	1030.75
Cu6	149.61	223.66	907.91
Cu7	185.55	231.83	822.46
Cu31	44.43	159.96	1292.44
Cu32	44.27	197.87	1223.01
Cu33	44.04	266.78	1142.9
Cu34	43.96	302.18	1068.13

by δ at each MS step, followed by energy minimization using the conjugate gradient algorithm. As δ is closer to 1, the change to the simulation cell will be smaller between simulation steps, thereby minimizing the artifacts associated with the discrete nature of the simulation. However, a small δ will also result in a high computational cost. To select an appropriate δ, we applied multiple values individually, from 0.999 to 0.9999999, to two cells containing a void with $R = 0.1544$ nm and $R = 4.632$ nm, respectively, with the Cu3 potential. Results, shown in Fig. 7(b) and (c), illustrate that the dilatation-pressure curve converges as $1 > \delta \geq 0.999999$. Thus, that threshold δ value is adopted in the remainder of this paper.

At each MS step, the three normal strains (ϵ_x, ϵ_y, ϵ_z) along the three orthogonal directions are calculated by:

$$\epsilon_x = \frac{L_x - L_{x0}}{L_{x0}}, \epsilon_y = \frac{L_y - L_{y0}}{L_{y0}}, \epsilon_z = \frac{L_z - L_{z0}}{L_{z0}} \tag{8}$$

where L_x, L_y, and L_z denote the current edge length of the compressed simulation cell.

The three normal strains are related to the dilatation e, which is the change in the volume of a material, by

$$e = \epsilon_x + \epsilon_y + \epsilon_z \tag{9}$$

As the dilatation e increases, the void size (and porosity) will decrease until the void collapses completely. To measure the porosity at any given e, we need to quantify the void volume from the corresponding atomistic structure. To that end, the simulation cell is divided into many cubic voxels with an edge length of 3.4657 Å, which is between the first nearest neighbor distance (i.e., $a_0/\sqrt{2}$) and the second nearest neighbor distance (i.e., a_0). A voxel is considered part of the void if it is empty, i.e., there is no atom within it [81]. The porosity is then calculated as the fraction of empty voxels among the total number of voxels. Since some partially filled voxels will be considered as non-empty voxels, the estimated porosity will be smaller than the the

actual value. As shown in Sect. "Porosity-dilatation response", the estimated initial porosity, around 0.46%, is smaller than the actual value, 0.5%. To keep the error consistent, the voxel size will decrease as the simulation cell is compressed.

DSO

To avoid issues with units in the complex expressions, we first normalize the input and output variables by dividing each by their respective maximum values from the original data. This normalization ensures that all variables are between 0 and 1, simplifying the mathematical expressions. The maximum values for void radius, ISFE, USFE, SE, and critical pressure are 4.9408 nm, 185.55, 302.18, 1292.44 mJ/m^2, and 19.62 GPa, respectively. As a result, a data set with values 0.1544 nm, 44.43, 159.96, 1292.44 mJ/m^2, and 19.62 GPa would be normalized to 0.031, 0.239, 0.529, 1.0, and 1.0, respectively.

DSO leverages deep learning, specifically a recurrent NN (RNN), to generate and optimize mathematical expressions. The process can be broken down into the following steps:

1. *Expression representation.* Mathematical expressions are represented as trees where internal nodes are operators (e.g., $+, -, \times, \div$) and leaf nodes are either constants or variables. These trees can be linearized into sequences of tokens using pre-order traversal.
2. *RNN-based generation.* An RNN generates expressions by emitting a sequence of tokens. Each token is sampled from a categorical distribution conditioned on the previously generated tokens. This process allows the RNN to construct a diverse set of candidate expressions.
3. *Fitness evaluation.* Each generated expression is instantiated and evaluated based on its fitness, which is the MSE, which measures the average squared difference between actual and predicted values. Formally, the fitness F of an expression f is defined as

$$F(f) = -\frac{1}{N}\sum_{i=1}^{N}\left(y_i - f(X_i)\right)^2 \tag{10}$$

   where N is the number of data points, X_i is the i-th input, and y_i is the corresponding output.
4. *Policy gradient optimization.* The RNN is trained using a risk-seeking policy gradient algorithm, which optimizes for best-case performance rather than expected performance. The policy gradient update can be expressed as

$$\nabla_\theta J(\theta) = \mathbb{E}_{p_\theta(f)}\left[F(f)\nabla_\theta \log p_\theta(f)\right] \tag{11}$$

   where θ represents the parameters of the RNN, $p_\theta(f)$ is the probability of generating expression f, and $J(\theta)$ is the objective function.
5. *Sampling and training.* During training, expressions are sampled from the RNN's distribution, evaluated, and used to update the RNN's parameters. Over time, the RNN adjusts the probabilities to favor expressions with higher fitness scores, thus converging to an optimal or near-optimal solution.

The training process of DSO involves iteratively generating, evaluating, and refining mathematical expressions. Here are the detailed steps:

1. *Initialization:* Initialize the RNN with random parameters.
2. *Expression sampling:* Generate a batch of expressions by sampling from the RNN.
3. *Fitness calculation:* Evaluate the fitness of each sampled expression using the dataset.
4. *Policy update:* Update the RNN parameters using the risk-seeking policy gradient method to maximize the fitness.
5. *Iteration:* Repeat the process of sampling, evaluating, and updating until convergence or a predefined stopping criterion is met.

Regarding parameter settings, the task type was set to *regression*, aiming to derive mathematical expressions that best fit the given dataset. The *function_set* was configured to include basic arithmetic operations and other mathematical functions, specifically: `[add, sub, mul, div, poly, sqrt, log, const]`. The optimization process included a polynomial optimizer, activated because the *poly* function was part of the *function_set*. The polynomial degree was capped at 3, with a coefficient tolerance of 10^{-6}, using the `dso_least_squares` regressor for fitting the data. This selection provides a diverse set of building blocks for constructing the symbolic expressions.

The primary metric for the reward function was the inverse normalized root MSE (*inv_nrmse*), which optimizes the expressions by minimizing the error between the predicted and actual values. The metric parameters were set to [1.0]. Training hyperparameters were crucial for effective model learning. A large sample size of 2,000,000 was used, with a batch size of 1000, and the ϵ parameter was set to 0.05 to balance exploration and exploitation. The computations were performed using a single core (*n_cores_batch* set to 1), although utilizing more cores is recommended for enhanced performance, especially when the *const* token is included. The policy optimizer's learning rate was 0.0005, with an entropy weight of 0.03 and an entropy decay factor (*entropy_gamma*) of 0.7. These settings help maintain a balance between exploring new expressions and refining the best-performing ones, ensuring that the resulting symbolic expressions were both accurate and interpretable, providing meaningful insights into the relationship between void radius and critical pressure, as influenced by material parameters. Two performance metrics are used: MSE and R-squared score, the latter

of which indicates the proportion of variance in the dependent variable predictable from the independent variables.

DNN

For data preprocessing, normalization was applied, and the input data and target output are separated. To approximate the function $y = f(x_1, x_2, x_3, x_4)$, the following layers are used: an input layer with 4 neurons corresponding to the inputs from the dataset, a first hidden layer with 50 neurons, a second hidden layer with 40 neurons, and an output layer with 1 neuron corresponding to the output y. The forward pass computations for each layer are as follows: the first hidden layer activations are given by $\mathbf{a}_1 = \sigma(W_1\mathbf{x} + \mathbf{b}_1)$, the second hidden layer activations are given by $\mathbf{a}_2 = \sigma(W_2\mathbf{a}_1 + \mathbf{b}_2)$, and the output layer activations are given by $y = \phi(W_3\mathbf{a}_2 + \mathbf{b}_3)$, where $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$ is the input vector, W_1, W_2, W_3 are the weight matrices for the respective layers, $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$ are the bias vectors for the respective layers, σ is the activation function for the hidden layers, and ϕ is the linear activation function for the output layer [82].

The backpropagation algorithm is utilized to update the network's weights and biases, minimizing the loss function, which is the MSE, i.e.,

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{12}$$

where y_i is the actual output, $\hat{y}_i$ is the predicted output by the network, and n is the number of training samples. The backpropagation steps are as follows. The gradient of the loss with respect to the predicted output is computed as:

$$\delta_{\text{output}} = \frac{\partial \text{MSE}}{\partial \hat{y}} = -2(y - \hat{y}) \tag{13}$$

The error is then propagated backward through the network as:

$$\delta_2 = \delta_{\text{output}} \cdot \phi'(W_3\mathbf{a}_2 + \mathbf{b}_3) \tag{14}$$

$$\delta_1 = (\delta_2 W_2^T) \cdot \sigma'(W_1\mathbf{x} + \mathbf{b}_1) \tag{15}$$

With the gradient descent method, the MSE is reduced by updating the weights and bias in the descent direction. The weights and biases of the output and hidden layers are updated using the following equations:

$$W_3 := W_3 - \eta \cdot \delta_{\text{output}}\mathbf{a}_2^T \tag{16}$$

$$\mathbf{b}_3 := \mathbf{b}_3 - \eta \cdot \delta_{\text{output}} \tag{17}$$

$$W_2 := W_2 - \eta \cdot \delta_2\mathbf{a}_1^T \tag{18}$$

$$\mathbf{b}_2 := \mathbf{b}_2 - \eta \cdot \delta_2 \tag{19}$$

$$W_1 := W_1 - \eta \cdot \delta_1\mathbf{x}^T \tag{20}$$

$$\mathbf{b}_1 := \mathbf{b}_1 - \eta \cdot \delta_1 \tag{21}$$

where η is the learning rate which controls how much the weights of an NN change during each iteration. Here, $\eta = 0.01$ by which the model learns accurately to help the optimization process converge. σ' and ϕ' are the derivatives of the activation functions. To evaluate the model's generalization capability, 10-fold cross-validation is employed. The dataset is partitioned into 10 subsets, and the model is trained and validated 10 times, each time using a different subset for validation and the remaining subsets for training. The DNN model is trained using 80% of the data for training, 10% for validation and 10% for testing. Similar to the DSO model, the DNN model's performance is assessed on unseen dataset using the two metrics: MSE, shown in Eq. 12, and R-squared score.

GPT-4o

A popular generative LLM, GPT-4o, is employed. A recent study found that LLMs' mathematical reasoning capability may be affected by trivial changes such as those in variable names [83]. Hence, we consider three datasets, differing only in how each type of data is named. In the first dataset, each data type is associated with an index name (e.g., critical pressure) and units (e.g., GPa). This enables the model to interpret and learn relationships among variables in a contextually rich setting. In the second dataset, each type of input data is anonymized by assigning generic names, i.e., "variable 1" through "variable 4," while the output (i.e., critical pressure) is renamed as "result," with all units removed. By eliminating specific contextual clues, this setting intends to assess the model's adaptability to decontextualized inputs and to challenge it to identify relationships among variables without relying on units or domain-specific indices. In the third dataset, each type of data is represented by an arbitrary alphanumeric string with symbols, such as "#aL@5p!Qx" and "G &3*l^QjZ," with units similarly removed. This highest level of anonymization presents the model with minimal interpretative context. As such, the model needs to learn the numerical relationships between input and output. The goal is to push the model to its abstraction limits, where both naming conventions and units are stripped.

With each of these three datasets, we fine-tune a GPT-4o model, specifically `gpt-4o-2024-08-06`. As a result, three distinct models are obtained: GPT-4o-original, GPT-4o-generic, and GPT-4o-random. Similar to DSO and DNN models, the fine-tuned GPT-4o models' performances are assessed using MSE and R-squared score. OpenAI's auto-configured parameter settings are utilized throughout the fine-tuning process. Take the second model as an example, its dialogue examples are constructed as follows:

```
{
    "messages": [
        {"role": "system", "content": "You are a predictive
    machine that identifies relationships between variables and
    result."},
        {"role": "user", "content": "Given the value of
    variable 1 of 0.1544, the variable 2 of 44.43, the variable
    3 of 159.96, the variable 4 of 1292.44, what is the result
    ?"},
        {"role": "assistant", "content": "19.61759145"}
        ]
    }
```

Acknowledgments

This work used Bridges-2 at the Pittsburgh Supercomputing Center through allocation # MAT230058 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296. Some of the computing for this project was performed at the OU Supercomputing Center for Education & Research (OSCER) at the University of Oklahoma (OU).

Author contributions

S.X. designed the project. M.F. and W.C. conducted atomistic simulations. N.T. and Z.Z. developed the DNN model. M.Y.A. and T.G. developed the DSO model. X.W. and K.L. fine-tuned the GPT-4o model. M.F., T.G., Z.Z., and X.W. drafted the manuscript. S.X. and M.F. reviewed and edited the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding

We acknowledge the support of the Vice President for Research and Partnerships of the University of Oklahoma (OU) and the Data Institute for Societal Challenges. S.X. was supported by a grant from the Research Council of the OU Norman Campus. M.F., M.Y.A., and S.X. are grateful for the startup funds provided by OU.

Data availability

The data presented in this study are openly available at https://github.com/shuozhixu/FYRE2024.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

1. D. Farkas, Atomistic simulations of metallic microstructures. Curr. Opin. Solid State Mater. Sci. **17**, 284–297 (2013). https://doi.org/10.1016/j.cossms.2013.11.002
2. D. Perez, B.P. Uberuaga, Y. Shim, J.G. Amar, A.F. Voter, Chapter 4 Accelerated Molecular Dynamics Methods: Introduction and Recent Developments, in *Annual Reports in Computational Chemistry*, vol. 5, ed. by R.A. Wheeler (Elsevier, Amsterdam, 2009), pp.79–98. https://doi.org/10.1016/S1574-1400(09)00504-0
3. E.A. Barros de Moraes, J.L. Suzuki, M. Zayernouri, Atomistic-to-meso multi-scale data-driven graph surrogate modeling of dislocation glide. Comput. Mater. Sci. **197**, 110569 (2021). https://doi.org/10.1016/j.commatsci.2021.110569
4. C. Nyshadham, M. Rupp, B. Bekker, A.V. Shapeev, T. Mueller, C.W. Rosenbrock, G. Csányi, D.W. Wingate, G.L.W. Hart, Machine-learned multi-system surrogate models for materials prediction. npj Comput. Mater. **5**, 51 (2019). https://doi.org/10.1038/s41524-019-0189-9
5. J.C.S. Kadupitiya, F. Sun, G. Fox, V. Jadhao, Machine learning surrogates for molecular dynamics simulations of soft materials. J. Comput. Sci. **42**, 101107 (2020). https://doi.org/10.1016/j.jocs.2020.101107
6. C. Ruiz, A. Raj, S. Xu, Multivariate Gaussian process surrogates for predicting basic structural parameters of refractory non-dilute random alloys. APL Mach. Learn. **2**, 026107 (2024). https://doi.org/10.1063/5.0186045
7. S. Liu, B. Bocklund, J. Diffenderfer, S. Chaganti, B. Kailkhura, S.K. McCall, B. Gallagher, A. Perron, J.T. McKeown, A comparative study of predicting high entropy alloy phase fractions with traditional machine learning and deep neural networks. npj Comput. Mater. **10**, 172 (2024)
8. K.N. Petersen, M.L. Landajuela, T.N. Mundhenk, C. Pineda Santiago, S. Kim, J.J. Thiagarajan, Deep Symbolic Regression: Recovering Mathematical Expressions from Data via Risk-Seeking Policy Gradients, in *Proceedings of the 9th International Conference on Learning Representations, ICLR, Virtual Conference*. (2021), https://openreview.net/forum?id=Qlgo4g5Yh1
9. F.L. da Silva, A. Goncalves, S. Nguyen, D. Vashchenko, R. Glatt, T. Desautels, M. Landajuela, D. Faissol, B. Petersen, Language model-accelerated deep symbolic optimization. Neural Comput. Appl. (2023). https://doi.org/10.1007/s00521-023-08802-8
10. S.-M. Udrescu, M. Tegmark, A.I. Feynman, A physics-inspired method for symbolic regression. Sci. Adv. **6**, eaay2631 (2020)
11. Z. Xu, N. Yuktanan, M. Liu, T. Gu, M. Shi, Characterization of microstructures and micromechanical properties of Ti6Al4V powders. Powder Technol. **448**, 120352 (2024)
12. W. Sha, K. Edwards, The use of artificial neural networks in materials science based research. Mater. Des. **28**, 1747–1752 (2007)
13. S. Feng, H. Zhou, H. Dong, Using deep neural network with small dataset to predict material defects. Mater. Des. **162**, 300–310 (2019)
14. A. Korotcov, V. Tkachenko, D.P. Russo, S. Ekins, Comparison of deep learning with multiple machine learning methods and metrics using diverse drug discovery data sets. Mol. Pharm. **14**, 4462–4475 (2017)
15. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł Kaiser, I. Polosukhin, Attention is all you need. Adv. Neural. Inf. Process. Syst. **30**, 5998–6008 (2017)
16. A. Radford, Improving language understanding by generative pre-training, OpenAI Technical Report (2018), https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf
17. X. Wang, L. Huang, S. Xu, K. Lu, How does a generative large language model perform on domain-specific information extraction?-a comparison between gpt-4 and a rule-based method on band gap extraction. J. Chem. Inf. Model. **64**, 7895–7904 (2024). https://doi.org/10.1021/acs.jcim.4c00882
18. M.P. Polak, D. Morgan, Extracting accurate materials data from research papers with conversational language models and prompt engineering. Nat. Commun. **15**, 1569 (2024)
19. Y.J. Park, D. Kaplan, Z. Ren, C.-W. Hsu, C. Li, H. Xu, S. Li, J. Li, Can ChatGPT be used to generate scientific hypotheses? J. Materiomics **10**, 578–584 (2024)
20. K.M. Jablonka, Q. Ai, A. Al-Feghali, S. Badhwar, J.D. Bocarsly, A.M. Bran, S. Bringuier, L.C. Brinson, K. Choudhary, D. Circi et al., 14 examples of how LLMs can transform materials science and chemistry: a reflection on a large language model hackathon. Digit. Discov. **2**, 1233–1250 (2023)
21. H. Hao, X. Zhang, A. Zhou, Large language models as surrogate models in evolutionary algorithms: a preliminary study. Swarm Evol. Comput. **91**, 101741 (2024). https://doi.org/10.1016/j.swevo.2024.101741
22. C. Crussard, J. Plateau, R. Tamhankar, G. Henry, D. Lajeunesse, A Comparison of Ductile and Fatigue Fractures, in *Fracture*, vol. 1, ed. by E. Name (Wiley, New York, 1959), pp.31–37
23. H. Rogers, The tensile fracture of ductile metals. Metal. Soc. AIME **218**, 498–506 (1960)
24. M.F. Horstemeyer, A.M. Gokhale, A void-crack nucleation model for ductile metals. Int. J. Solids Struct. **36**, 5029–5055 (1999)
25. M. Ashby, C. Gandhi, D. Taplin, Overview No. 3 fracture-mechanism maps and their construction for FCC metals and alloys. Acta Metall. **27**, 699–729 (1979)
26. R.N. Gardner, T. Pollock, H. Wilsdorf, Crack initiation at dislocation cell boundaries in the ductile fracture of metals. Mater. Sci. Eng. **29**, 169–174 (1977)
27. M.R.G. Prasad, A. Neogi, N. Vajragupta, R. Janisch, A. Hartmaier, Influence of temperature on void collapse in single crystal nickel under hydrostatic compression. Materials **14**, 2369 (2021)

28. G.R. Johnson, W.H. Cook, Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures. Eng. Fract. Mech. **21**, 31–48 (1985)
29. J.R. Rice, D.M. Tracey, On the ductile enlargement of voids in triaxial stress fields. J. Mech. Phys. Solids **17**, 201–217 (1969)
30. Y. Guo, C. Paramatmuni, E. Avcu, Void nucleation and growth from heterophases and the exploitation of new toughening mechanisms in metals. Curr. Comput. Aided Drug Des. **13**, 860 (2023)
31. J. Leclerc, M. Marteleur, M.-S. Colla, T. Pardoen, L. Noels, V.-D. Nguyen, Ductile fracture of high strength steels with morphological anisotropy, Part II: Nonlocal micromechanics-based modeling. Eng. Fract. Mech. **248**, 107716 (2021)
32. N. Bonora, G. Testa, Plasticity damage self-consistent model incorporating stress triaxiality and shear controlled fracture mechanisms—model formulation. Eng. Fract. Mech. **271**, 108634 (2022)
33. S. Xu, Y. Su, S.Z. Chavoshi, Deformation of periodic nanovoid structures in Mg single crystals. Mater. Res. Express **5**, 016523 (2018)
34. L. Xiong, S. Xu, D.L. McDowell, Y. Chen, Concurrent atomistic-continuum simulations of dislocation-void interactions in FCC crystals. Int. J. Plast. **65**, 33–42 (2015)
35. S. Xu, Y. Su, Nanovoid growth in bcc α-fe: influences of initial void geometry. Model. Simul. Mater. Sci. Eng. **24**, 085015 (2016)
36. K. Chen, Y. Zhang, G. Cheng, Y. Zhang, A machine learning and finite element simulation-based void inspection for higher solder joint reliability. Microelectron. Reliab. **154**, 115323 (2024)
37. D. Kong, K. Motoyama, H. Huang, B. Mendoza, M. Breton, G.R. Muthinti, H. Shobha, L. Jiang, J. Li, J.J. Demarest et al., Machine Learning and Hybrid Metrology Using Scatterometry and LE-XRF to Detect Voids in Copper Lines, in *Metrology, Inspection, and Process Control for Microlithography XXXIII*, vol. 10959 (SPIE, 2019), pp. 37–50
38. A. Saleh, H. Zahedmanesh, H. Ceric, K. Croes, I. De Wolf, Dynamics of Electromigration Voids in Cu Interconnects: Investigation Using a Physics-Based Model Augmented by Neural Networks, in *IEEE International Interconnect Technology Conference (IITC)*. (IEEE, 2022), pp.25–27
39. G. Potirniche, M. Horstemeyer, G. Wagner, P. Gullett, A molecular dynamics study of void growth and coalescence in single crystal nickel. Int. J. Plast **22**, 257–278 (2006)
40. K. Zhao, C. Chen, Y. Shen, T. Lu, Molecular dynamics study on the nano-void growth in face-centered cubic single crystal copper. Comput. Mater. Sci. **46**, 749–754 (2009)
41. S. Traiviratana, E.M. Bringa, D.J. Benson, M.A. Meyers, Void growth in metals: atomistic calculations. Acta Mater. **56**, 3874–3886 (2008)
42. T. Tang, S. Kim, M. Horstemeyer, Fatigue crack growth in magnesium single crystals under cyclic loading: molecular dynamics simulation. Comput. Mater. Sci. **48**, 426–439 (2010)
43. T. Tang, S. Kim, M. Horstemeyer, Molecular dynamics simulations of void growth and coalescence in single crystal magnesium. Acta Mater. **58**, 4742–4759 (2010)
44. G. Potirniche, J. Hearndon, M. Horstemeyer, X. Ling, Lattice orientation effects on void growth and coalescence in FCC single crystals. Int. J. Plast **22**, 921–942 (2006)
45. R. Rudd, E. Seppälä, L. Dupuy, J. Belak, Void coalescence processes quantified through atomistic and multiscale simulation,. J. Comput. Aided Mater. Des. **14**, 425–434 (2007)
46. R.E. Rudd, Void growth in bcc metals simulated with molecular dynamics using the finnis-sinclair potential. Philos. Mag. **89**, 3133–3161 (2009)
47. J. Segurado, J. LLorca, Discrete dislocation dynamics analysis of the effect of lattice orientation on void growth in single crystals. Int. J. Plast **26**, 806–819 (2010)
48. X.-X. Zhang, Z.-S. Cui, Theoretical study of void closure in nonlinear plastic materials. Appl. Math. Mech. **30**, 631–642 (2009)
49. X. Yang, T. Zhou, C. Chen, Effective elastic modulus and atomic stress concentration of single crystal nano-plate with void. Comput. Mater. Sci. **40**, 51–56 (2007)
50. A. Needleman, Void growth in an elastic-plastic medium. J. Appl. Mech. **39**, 964–970 (1972)
51. T. Pardoen, J. Hutchinson, An extended model for void growth and coalescence. J. Mech. Phys. Solids **48**, 2467–2512 (2000)
52. M. Kuna, D. Sun, Three-dimensional cell model analyses of void growth in ductile materials. Int. J. Fract. **81**, 235–258 (1996)
53. M. Worswick, R. Pick, Void growth and constitutive softening in a periodically voided solid. J. Mech. Phys. Solids **38**, 601–625 (1990)
54. N. Hosseini, J.C. Nieto-Fuentes, M. Dakshinamurthy, J.A. Rodríguez-Martínez, G. Vadillo, The effect of material orientation on void growth. Int. J. Plast **148**, 103149 (2022)
55. V. Monchiet, O. Cazacu, E. Charkaluk, D. Kondo, Macroscopic yield criteria for plastic anisotropic materials containing spheroidal voids. Int. J. Plast **24**, 1158–1189 (2008)
56. A.A. Benzerga, J.-B. Leblond, A. Needleman, V. Tvergaard, Ductile failure modeling. Int. J. Fract. **201**, 29–80 (2016)
57. S. Xu, Z. Hao, Y. Su, Y. Yu, Q. Wan, W. Hu, An analysis on nanovoid growth in body-centered cubic single crystalline vanadium. Comput. Mater. Sci. **50**, 2411–2421 (2011)
58. M. Bhatia, K. Solanki, A. Moitra, M. Tschopp, Investigating damage evolution at the nanoscale: molecular dynamics simulations of nanovoid growth in single-crystal aluminum. Metall. Mater. Trans. A **44**, 617–626 (2013)
59. J. Wang, Z. Yue, Z. Wen, D. Zhang, C. Liu, Orientation effects on the tensile properties of single crystal nickel with nanovoid: atomistic simulation. Comput. Mater. Sci. **132**, 116–124 (2017)
60. Y. Zhang, S. Jiang, X. Zhu, D. Sun, Orientation dependence of void growth at triple junction of grain boundaries in nanoscale tricrystal nickel film subjected to uniaxial tensile loading. J. Phys. Chem. Solids **98**, 220–232 (2016)

61. W. Liu, X. Zhang, J. Tang, Y. Du, Simulation of void growth and coalescence behavior with 3D crystal plasticity theory. Comput. Mater. Sci. **40**, 130–139 (2007)
62. P.M. Anderson, J.P. Hirth, J. Lothe, *Theory of Dislocations*, 3rd edn. (Cambridge University Press, Cambridge, 2017)
63. W. Li, S. Lu, Q.-M. Hu, B. Johansson, S.K. Kwon, M. Grehk, J.Y. Johnsson, L. Vitos, Generalized stacking fault energy of γ-Fe. Philos. Mag. **96**, 524–541 (2016)
64. W. Liu, N. Wang, Y. Ji, P. Song, C. Zhang, Z. Yang, L. Chen, Effects of surface energy anisotropy on void evolution during irradiation: a phase-field model. J. Nucl. Mater. **479**, 316–322 (2016)
65. M.A. Meyers, A. Mishra, D.J. Benson, Mechanical properties of nanocrystalline materials. Prog. Mater Sci. **51**, 427–556 (2006)
66. B.P. Uberuaga, R.G. Hoagland, A.F. Voter, S.M. Valone, Direct transformation of vacancy voids to stacking fault tetrahedra. Phys. Rev. Lett. **99**, 135501 (2007)
67. V. Borovikov, M.I. Mendelev, A.H. King, Effects of stable and unstable stacking fault energy on dislocation nucleation in nano-crystalline metals. Model. Simul. Mater. Sci. Eng. **24**, 085017 (2016)
68. Y.-R. Jeng, C.-M. Tan, Computer simulation of tension experiments of a thin film using an atomic model. Phys. Rev. B **65**, 174107 (2002)
69. O. Vinogradov, A static analog of molecular dynamics method for crystals. Int. J. Comput. Methods **3**, 153–161 (2006)
70. H.-J. Lee, B.D. Wirth, Molecular dynamics simulation of dislocation-void interactions in BCC Mo. J. Nucl. Mater. **386**, 115–118 (2009)
71. A.P. Thompson, H.M. Aktulga, R. Berger, D.S. Bolintineanu, W.M. Brown, P.S. Crozier, P.J. Int Veld, A. Kohlmeyer, S.G. Moore, T.D. Nguyen et al., LAMMPS-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. Comput. Phys. Comm. **271**, 108171 (2022)
72. A. Stukowski, Visualization and analysis of atomistic simulation data with ovito-the open visualization tool. Model. Simul. Mater. Sci. Eng. **18**, 015012 (2009)
73. A. Stukowski, Structure identification methods for atomistic simulations of crystalline materials. Model. Simul. Mater. Sci. Eng. **20**, 045021 (2012)
74. A. Kumar, Y. Bai, A. Eklund, C.B. Williams, Effects of hot isostatic pressing on copper parts fabricated via binder jetting. Proc. Manuf. **10**, 935–944 (2017)
75. V. Prithivirajan, M.D. Sangid, The role of defects and critical pore size analysis in the fatigue response of additively manufactured IN718 via crystal plasticity. Mater. Des. **150**, 139–153 (2018)
76. W. Ji, W.-R. Jian, Y. Su, S. Xu, I.J. Beyerlein, Role of stacking fault energy in confined layer slip in nanolaminated Cu. J. Mater. Sci. **59**, 4775–4787 (2024)
77. M.S. Daw, M.I. Baskes, Embedded-atom method: Derivation and application to impurities, surfaces, and other defects in metals. Phys. Rev. B **29**, 6443 (1984)
78. V. Borovikov, M.I. Mendelev, A.H. King, R. LeSar, Effect of stacking fault energy on mechanism of plastic deformation in nanotwinned FCC metals. Model. Simul. Mater. Sci. Eng. **23**, 055003 (2015)
79. Y. Su, S. Xu, I.J. Beyerlein, Density functional theory calculations of generalized stacking fault energy surfaces for eight face-centered cubic transition metals. J. Appl. Phys. **126**, 10 (2019)
80. R. Tran, Z. Xu, B. Radhakrishnan, D. Winston, W. Sun, K.A. Persson, S.P. Ong, Surface energies of elemental crystals. Sci. Data **3**, 160080 (2016). https://doi.org/10.1038/sdata.2016.80
81. A.M. Dongare, A.M. Rajendran, B. LaMattina, M.A. Zikry, D.W. Brenner, Atomic scale studies of spall behavior in nanocrystalline Cu. J. Appl. Phys. **108**, 113518 (2010)
82. I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning* (MIT Press, Cambridge, Massachusetts, 2016)
83. I. Mirzadeh, K. Alizadeh, H. Shahrokhi, O. Tuzel, S. Bengio, M. Farajtabar, GSM-Symbolic: Understanding the limitations of mathematical reasoning in large language models (2024), Preprint at http://arxiv.org/abs/2410.05229