Protocol to predict mechanical properties of multi-element ceramics using machine learning

Identifying and designing high-performance multi-element ceramics based on trial-and-error approaches are ineffective and expensive. Here, we present a machine-learning-accelerated method for prediction of mechanical properties of multi-element ceramics, based on the density functional theory calculation database. Specific bonding characteristics are used as highly efficient machine learning descriptors. This protocol describes a low-cost, high-efficiency, and reliable workflow for developing advanced ceramics with superior mechanical properties.

Publisher’s note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.

Yunqing Tang, Dong Zhang, Ruiliang Liu, Dongyang Li
yunqing1@ualberta.ca (Y.T.)
dongyang.li@ualberta.ca (D.L.)

Highlights
Accelerate design of high-mechanical-performance ceramics through machine learning
Predict mechanical properties of ceramics using low dimensional descriptors
Obtain mechanical properties of multi-element ceramics from simple ceramics
This protocol is applicable for multiple rock-salt ceramics and WC-type carbides

Tang et al., STAR Protocols 3, 101552
September 16, 2022 © 2022 The Author(s)
https://doi.org/10.1016/j.xpro.2022.101552

OPEN ACCESS
Protocol to predict mechanical properties of multi-element ceramics using machine learning

Yunqing Tang,1,2,* Dong Zhang,1 Ruiliang Liu,1 and Dongyang Li1,3,*

1Department of Chemical & Materials Engineering, University of Alberta, Edmonton, AB T6G 2H5, Canada
2Technical contact
3Lead contact
*Correspondence: yunqing1@ualberta.ca (Y.T.), dongyang.li@ualberta.ca (D.L.)
https://doi.org/10.1016/j.xpro.2022.101552

SUMMARY
Identifying and designing high-performance multi-element ceramics based on trial-and-error approaches are ineffective and expensive. Here, we present a machine-learning-accelerated method for prediction of mechanical properties of multi-element ceramics, based on the density functional theory calculation database. Specific bonding characteristics are used as highly efficient machine learning descriptors. This protocol describes a low-cost, high-efficiency, and reliable workflow for developing advanced ceramics with superior mechanical properties. For complete details on the use and execution of this protocol, please refer to Tang et al. (2021).

BEFORE YOU BEGIN
Operating system and software
In this protocol, VASP is used for density functional theory (DFT) calculations and MATLAB is used for machine-learning calculations, both of which are commercially available, so licenses for the software are required to execute this protocol. This protocol was written for the Linux environment.

Systems to be predicted
The protocol below describes the steps to predict mechanical properties of multi-element rock-salt carbides, nitrides and carbonitrides, which are widely used as industrially significant hard ceramics. This protocol is proven to be also applicable for other rock-salt ceramics like oxides, sulfides, chlorides and fluorides and WC-type carbides (Tang et al., 2021).

Machine-learning descriptors (inputs)
Proper selection of machine-learning descriptors determines the efficiency and accuracy of machine-learning prediction models. Based on the theoretical investigation on the correlation between atomic bond strengths and mechanical properties of ceramics (Tang et al., 2021), three bonding characteristic parameters (average values in a unit cell), which are the sum of bond order contributed to the whole system by a pair of bonded atoms, the bond length, and the net charge of a cation, respectively, are used as the machine-learning descriptors. The descriptors have been theoretically proven effective to reflect strengths of mixed covalent, ionic and metallic bonds in ceramics. The three bonding characteristic parameters can be obtained by DFT calculations. These low dimensional descriptors have been confirmed to work for rock-salt carbides, nitrides, carbonitrides, oxides, sulfides, chlorides and fluorides and WC-type carbides (Tang et al., 2021).

Target properties (outputs)
In this protocol, the outputs are mechanical properties including bulk, shear and Young’s moduli, hardness and brittleness. In the DFT calculation database, polycrystalline bulk, shear and Young’s
moduli are estimated from DFT calculated elastic constants using the Voigt-Reuss-Hill approximation (Hill, 1952). Hardness (Chen et al., 2011) and brittleness (Pugh, 1954) can be estimated with empirical models based on information of bulk and shear moduli. Other moduli related mechanical properties such as Poisson’s ratio and empirical fracture toughness (e.g., Niu et al., 2019) could also be included in the targeted properties. In the present stage, other mechanical properties such as ultimate stress, ultimate strain, anisotropy and thermal expansion cannot be predicted using this protocol.

Train-test set split
The train set and test set are split from the DFT calculation database based on theoretical considerations, following the guideline proposed in our previous work (Tang et al., 2021). For the training of prediction models for rock-salt multi-element carbides, nitrides and carbonitrides, the indispensable subset of the train set includes mechanical and bond properties of mono-carbides and nitrides covering all different concerned cations, which contains the most basic information to learn bond-mechanical behavior correlations for these rock-salt ceramics. The test set only contains information on multi-element rock-salt carbides, nitrides and carbonitrides. Including information on multi-element rock-salt ceramics in the train set should be able to increase the reliability of the machine-learning prediction models; alternatively, using a small train set that only contains the information on mono-carbides and nitrides is also sufficient to obtain reliable machine-learning prediction models.

Machine-learning algorithm
Many machine-learning algorithms have been successfully applied to predict new multi-element ceramics or their properties, such as random forest (Kaufmann et al., 2020), Gaussian-process regression (Tang et al., 2021), and support vector machine (Zhang et al., 2022). In this protocol, the Gaussian-process regression is employed to train the models, which shows a good performance. Other machine-learning algorithms should also be fine for the prediction model training.

KEY RESOURCES TABLE

| REAGEN or RESOURCE | SOURCE | IDENTIFIER |
|--------------------|--------|------------|
| Deposited data     |        |            |
| Structure files of ceramics | Materials project | Rock-salt ceramic: mp-1282 |
|                    |        | WC-type ceramic: mp-1894 |
| DFT calculation database | project GitHub | https://github.com/TangYunqing/HECdesign_moduli |
| Inputs for prediction | project GitHub | https://github.com/TangYunqing/HECdesign_moduli |
| Code for machine-learning | project GitHub | https://github.com/TangYunqing/HECdesign_moduli |
| Software and algorithms | VASP (version 5.4.4) | VASP website: https://www.vasp.at/ |
|                    | VESTA (version 4.6.0) | VESTA website: https://jp-minerals.org/vesta/en/ |
|                    | Chargemol (version 3.5) | Manz and Limas (2016): https://sourceforge.net/projects/ddec/ |
|                    | MATLAB 2020a | MathWorks: https://www.mathworks.com/products/matlab.html |
| Other              | HPC cluster | Compute Canada: https://www.computecanada.ca/ |
|                    | Workstation | Lenovo ThinkStation P720 Workstation |

Alternatives: The open source software Python can be used as an alternative for the commercial software MATLAB.

Alternatives: Other versions of these software having required functions are also applicable for executing this protocol.
Note: Installation packages and detailed installation methods of above software are available from corresponding websites provided in the key resources table.

Note: Detailed tutorials for VASP could be found at https://www.vasp.at/tutorials/latest/.

MATERIALS AND EQUIPMENT

- Hardware
  - DFT calculations
    DFT calculations were implemented on Compute Canada HPC clusters. 64 CPU cores and 1 GB RAM memory per CPU core were used for each task.

  Alternatives: 1. Fewer CPU cores are also fine for the calculations, at the expense of computing time. 2. The computational resource is requested for property calculations for rock-salt ceramics with unit cells (8 atoms per unit cell). More RAM memory is required if properties of multi-element ceramics having large supercells are calculated.

  - Machine-learning modeling
    The machine-learning modeling was implemented on a local computational workstation (operating system: CentOS 7), which has 48 CPU cores and 128 GB RAM memory.

  Alternatives: The machine-learning model training here is a lightweight calculation, fewer CPU cores and less RAM memory are also sufficient to complete the calculations.

  Note: The information on hardware presented here is not a mandatory requirement but a reference for users to obtain an idea about the relationship between the computational resource and consumed calculation time.

STEP-BY-STEP METHOD DETAILS

Build the DFT calculation database

© Timing: 6–7 days

In this section, we conduct DFT calculations to build the database containing mechanical and bond properties of ceramics for further machine-learning model training.

Note: The time for DFT calculations largely depends on sizes of the models, the number of cases involved in the database, and the hardware. This calculation time in this protocol is obtained using 64 CPU cores for 438 cases having 8 atoms in each unit cell, which is a reference of time arrangement for users. In most cases, the model size is the main factor influencing computational cost, so small models such as the unit cell containing 8 atoms shown in Figure 1 are recommended for the DFT calculation database construction.

1. Determine the type of ceramic for mechanical property prediction. In this protocol, the DFT calculation database contains properties of 438 rock-salt carbides, nitrides and carbonitrides, which is available on the project GitHub.

2. Geometry optimization calculations.
   a. Prepare input structure “POSCAR” files. Use the structure file of VC as the template “POSCAR” file for all rock-salt ceramics as shown in Figure 1, replace m and n by metal and nonmetal element names, respectively.
b. Prepare input pseudopotential “POTCAR” files. Taking the structure in Figure 1 as an example, create the “POTCAR” file for this ceramic using the command:

```
> cat m1 m2 m3 m4 n1 n2 n3 n4 >> POTCAR
```

**Note:** “m1”, “m2”, “m3”, “m4”, “n1”, “n2”, “n3”, and “n4” are names of pseudopotential files (provided by VASP) of elements m1, m2, m3, m4, n1, n2, n3 and n4, respectively.

c. Prepare input “KPOINTS” files. Use the “KPOINTS” file as follows:

```
> Automatic mesh
> 0
> Gamma
> 11 11 11
> 0 0 0
```

d. Prepare input “INCAR” files. Use the “INCAR” file as follows:

```
> ENCUT = 600
> ISTART = 0
> ICHARG = 2
> ISMEAR = -5
> SIGMA = 0.05
> NSW = 200
> IBRION = 2
> ISIF = 3
```
e. Put the four input files, namely “POSCAR”, “POTCAR”, “KPONITS” and “INCAR”, in the same dictionary, and then run the calculation using the command:

```bash
> mpirun -n number_of_cores vasp_std >& log
```

3. Bond property calculations.
   a. Prepare input “POSCAR” files. Rename the relaxed structure file “CONTCAR” obtained by the geometry optimization calculation by “POSCAR”.
   b. Prepare input “POTCAR” and “KPONITS” files. Use the same input files with those used in the optimization calculation.
   c. Prepare input “INCAR” files. Use the “INCAR” file as follows:

```bash
>ENCUT = 600
>ISTART = 0
>ICHARG = 2
>ISMEAR = 0
>SIGMA = 0.05
>LREAL = .FALSE.
>SYMPREC = 1E-5
>ispin = 2
>IBRION = -1
>ISIF = 3
>NSW = 0
>POTIM = 0.2
>EDIFF = 1E-5
>PREC = Accurate
>LORBIT = 11
>LAECHG = .TRUE.
>PREC = Accurate
```

d. Put the four input files, namely “POSCAR”, “POTCAR”, “KPONITS” and “INCAR”, in the same dictionary, and then run the calculation using the command:

```bash
> mpirun -n number_of_cores vasp_std >& log
```

e. Put the bond property DFT calculation generated files “AECCAR0”, “AECCAR2” and “CHGCAR” and the corresponding “POTCAR” file, together with the file “job_control.txt” (provided by Chargemol) and the Chargemol executable file “Chargemol_09_26_2017_linux_serial”
in the same directory, and then run the density derived electrostatic and chemical (DDEC) calculation using the command:

```bash
./Chargemol_09_26_2017_linux_serial
```

f. Calculate the bonding characteristic: net charge (NETM).

**Note:** The file “DDEC6_even_tempered_net_atomic_charges.xyz” generated after the DDEC calculation contains the value of net charge on each atom. Taking the DDEC calculation results for VC as an example, Figure 2 shows the first few lines of the “DDEC6_even_tempered_net_atomic_charges.xyz” file, and we can have the average value for metallic (V) atoms in the red box as the net charge for the ceramic VC.

g. Calculate the bonding characteristic: sum of bond order (SBO).

**Note:** The file “DDEC6_even_tempered_bond_orders.xyz” generated after the DDEC calculation contains the information of sum of bond order of each atom. Taking the DDEC calculation results for VC as an example, Figure 3 shows the first few lines of the “DDEC6_even_tempered_bond_orders.xyz” file, and we can use the sum of average values respectively for V and C atoms in the red boxes as the value of sum of bond order for the ceramic VC.

h. Calculate the bonding characteristic: bond length (BL).

**Note:** The average bond length of the relaxed structure equals to \( \sqrt{V/8} \), where \( V \) is the volume of the relaxed unit cell, which is given by VESTA as shown in Figure 4 after reading the “CONTCAR” file obtained by the geometry optimization calculation.

4. Mechanical property calculations.
   a. Prepare input “POSCAR”, “POTCAR” and “KPONITS” files. Use the same input files with those used in the optimization calculation.
   b. Prepare input “INCAR” files. Use the “INCAR” file as follows:

```plaintext
>ENCUT = 600
>ISTART = 0
>ICHARG = 2
>SYMPREC = 1E-5
>ispin = 2
>ISMEAR = 0
>SIGMA = 0.05
>POTIM = 0.05
>LCHARG = .TRUE.
>LWAVE = .TRUE.
>LREAL = .FALSE.
>IBRION = 6
>ISIF = 3
>NFS = 1
>NFREE = 4
```
c. Put the four input files, namely “POSCAR”, “POTCAR”, “KPOINTS” and “INCAR”, in the same dictionary, and then run the calculation using the command:

```
> mpirun -n number_of_cores vasp_std > & log
```

d. Calculate mechanical properties: bulk ($B$), shear ($G$) and Young’s ($E$) moduli.

**Note:** The “OUTCAR” file obtained by the mechanical property DFT calculation contains the elastic constants, based on which we can calculate the bulk, shear and Young’s moduli using the Voigt-Reuss-Hill approximation (Hill, 1952).

**Alternatives:** The commands in steps 1–4 are for case-by-case calculations. If many cases are involved in a dataset, we can prepare all input files for a case in one dictionary and run a series of jobs for all cases automatically using the script “job.sh” by: `> sh job.sh`, as is shown in Figure 5.

**Alternatives:** The command “mpirun” is used to start the parallel version of VASP compiled with MPI. To run a serial version of VASP, it is needed to replace the “mpirun” line by “> vasp_std”. The parallel version of VASP is recommended for the higher calculation efficiency.

**Note:** In this example, 438 cases are involved in the dataset. For the practical application, a small dataset containing information on simple ceramics is sufficient to build reliable machine-learning models. E.g., involving information on rock-salt ScC, TiC, VC, CrC, MnC, FeC, CoC, NiC, ZnC, YC, ZrC, NbC, MoC, HfC, TaC, WC carbides into the DFT dataset is sufficient to build reliable machine-learning models for complex rock-salt (ScTiVCrMnFeCoNiZnYrZrNbMoHfTaW) C carbides.

**Note:** The commands and calculations are implemented under Linux environment.

**Note:** The parameters in “INCAR” and “KPOINTS” files are the same for different cases for the same kind of DFT calculations.

### Figure 2. Results of the net charge calculation

```
$ jmolscript: load "" [1 1 l] spacegroup "x,y,z" unitcell [{ 4.157101 0.000000 0.000000 }, { 0.000000 4.157101 0.000000 },
{ 0.000000 0.000000 4.157101 }]
C 2.078550 2.078550 2.078550 -1.486069
C 2.078550 2.078550 0.000000 -1.486069
C 0.000000 2.078550 0.000000 -1.486069
C 0.000000 0.000000 2.078550 -1.486069
V 0.000000 0.000000 0.000000 1.486069
V 0.000000 2.078550 2.078550 1.486069
V 2.078550 0.000000 2.078550 1.486069
V 2.078550 2.078550 0.000000 1.486069

NetM = 1.486069

Chargemol version 3.2 September 26, 2017.
See ddec.sourceforge.net for latest version.
```
Note: To build new DFT datasets for other rock-salt ceramics like oxides, sulfides, chlorides and fluorides, no significant modifications are required, simply replacing m and n in steps 2b and 2c by specific elements will work.

Note: To build new DFT datasets for WC-type carbides, modifications of input files and the calculation method for average bond length are needed.
Train machine-learning prediction models

© Timing: ~30 min

In this section, we train the machine-learning models to learn correlations between bonding characteristics and mechanical properties based on the DFT calculation database. The dataset for machine-learning training, the training codes and the obtained prediction models are all available on the project GitHub.

5. Prepare the dataset for machine-learning model training. Store the data in a .csv file as shown in Figure 6 and then load the .csv file to MATLAB with the name “trainingData”.

6. Prepare the training code. Taking the training of machine-learning prediction model for bulk modulus as an example, use the training code saved in the file “trainModelB.m” as follows:

```matlab
function [ModelB, validationRMSE] = trainModelB(trainingData)

inputTable = trainingData;
predictorNames = {'SBO', 'NETM', 'BL'};
predictors = inputTable(:, predictorNames);
response = inputTable.B;
isCategoricalPredictor = [false, false, false];

> predictors, ...
> response, ...
> 'BasisFunction', 'constant', ...
> 'KernelFunction', 'exponential', ...
```
7. Perform the machine-learning training. Taking the training of machine-learning prediction model for bulk modulus as an example, put the training code file “trainModelB.m” and the training dataset file “trainingData” in the same workspace, and then run the command in MATLAB as follows to obtain the machine-learning prediction model “ModelB”:

```matlab
> [ModelB, validationRMSE] = trainModelB(trainingData)
```

**Note:** As the brittleness and hardness estimated using empirical models have intrinsic analytic relations with the shear and bulk moduli, here only mechanical properties of bulk, shear and Young’s moduli are set as targeted outputs.

**Note:** Machine-learning prediction models for shear and Young’s modulus can be obtained by respectively replacing the keyword “B” in the code by “G” and “E” in steps 6 and 7.
**Prepare the inputs for prediction**

@ Timing: ~30 min

Bonding characteristics of a multi-element ceramic are the inputs to predict its mechanical properties based on machine-learning prediction models. In this section, we calculate bond parameters of multi-element ceramics based on those of simple ceramics from the DFT dataset.

8. Calculate bonding characteristics of multi-element ceramics.

   **Note:** Bonding characteristics of multi-element carbides, nitrides and carbonitrides equals to the weighted value of those of involved constituents (mono-carbides and nitrides) according to their atomic concentrations (Tang et al., 2021). For instance, for the multi-element rock-salt carbonitride (VNbTaW) (CN), its bonding characteristics are average values of corresponding bonding characteristics of VC, NbC, TaC, WC, VN, NbN, TaN and WN.

9. Collect bonding characteristics of all concerned multi-element rock-salt carbides, nitrides and carbonitrides in a .csv file and then load the .csv file into MATLAB by the name of “predictioninputs”.

   **Note:** The sum of bond order, net charge and bond length is respectively labeled as “SBO”, “NETM” and “BL”. The prediction inputs dataset is available on the project GitHub.

**Build the machine-learning prediction database**

@ Timing: ~10 min

Using machine-learning prediction models and empirical models, we predict mechanical properties of multi-element ceramics based on bonding characteristics to build the mechanical property database for all potential multi-element ceramics.

10. Predict mechanical properties: bulk, shear and Young’s moduli. Put the prediction inputs file “predictioninputs” and the prediction model files “ModelB”, “ModelG” and “ModelE” in the same workspace, and then run the commands as follows in MATLAB to get bulk, shear and Young’s moduli stored in “fitB”, “fitG” and “fitE”, respectively:

    ```
    >>fitB = ModelB.predictFcn(predictioninputs)
    >>fitG = ModelG.predictFcn(predictioninputs)
    >>fitE = ModelE.predictFcn(predictioninputs)
    ```

11. Calculate mechanical properties: brittleness and hardness.
Note: These two properties are estimated by empirical models based on predicted bulk and shear moduli. Specifically, the brittleness equals to $G/B$ (Pugh, 1954), the hardness equals to $2(k^2G)^{0.585} - 3$ (Chen et al., 2011).

12. Collect information on rock-salt carbides, nitrides and carbonitrides, including their names and predicted bulk, shear and Young’s moduli, brittleness and hardness, in one Excel file to form the machine-learning prediction database.

EXPECTED OUTCOMES

In this protocol, we describe a step-by-step workflow to predict mechanical properties for multi-element ceramics combining DFT calculations and machine-learning. Rock-salt carbides, nitrides and carbonitrides are taken as examples for the construction of a machine-learning prediction database for their mechanical properties. Figure 7 shows the distribution of Pugh’s modulus ratio (brittleness) along with the changes in hardness and Young’s modulus based on the machine-learning prediction data, from which ceramics with desired combination of mechanical properties can be customized by directly picking ceramics from the distribution map, and the relationships among different mechanical properties can be investigated for understanding the mechanism and establishing design guidelines. This strategy should be helpful to accelerate developing high-performance multi-element ceramics with superior mechanical properties and identify key factors that govern the mechanical properties of ceramics.

LIMITATIONS

There are several limitations for this protocol. Firstly, the applicability of this protocol is dependent on the understanding of the bond-mechanical property correlations for specific ceramics. Currently, this strategy has been proven working for two kinds of industrially significant ceramics: rock-salt ceramics and WC-type carbides, but the applicability for other ceramics needs to be validated. Secondly, only one machine-learning algorithm (Gaussian-process regression) is illustrated here because the critical factor determining the effectiveness of prediction model is the selection of machine-learning descriptors. More different machine-learning algorithms could be tried for better prediction accuracy. Thirdly, since some mechanical properties in the prediction database are deduced from elastic moduli based on empirical models, the accuracy and reliability of these data are limited by the performance of applied empirical models. Fourthly, the bonding characteristics derived descriptors used in this protocol are applicable to ceramics in which all anions or cations have equivalent atomic occupations. Further investigations are needed to modify the bonding...
characteristics derived descriptors for the application in ceramics having more complex structures. Fifthly, the validation steps are not included in this protocol. Validations of the machine-learning models for applicable systems (rock-salt ceramics and WC-type carbides) have been done in our previous work (Tang et al., 2021), where the detailed data and analysis are available.

**TROUBLESHOOTING**

**Problem 1**
The codes or commands used in this protocol do not work when reproducing the workflow. (steps 2–4, 6–7, 10).

**Potential solution**
The codes or commands may be different for different versions of software or in different operating systems. Please make sure all software are properly installed, and modify the codes or commands according to the versions of software and the operating system or use the same versions of software and operating system with this protocol (see key resources table and materials and equipment).

**Problem 2**
The DFT calculations are not convergent. (steps 2–4).

**Potential solution**
A divergent DFT calculation, for either ionic relaxation loop or electronic self-consistency loop, will lead to unreliable structure or properties, so the convergence of DFT calculation needs to be checked after the calculation. Parameters in “INCAR” or in “KPOINTS” need to be modified if the DFT calculation is not convergent. Usually decreasing the k-point density (the fourth line of the file) in “KPOINTS” or increasing the SYMPREC item in “INCAR” will help solve the problem.

**Problem 3**
Descriptors in this protocol may not be appropriate for property predictions for ceramics other than rock-salt ceramics or WC-type carbides. (steps 3, 6 and 8).

**Potential solution**
Two validations need to be checked for the applicability of descriptors for new systems:

- The applicable systems should have positively correlated distributions for relationships of Young’s moduli – SBO x NETM/BL, shear moduli – SBO x NETM/BL, and bulk moduli – SBO/BL (Tang et al., 2021). Therefore, a small DFT dataset containing simple ceramics could be firstly built to check if bond-mechanical property correlations of the new system meet the theoretical expectation.
- The relationship between bonding characteristics of multi-element ceramics and those of involved constituents also need to be validated (Tang et al., 2021). Bonding characteristics (SBO, NETM and BL) of multi-element ceramics (e.g., ZnCuGaO) should be weighted from those of involved constituents (e.g., ZnO, CuO, GaO) based on their atomic concentrations (Tang et al., 2021).

**Problem 4**
Input files to build DFT dataset in this protocol are for rock-salt ceramics but not for WC-type carbides. (steps 2 and 3).

**Potential solution**
To build a new DFT dataset for WC-type carbides, the workflow of this protocol still works, but some input files and the method to calculate average bond length need to be modified.

For step 2a, the template “POSCAR” file should be changed to the 2 x 2 x 1 supercell based on unit cell of WC (Materials project ID: mp-1894).
For step 2c, the fourth line of the “KPOINTS” file should be changed from “11 11 11” to “9 9 11”.

For step 3h, the calculation method for the average bond length of the relaxed structure should be changed to $0.5036 \times \sqrt{V}$, where $V$ is the volume of the relaxed unit cell.

Problem 5
The RMSE is not sufficient for validations for ceramics other than rock-salt ceramics or WC-type carbides. (step 7).

Potential solution
Details about validation of the machine-learning prediction models for rock-salt ceramics or WC-type carbides are provided in our previous work (Tang et al., 2021), in which RMSE, coefficient of determination (R-Squared) and mean absolute error (MAE) are calculated to estimate reliabilities of machine-learning models. According to the experience obtained from modeling for rock-salt ceramics and WC-type carbides, R-Squared higher than 0.8 is sufficient to guarantee reliabilities of machine-learning models. The validation based on theoretically guided train and test set split is required to estimate the performance and avoid the bias of prediction models (Train-test set split, Tang et al., 2021), in which information on simple ceramics needs to be included in the train set while information on complex ceramics needs to be included in the test set. RMSE, R-Squared and MAE can be calculated in MATLAB as follows:

```matlab
> RMSE = sqrt(mean((YPred-YCalc).^2))
> r2 = 1-(sum((YPred-YCalc).^2)/sum((YReal-mean(YCalc)).^2))
> MAE = mean(abs(YCalc-YPred))
```

in which “YCalc” contains calculated mechanical properties, “YPred” contains mechanical properties predicted by machine-learning models.

**RESOURCE AVAILABILITY**

**Lead contact**
Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Dongyang Li (dongyang.li@ualberta.ca).

**Materials availability**
This study did not generate new, unique reagents.

**Data and code availability**
The code and data to reproduce the results of this work can be found in the following:

GitHub page: https://github.com/TangYunqing/HECdesign_moduli (10.5281/zenodo.6610882).

**ACKNOWLEDGMENTS**
The authors are grateful for support from the Natural Science and Engineering Research Council of Canada, Compute Canada, Weir Minerals, and Alberta Innovates. We also would like to thank Qing-feng Liu for the assistance in processing data.

**AUTHOR CONTRIBUTIONS**
Y.T. and D.L. conceived the research idea. Y.T., D.Z., and R.L. performed the DFT simulations. Y.T. carried out machine-learning modeling and wrote the manuscript. D.L. supervised the project and revised the manuscript. All authors discussed the results and made suggestions for the manuscript.
DECLARATION OF INTERESTS
The authors declare no competing interests.

REFERENCES

Chen, X.Q., Niu, H., Li, D., and Li, Y. (2011). Modeling hardness of polycrystalline materials and bulk metallic glasses. Intermetallics 19, 1275–1281. https://doi.org/10.1016/j.intermet.2011.03.026.

Hill, R. (1952). The elastic behaviour of a crystalline aggregate. Proc. Phys. Soc. A 65, 349–354. https://doi.org/10.1088/0370-1298/65/5/307.

Kaufmann, K., Maryanovsky, D., Mellor, W.M., Zhu, C., Rosengarten, A.S., Harrington, T.J., Oses, C., Toher, C., Curtarolo, S., and Vecchio, K.S. (2020). Discovery of high-entropy ceramics via machine learning. NPJ Comput. Mater. 6, 1–9. https://doi.org/10.1038/s41524-020-0317-6.

Manz, T.A., and Limas, N.G. (2016). Introducing DDEC6 atomic population analysis: part 1. Charge partitioning theory and methodology. RSC Adv. 6, 47771–47801. https://doi.org/10.1039/c6ra04666h.

Niu, H., Niu, S., and Oganov, A.R. (2019). Simple and accurate model of fracture toughness of solids. J. Appl. Phys. 125, 065105. https://doi.org/10.1063/1.5066311.

Pugh, S.F. (1954). XCII. Relations between the elastic moduli and the plastic properties of polycrystalline pure metals. Lond. Edinb. Dublin Philos. Mag. J. Sci. 45, 823–843. https://doi.org/10.1080/1478644080520496.

Tang, Y., Zhang, D., Liu, R., and Li, D. (2021). Designing high-entropy ceramics via incorporation of the bond-mechanical behavior correlation with the machine-learning methodology. Cell Rep. Phys. Sci. 2, 100640. https://doi.org/10.1016/j.xcrp.2021.100640.

Zhang, J., Xu, B., Xiong, Y., Ma, S., Wang, Z., Wu, Z., and Zhao, S. (2022). Design high-entropy carbide ceramics from machine learning. NPJ Comput. Mater. 8, 1–12. https://doi.org/10.1038/s41524-021-00678-3.