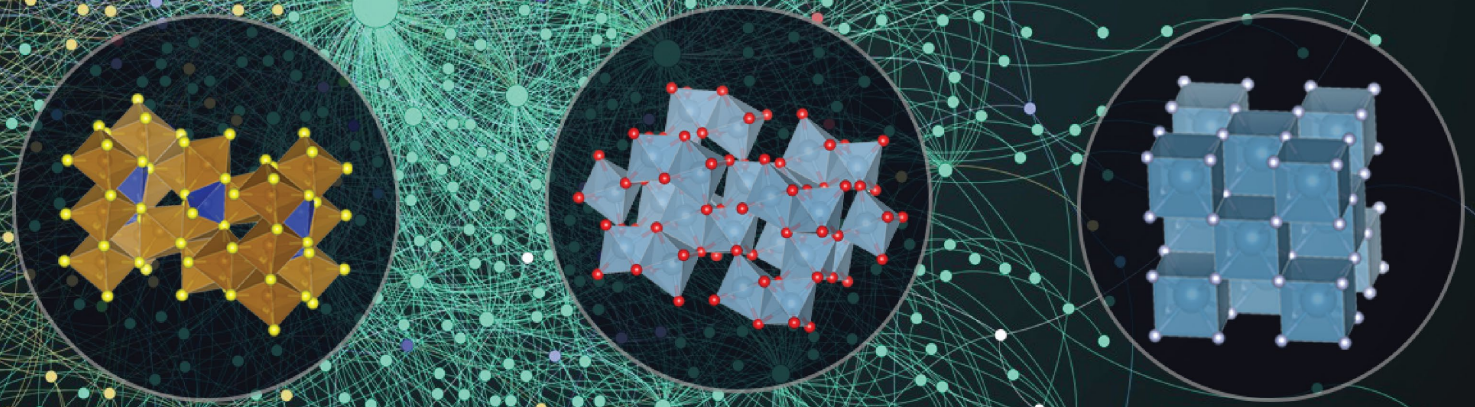
Composition-based feature vector



How do we featurize a chemical formula for machine learning?

Al_2O_3

$[C_2F_4]_n$

Cu_2Zn

How do we featurize a chemical formula for machine learning?

Al_2O_3

ceramic, hard, high melting point, strong

$[C_2F_4]_n$

polymer, non-stick, low melting point, low friction

Cu_2Zn

metal, alloy, intermediate melting point, ductile, corrosion resistant

We use element properties to generate a Composition-Based Feature Vector (CBFV)

Element	Number	Atomic_Weight	Period	Group	Families	Metal	Nonmetal	Metalliod	Atomic_Radius
H	1	1.00794	1	1	7	0	1	0	0.53
He	2	4.002602	1	18	9	0	1	0	0.31
Li	3	6.941	2	1	1	1	0	0	1.67
Be	4	9.01218	2	2	2	1	0	0	1.12
B	5	10.811	2	13	6	0	0	1	0.87
C	6	12.011	2	14	7	0	1	0	0.67
N	7	14.00674	2	15	7	0	1	0	0.56
O	8	15.9994	2	16	7	0	1	0	0.48
F	9	18.998403	2	17	8	0	1	0	0.42
Ne	10	20.1797	2	18	9	0	1	0	0.38
Na	11	22.989768	3	1	1	1	0	0	1.9
Mg	12	24.305	3	2	2	1	0	0	1.45
Al	13	26.981539	3	13	5	1	0	0	1.18
Si	14	28.0855	3	14	6	0	1	0	1.11

Unique formulae end up with unique vector representations

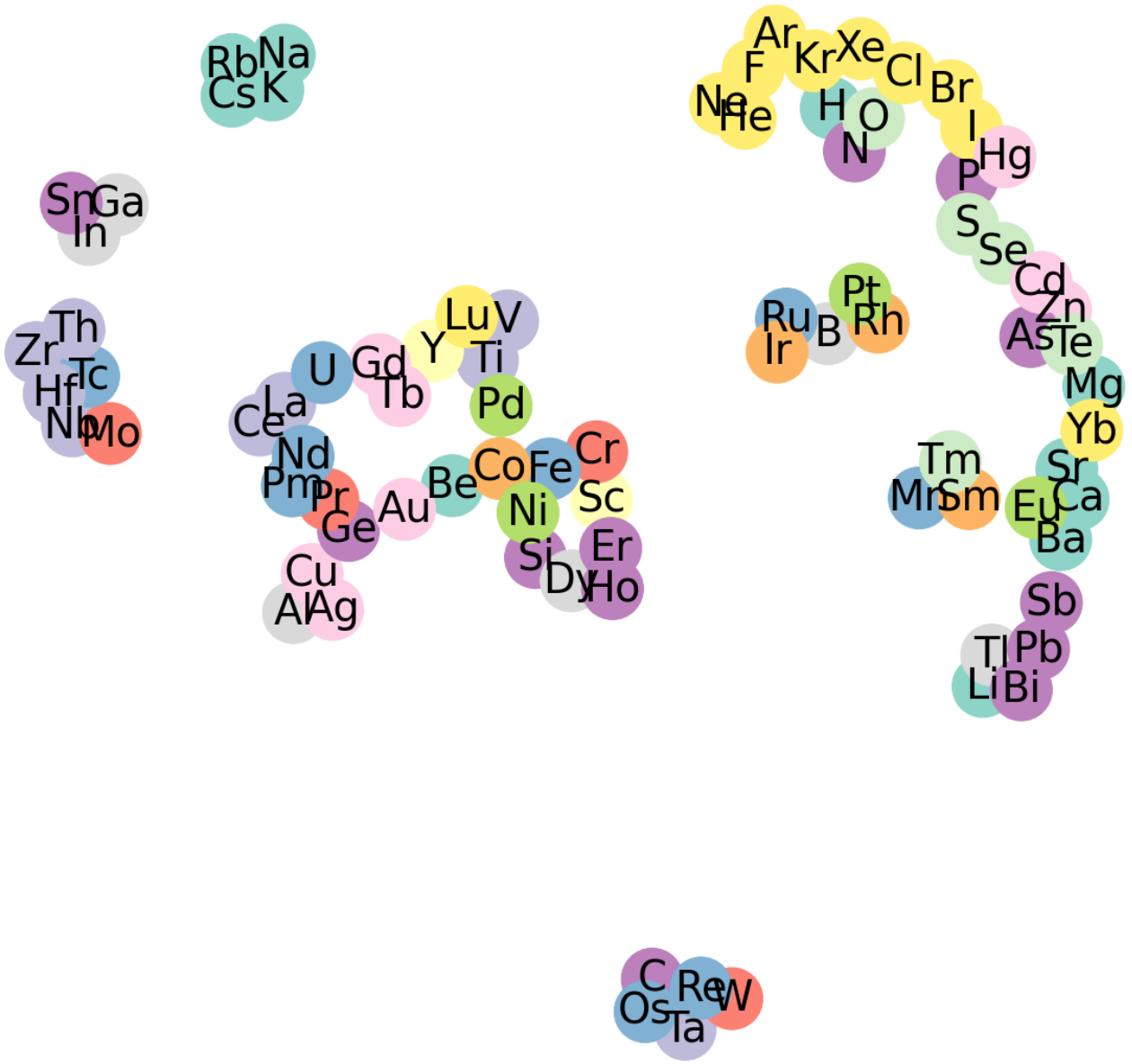
$$Al_2O_3 \rightarrow Al_{0.4}O_{0.6}$$

C	6	12.011	2	14	7	0	1	0	0.67
N	7	14.00674	2	15	7	0	1	0	0.56
O	8	15.9994	2	16	7	0	1	0	0.48
F	9	18.998403	2	17	8	0	1	0	0.42
Ne	10	20.1797	2	18	9	0	1	0	0.38
Na	11	22.989768	3	1	1	1	0	0	1.9
Mg	12	24.305	3	2	2	1	0	0	1.45
Al	13	26.981539	3	13	5	1	0	0	1.18

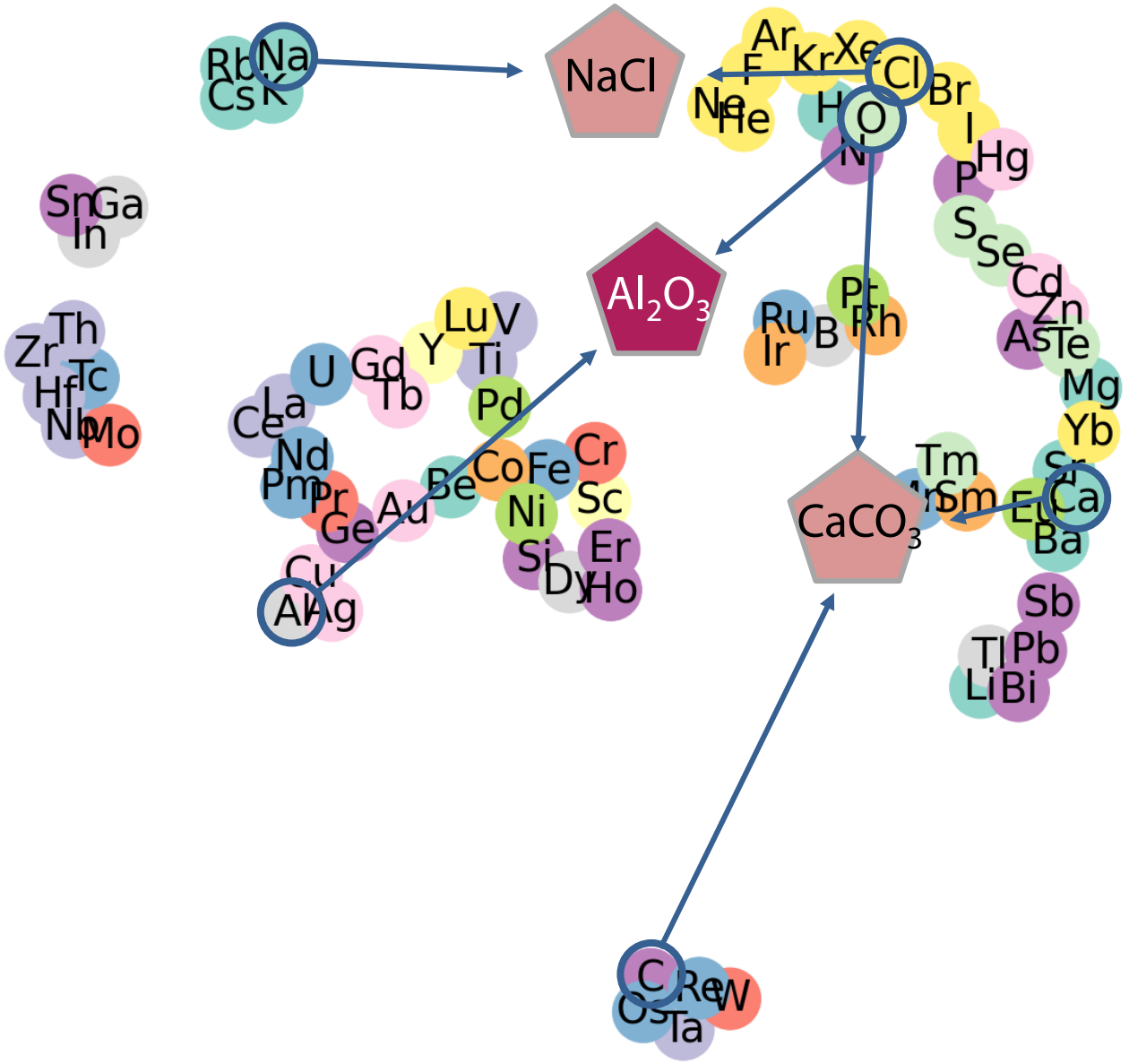
$$Al_{0.4}O_{0.6} = [8 \times 0.6 + 13 \times 0.4,\ 16 \times 0.6 + 27 \times 0.4,\ \ldots,\ 0.48 \times 0.6 + 1.18 \times 0.4]$$

$$Al_{0.4}O_{0.6} = [10,\ 20.4,\ \ldots,\ 3.35]$$

Projecting this CBFV into reduced dimensions we see cluster of chemically similar materials

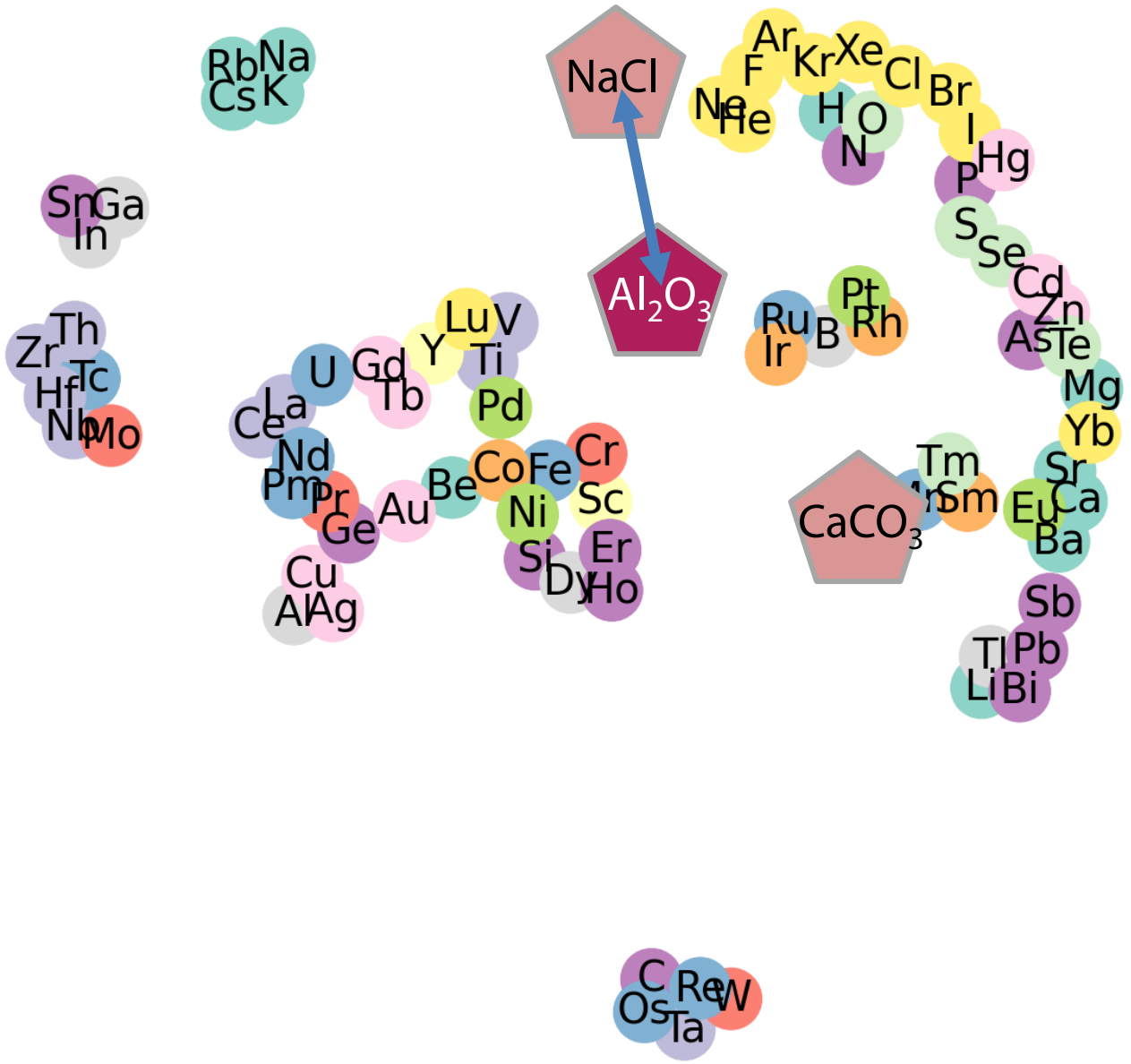


Projecting this CBFV into reduced dimensions we see cluster of chemically similar materials



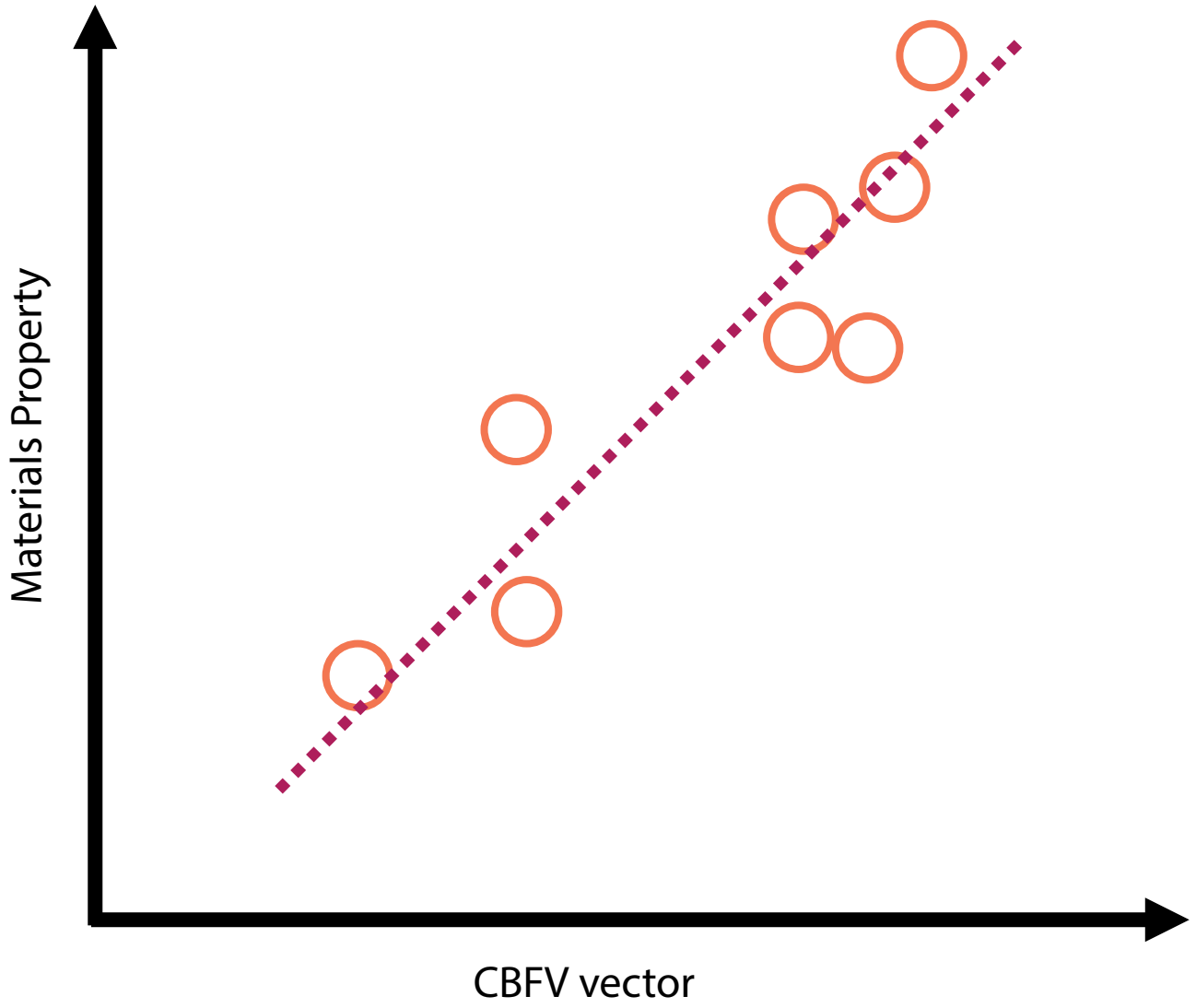
Compounds appear at intermediate positions based on pure elemental constituents

Compounds near to one another should share similar properties



Nearest Neighbor:
Find closest datapoint
that already has label

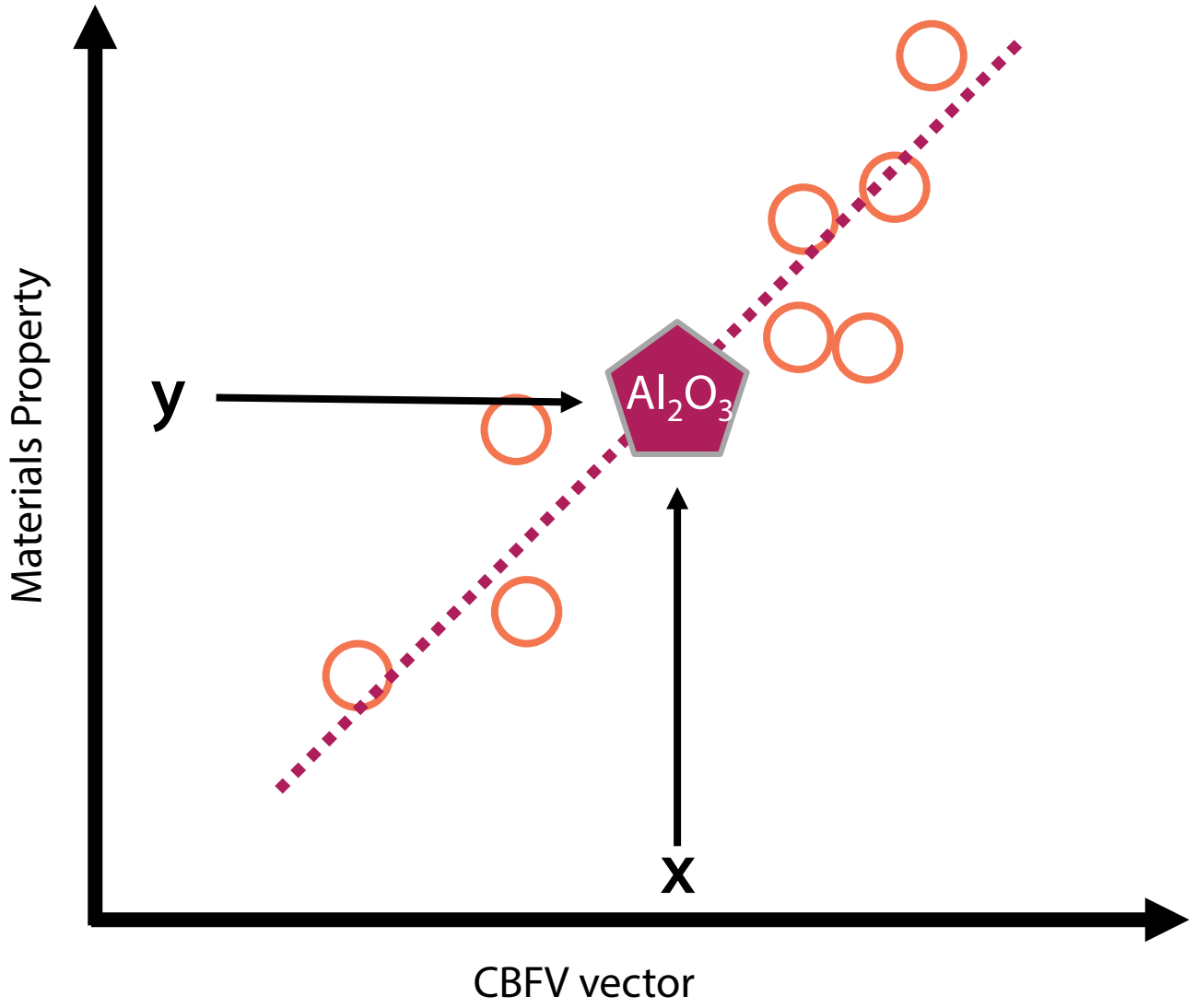
Many different models and algorithms available for prediction



$$y = mx + b$$

Fit line to known data

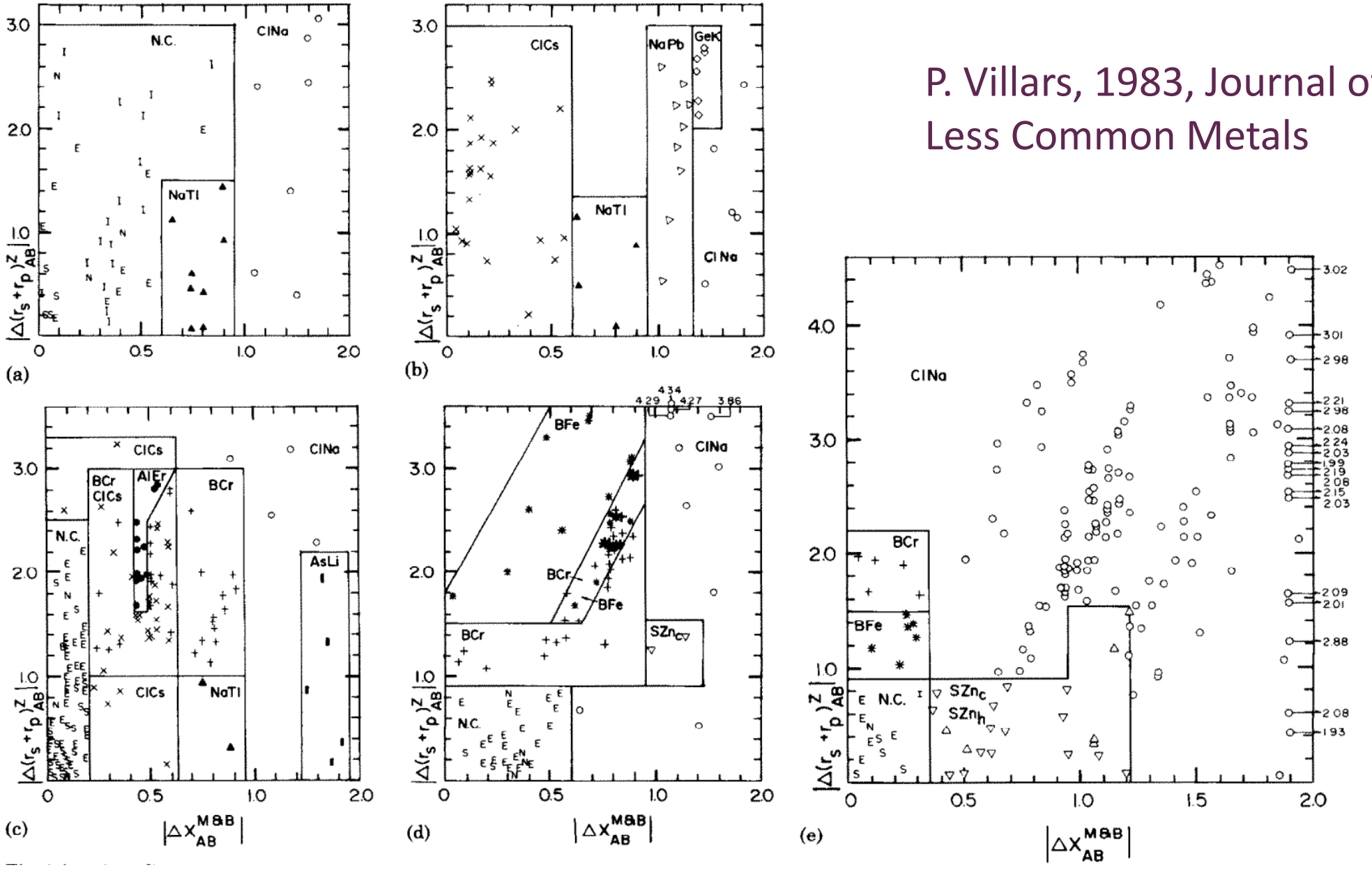
Many different models and algorithms available for prediction



$$y = mx + b$$

1. Given a new **x**
2. Generate a prediction **y**

Villars developed chemical descriptors long before materials informatics!



P. Villars, 1983, Journal of Less Common Metals

There are many different elemental property databases used for CBFV

JARVIS

Oliynyk

materialsintelligence/
mat2vec

Supplementary Materials for Tshitoyan et al. "Unsupervised word embeddings capture latent knowledge from materials science literature", Nature (2019).

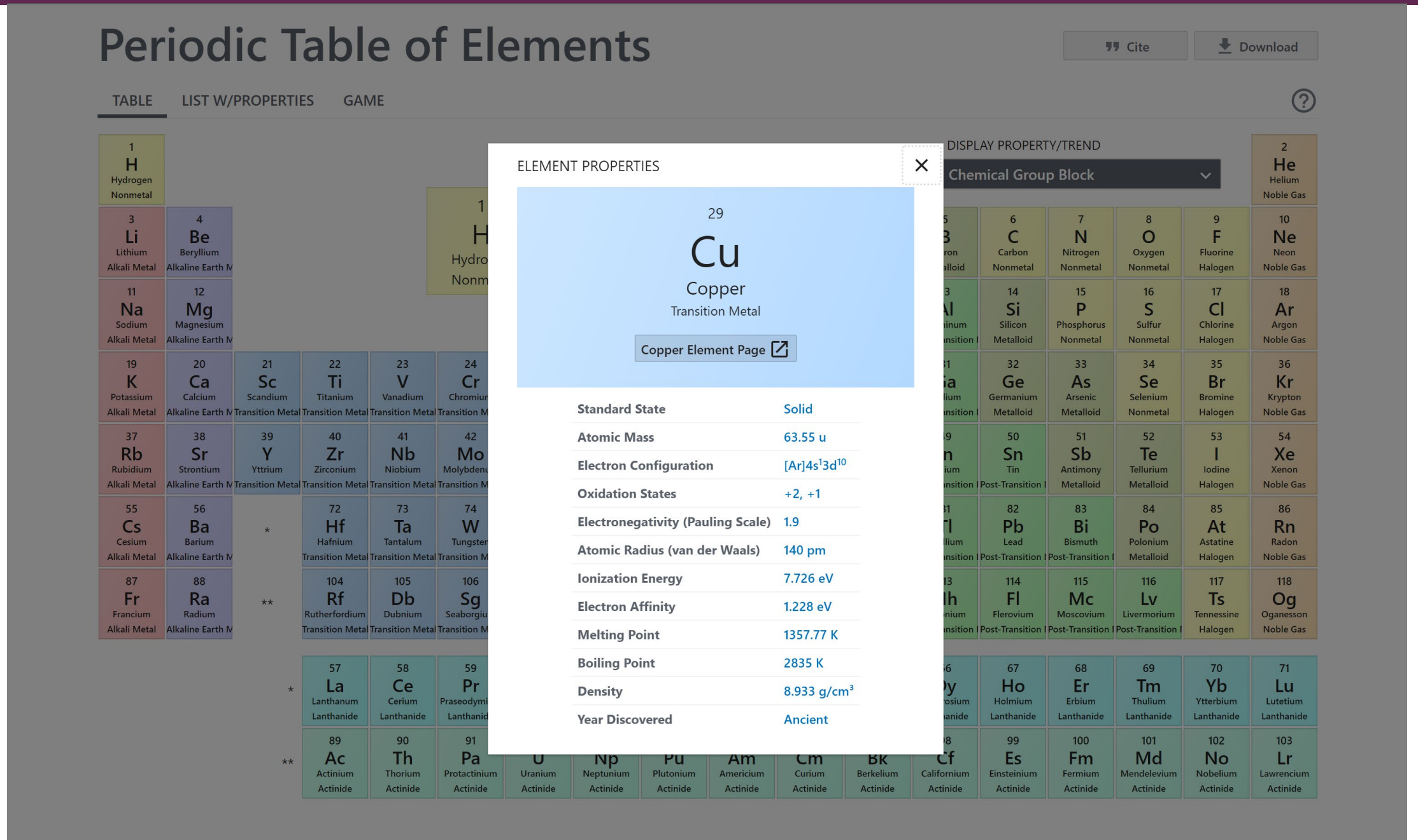
idocx/**Atom2Vec**

A python implement of Atom2Vec: a simple way to describe atoms for machine learning

MAGPIE

Materials Agnostic Platform for Informatics and Exploration

JARVIS, MAGPIE, and Oliynyk are all based off of elemental property databases



mat2vec extracts chemical domain knowledge from journal abstracts via word embeddings

LETTER

https://doi.org/10.1038/s41586-019-1335-8

Unsupervised word embeddings capture latent knowledge from materials science literature

Vahe Tshitoyan[1,3]*, John Dagdelen[1,2], Leigh Weston[1], Alexander Dunn[1,2], Ziqin Rong[1], Olga Kononova[2], Kristin A. Persson[1,2], Gerbrand Ceder[1,2]* & Anubhav Jain[1]*

The overwhelming majority of scientific knowledge is published as text, which is difficult to analyse by either traditional statistical analysis or modern machine learning methods. By contrast, the main source of machine-interpretable data for the materials research community has come from structured property databases[1,2], which encompass only a small fraction of the knowledge present in the research literature. Beyond property values, publications contain valuable knowledge regarding the connections and relationships between data items as interpreted by the authors. To improve the identification and use of this knowledge, several studies have focused on the retrieval of information from scientific literature using supervised natural language processing[3–10], which requires large hand-labelled datasets for training. Here we show that materials science knowledge present in the published literature can be efficiently encoded as information-dense word embeddings[11–13] (vector representations of words) without human labelling or supervision. Without any explicit insertion of chemical knowledge, these embeddings capture complex materials science concepts such as the underlying structure of the periodic table and structure–property relationships in materials. Furthermore, we demonstrate that an unsupervised method can recommend materials for functional applications several years before their discovery. This suggests that latent knowledge regarding future discoveries is to a large extent embedded in past publications. Our findings highlight the possibility of extracting knowledge and relationships from the massive body of scientific literature in a collective manner, and point towards a generalized approach to the mining of scientific literature.

Assignment of high-dimensional vectors (embeddings) to words in a text corpus in a way that preserves their syntactic and semantic relationships is one of the most fundamental techniques in natural language processing (NLP). Word embeddings are usually constructed using machine learning algorithms such as GloVe[13] or Word2vec[11,12], which use information about the co-occurrences of words in a text corpus. For example, when trained on a suitable body of text, such methods should produce a vector representing the word 'iron' that is closer by cosine distance to the vector for 'steel' than to the vector for 'organic'. To train the embeddings, we collected and processed approximately 3.3 million scientific abstracts published between 1922 and 2018 in more than 1,000 journals deemed likely to contain materials-related research, resulting in a vocabulary of approximately 500,000 words. We then applied the skip-gram variation of Word2vec, which is trained to predict context words that appear in the proximity of the target word as a means to learn the 200-dimensional embedding of that target word, to our text corpus (Fig. 1a). The key idea is that, because words with similar meanings often appear in similar contexts, the corresponding embeddings will also be similar. More details about the model are included in the Methods and in Supplementary Information sections S1 and S2, where we also discuss alternative algorithm options such as GloVe. We find that, even though no chemical information or interpretation is added to the algorithm, the obtained word embeddings behave consistently with chemical intuition when they are combined using various vector operations (projection, addition, subtraction). For example, many words in our corpus represent chemical compositions of materials, and the five materials most similar to $LiCoO_2$ (a well-known lithium-ion cathode compound) can be determined through a dot product (projection) of normalized word embeddings. According to our model, the compositions with the highest similarity to $LiCoO_2$ are $LiMn_2O_4$, $LiNi_{0.5}Mn_{1.5}O_4$, $LiNi_{0.8}Co_{0.2}O_2$, $LiNi_{0.8}Co_{0.15}Al_{0.05}O_2$ and $LiNiO_2$—all of which are also lithium-ion cathode materials.

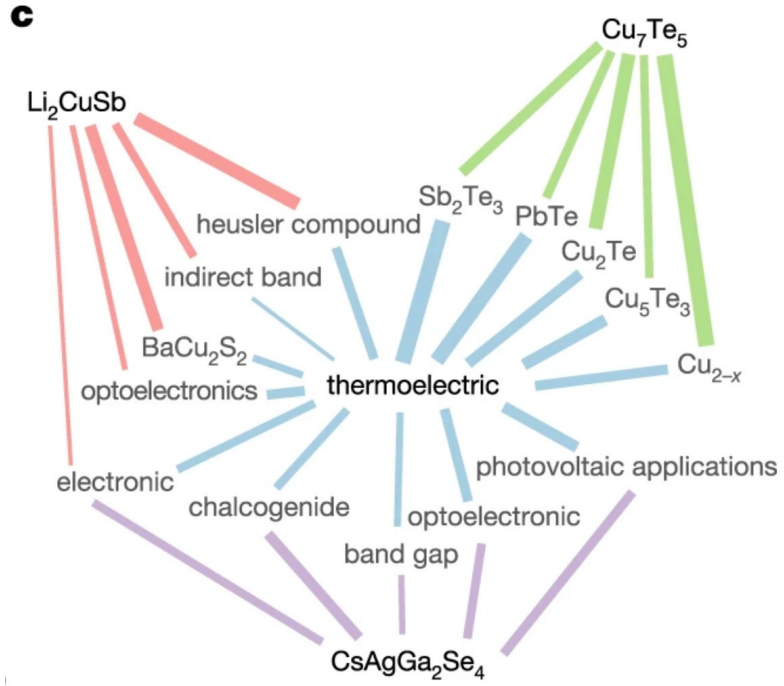
Similar to the observation made in the original Word2vec paper[11], these embeddings also support analogies, which in our case can be domain-specific. For instance, 'NiFe' is to 'ferromagnetic' as 'IrMn' is to '?', where the most appropriate response is 'antiferromagnetic'. Such analogies are expressed and solved in the Word2vec model by finding the nearest word to the result of subtraction and addition operations between the embeddings. Hence, in our model,

$$\text{ferromagnetic} - \text{NiFe} + \text{IrMn} \approx \text{antiferromagnetic}$$

To better visualize such embedded relationships, we projected the embeddings of Zr, Cr and Ni, as well as their corresponding oxides and crystal structures, onto two dimensions using principal component analysis (Fig. 1b). Even in reduced dimensions, there is a consistent operation in vector space for the concepts 'oxide of' ($Zr - ZrO_2 \approx Cr - Cr_2O_3 \approx Ni - NiO$) and 'structure of' ($Zr - HCP \approx Cr - BCC \approx Ni - FCC$). This suggests that the positions of the embeddings in space encode materials science knowledge such as the fact that zirconium has a hexagonal close packed (HCP) crystal structure under standard conditions and that its principal oxide is ZrO_2. Other types of materials analogies captured by the model, such as functional applications and crystal symmetries, are listed in Extended Data Table 1. The accuracies for each category are close to 50%—similar to the baseline set in the original Word2vec study[12]. We stress that Word2vec treats these entities simply as strings, and no chemical interpretation is explicitly provided to the model; rather, materials knowledge is captured through the positions of the words in scientific abstracts. Notably, we also found that embeddings of chemical elements are representative of their positions in the periodic table when projected onto two dimensions (Extended Data Fig. 1a, b, Supplementary Information sections S4 and S5) and can serve as effective feature vectors in quantitative machine learning models such as formation energy prediction—outperforming several previously reported curated feature vectors (Extended Data Fig. 1c, d, Supplementary Information section S6).

The main advantage and novelty of this representation, however, is that application keywords such as 'thermoelectric' have the same representation as material formulae such as 'Bi_2Te_3.' When the cosine similarity of a material embedding and the embedding of 'thermoelectric' is high, one might expect that the text corpus necessarily includes abstracts reporting on the thermoelectric behaviour of this material[14,15]. However, we found that a number of materials that have relatively high cosine similarities to the word 'thermoelectric' never

[1]Lawrence Berkeley National Laboratory, Berkeley, CA, USA. [2]Department of Materials Science and Engineering, University of California, Berkeley, CA, USA. [3]Present address: Google LLC, Mountain View, CA, USA. *e-mail: vahe.tshitoyan@gmail.com; gceder@lbl.gov; ajain@lbl.gov



ABSTRACT: $Cu_{9.1}Te_4Cl_3$ is a new polymorphic compound in the class of coinage metal polytelluride halides. Copper is highly mobile, which results in multiple order–disorder phase transitions in a limited temperature interval from 240 to 370 K. Mainly as a consequence of thermal transport properties, the compound's thermoelectric figure of merit reaches values up to ZT = 0.15 in the temperature range between room temperature and 523 K. Its structure is closely related to that of $Ag_{10}Te_4Br_3$, another coinage metal polytelluride halide, which represents the first p–n–p-switchable semiconductor approachable by a simple temperature change. The title compound outperforms $Ag_{10}Te_4Br_3$ in terms of thermoelectric properties by 1 order of magnitude and therefore acts as a link between the class of p–n–p compounds and thermoelectric

mat2vec extracts chemical domain knowledge from journal abstracts via word embeddings

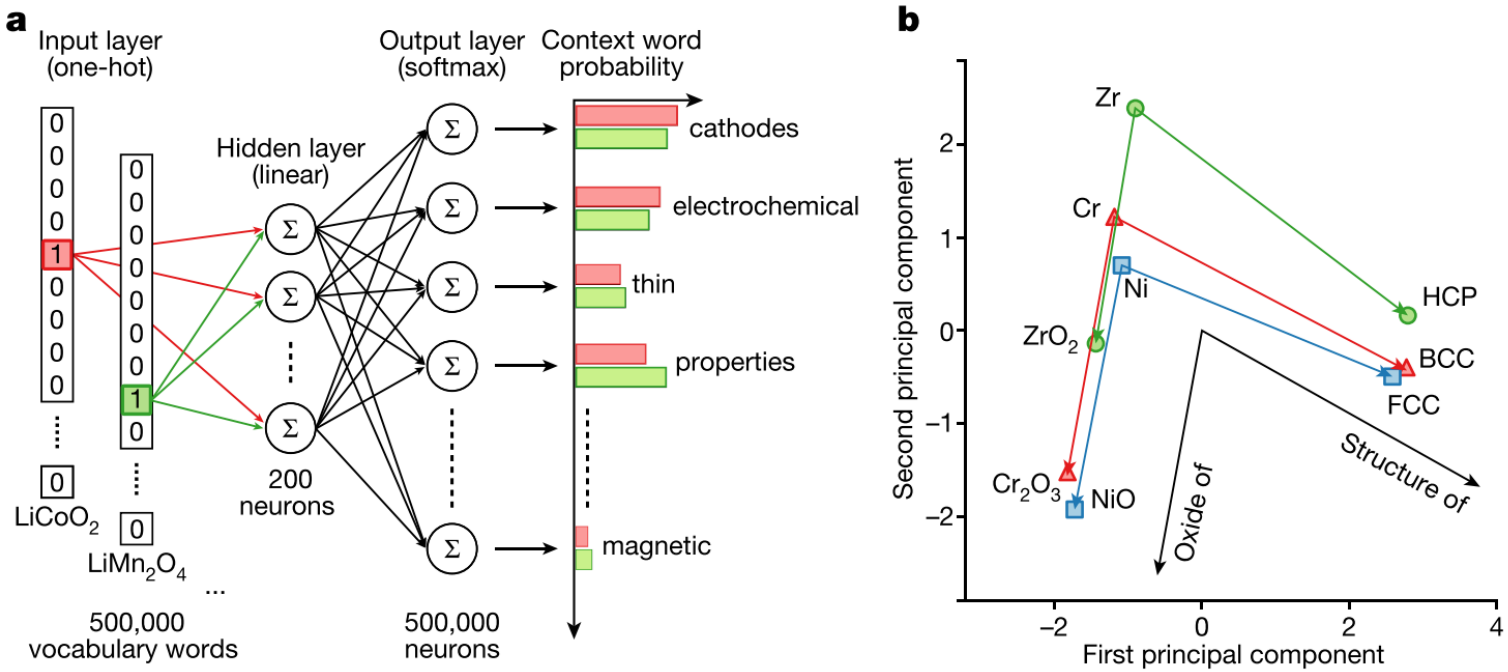


Fig. 1 | Word2vec skip-gram and analogies. a, Target words '$LiCoO_2$' and '$LiMn_2O_4$' are represented as vectors with ones at their corresponding vocabulary indices (for example, 5 and 8 in the schematic) and zeros everywhere else (one-hot encoding). These one-hot encoded vectors are used as inputs for a neural network with a single linear hidden layer (for example, 200 neurons), which is trained to predict all words mentioned within a certain distance (context words) from the given target word. For similar battery cathode materials such as $LiCoO_2$ and $LiMn_2O_4$, the context words that occur in the text are mostly the same (for example, 'cathodes', 'electrochemical', and so on), which leads to similar hidden layer weights after the training is complete. These hidden layer weights are the actual word embeddings. The softmax function is used at the output layer to normalize the probabilities. **b**, Word embeddings for Zr, Cr and Ni, their principal oxides and crystal symmetries (at standard conditions) projected onto two dimensions using principal component analysis and represented as points in space. The relative positioning of the words encodes materials science relationships, such that there exist consistent vector operations between words that represent concepts such as 'oxide of' and 'structure of'.

Atom2Vec creates a vector based on known chemical ratios

Learning atoms for materials discovery

Quan Zhou[a], Peizhe Tang[a], Shenxiu Liu[a], Jinbo Pan[b], Qimin Yan[b], and Shou-Cheng Zhang[a,c,1]

[a]Department of Physics, Stanford University, Stanford, CA 94305-4045; [b]Department of Physics, Temple University, Philadelphia, PA 19122; and [c]Stanford Institute for Materials and Energy Sciences, SLAC National Accelerator Laboratory, Menlo Park, CA 94025

Contributed by Shou-Cheng Zhang, June 4, 2018 (sent for review February 2, 2018; reviewed by Xi Dai and Stuart P. Parkin)

Exciting advances have been made in artificial intelligence (AI) during recent decades. Among them, applications of machine learning (ML) and deep learning techniques brought human-competitive performances in various tasks of fields, including image recognition, speech recognition, and natural language understanding. Even in Go, the ancient game of profound complexity, the AI player has already beat human world champions convincingly with and without learning from the human. In this work, we show that our unsupervised machines (Atom2Vec) can learn the basic properties of atoms by themselves from the extensive database of known compounds and materials. These learned properties are represented in terms of high-dimensional vectors, and clustering of atoms in vector space classifies them into meaningful groups consistent with human knowledge. We use the atom vectors as basic input units for neural networks and other ML models designed and trained to predict materials properties, which demonstrate significant accuracy.

atomism | machine learning | materials discovery

The past 20 y witnessed the accumulation of an unprecedentedly massive amount of data in materials science via both experimental explorations and numerical simulations (1–5). The huge datasets not only enable but also call for data-based statistical approaches. As a result, a new paradigm has emerged which aims to harness artificial intelligence (AI) and machine-learning (ML) techniques (6–10) to assist materials research and discovery. Several initial attempts have been made along this path (11–16). Most of them learned maps from materials information (input) to materials properties (output) based on known materials samples. The input or feature of materials involves descriptors of constituents: Certain physical or chemical attributes of atoms are taken, depending on the materials property under prediction (11, 14, 17). Despite the success so far, these works heavily rely on researchers' wise selection of relevant descriptors; thus the degree of intelligence is still very limited from a theoretical perspective. And practically, extra computations are usually unavoidable for machines to interpret such atom descriptors which are in the form of abstract human knowledge.

To create a higher level of AI and to overcome the practical limitation, we propose Atom2Vec in this paper, which lets machines learn their own knowledge about atoms from data. Atom2Vec considers only existence of compounds in a materials database, without reference to any specific property of materials. This massive dataset is leveraged for a learning feature in materials science in an unsupervised manner (18–23). Because of the absence of materials property labels, Atom2Vec is naturally prevented from being biased to one certain aspect. As a result, the learned knowledge can yield complete and universal descriptions of atoms in principle, as long as the dataset is sufficiently large and representative. Atom2Vec follows the core idea that properties of an atom can be inferred from the environments it lives in, which is similar to the distributional hypothesis in linguistics (24). In a compound, each atom can be selected as a target type, while the environment refers to all remaining atoms together with their positions relative to the target atom. Intuitively, similar atoms tend to appear in similar environments, which allows our Atom2Vec to extract knowledge from the associations between atoms and environments and then represent it in a vector form as discussed in the following.

Atom2Vec Workflow

We begin to illustrate the full workflow of Atom2Vec as shown in Fig. 1. To capture relations between atoms and environments, atom–environment pairs are generated for each compound in a materials dataset as the first step. Before pair generation, a more explicit definition of environment is needed, whereas atoms are represented by chemical symbols conveniently. Although a complete environment should involve both chemical composition and crystal structure as mentioned before, we take into account only the former here as a proof of concept. Under this simplification, environment covers two aspects: the count of the target atom in the compound and the counts of different atoms in the remains. As an example, let us consider the compound Bi_2Se_3 from the mini-dataset of only seven samples given in Fig. 1. Two atom–environment pairs are generated from Bi_2Se_3: For atom Bi, the environment is represented as "$(2)Se3$"; for atom Se, the environment is represented as "$(3)Bi2$." Specifically, for the first pair, "(2)" in the environment $(2)Se3$ means there are two target atoms (Bi here for the compound), while "$Se3$" indicates that three Se atoms exist in the environment. Following the notation, we collect all atom–environment pairs from the dataset and then record them in an atom–environment matrix $\boldsymbol{X}$, where its entry X_{ij} gives the count of pairs with the ith atom and the jth environment. Such a matrix for the mini-dataset is also given in Fig. 1 for illustration purposes. Clearly, each row vector gives counts with different environments for one atom, and each column vector yields counts with different atoms for one environment. According to the previously mentioned intuition, two atoms behave similarly if their corresponding row vectors are close to each other in the vector space.

Although revealing similarity to some extent, descriptions of atoms in terms of row vectors of the atom–environment matrix are still very primitive and inefficient, since the vectors can be

Significance

Motivated by the recent achievements of artificial intelligence (AI) in linguistics, we design AI to learn properties of atoms from materials data on its own. Our work realizes knowledge representation of atoms via computers and could serve as a foundational step toward materials discovery and design fully based on machine learning.

Author contributions: Q.Z., P.T., S.L., Q.Y., and S.-C.Z. designed research; Q.Z., P.T., and S.L. performed research; Q.Z., P.T., and S.L. contributed new reagents/analytic tools; Q.Z., P.T., S.L., J.P., and Q.Y. analyzed data; and Q.Z. and P.T. wrote the paper.

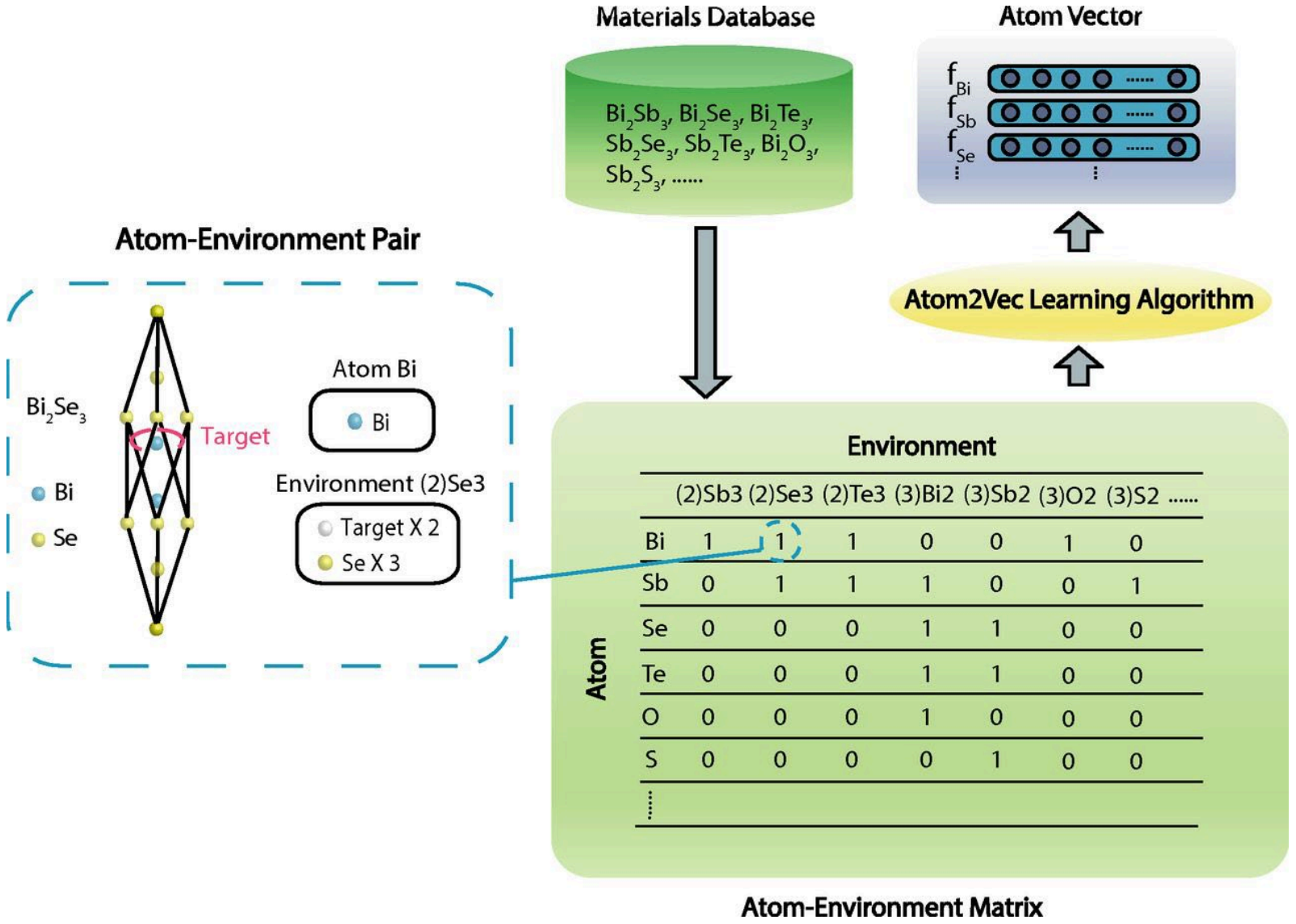
Reviewers: X.D., Institute of Physics, Chinese Academy of Sciences; and S.P.P., Max Planck Institute of Microstructure Physics in Halle, Martin-Luther-University Halle-Wittenberg.

The authors declare no conflict of interest.

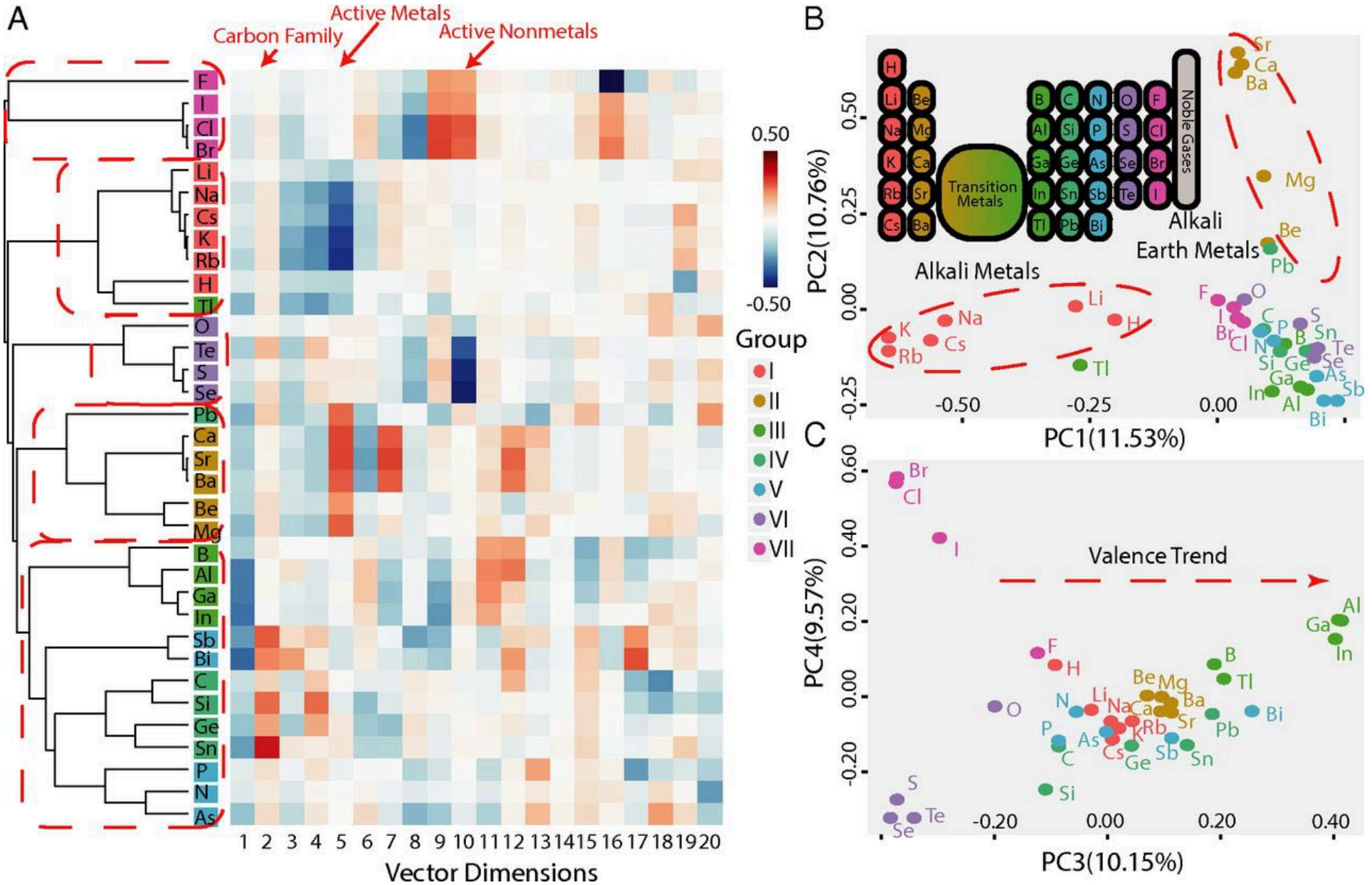
[1]To whom correspondence should be addressed. Email: sczhang@stanford.edu.

This article contains supporting information online at www.pnas.org/lookup/suppl/doi:10.1073/pnas.1801181115/-/DCSupplemental.

Published online June 26, 2018.



Atom2Vec creates a vector based on known chemical ratios



ElemNet is a simple one-hot encoding feature paired with deep learning

www.nature.com/scientificreports

SCIENTIFIC REPORTS

OPEN

ElemNet: Deep Learning the Chemistry of Materials From Only Elemental Composition

Received: 1 August 2018
Accepted: 6 November 2018
Published online: 04 December 2018

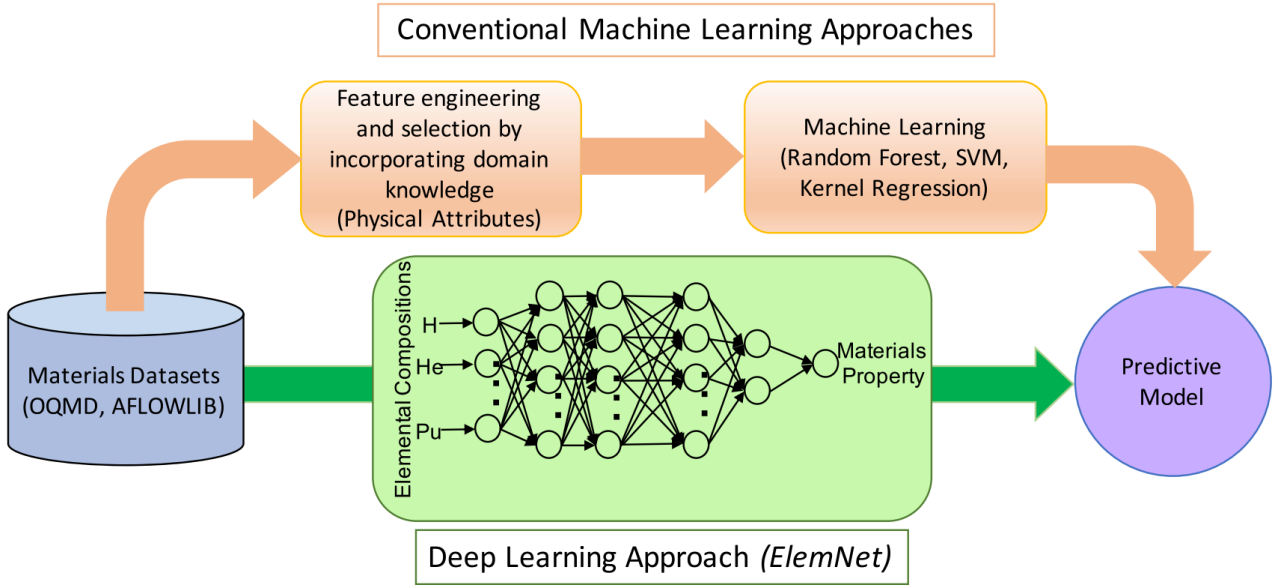
Dipendra Jha[1], Logan Ward[2], Arindam Paul[1], Wei-keng Liao[1], Alok Choudhary[1], Chris Wolverton[3] & Ankit Agrawal[1]

Conventional machine learning approaches for predicting material properties from elemental compositions have emphasized the importance of leveraging domain knowledge when designing model inputs. Here, we demonstrate that by using a deep learning approach, we can bypass such manual feature engineering requiring domain knowledge and achieve much better results, even with only a few thousand training samples. We present the design and implementation of a deep neural network model referred to as *ElemNet*; it automatically captures the physical and chemical interactions and similarities between different elements using artificial intelligence which allows it to predict the materials properties with better accuracy and speed. The speed and best-in-class accuracy of *ElemNet* enable us to perform a fast and robust screening for new material candidates in a huge combinatorial space; where we predict hundreds of thousands of chemical systems that could contain yet-undiscovered compounds.

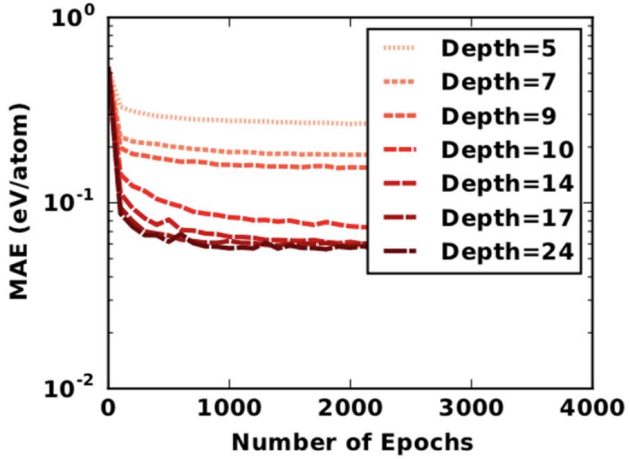
Materials scientists, condensed matter physicists and solid-state chemists rely on data generated by experiments and simulation-based models to discover new materials and understand their characteristics. For the major part of the history of materials science, experimental observations have been the primary means to know the various chemical and physical properties of materials[1–6]. Nevertheless, experimentation of all possible combinations of material composition and crystal structures is not feasible as that would be very expensive and time-consuming, and the composition space is practically infinite. Computational methods, such as Density Functional Theory (DFT)[7], offer a less expensive means to predict many material properties and processes on the atomic level[8]. DFT calculations have offered opportunities for large-scale data collection such as the Open Quantum Materials Database (OQMD)[9,10], the Automatic Flow of Materials Discovery Library (AFLOWLIB)[11], the Materials Project[12], and the Novel Materials Discovery (NoMaD)[13]; they contain DFT computed properties of ~10^4–10^6 of experimentally-observed and hypothetical materials. In the past few decades, such materials datasets have led to the new data-driven paradigm of materials informatics[14–19]. The availability of such large data resources has spurred the interest of researchers in applying advanced data-driven based machine learning (ML) techniques for accelerated discovery and design of new materials with select engineering properties[19–39].

Conventionally, constructing an effective ML model requires first developing a suitable representation for the input data as shown in Fig. 1. As has been discussed in several recent works, the best representations are those that encode knowledge about the physics of the underlying problem. To that end, there have been many distinct approaches for encoding information regarding the composition[23,32] or crystal structure[34,37,40,41] of a material. For instance, Ward *et al.* developed a set of attributes based on the composition of a material that can be useful for problems including predicting formation enthalpies of crystalline materials and glass-forming ability of metal alloys[32]. Ghiringhelli *et al.*[42] analyzed the tendency for materials to form different crystal structures using thousands of descriptors. Developing ML models based on intuitive representations is evidently successful given the large number and growing rate of ML models constructed over the past several years using this approach[18,19,43]. However, the prediction accuracy for these problems is limited by our ability to feature engineer the materials representation to incorporate all the domain knowledge required to make correct predictions. Given that one of the major use cases of ML is for problems where the physics driving behavior is yet to be understood[19], this limit

[1]Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, USA. [2]Computation Institute, University of Chicago, Chicago, USA. [3]Department of Materials Science and Engineering, Northwestern University, Evanston, USA. Correspondence and requests for materials should be addressed to A.A. (email: ankitag@eecs.northwestern.edu)



Layer Types	No. of units	Activation	Layer Positions
Fully-connected Layer	1024	ReLU	First to 4th
Drop-out (0.8)	1024		After 4th
Fully-connected Layer	512	ReLU	5th to 7th
Drop-out (0.9)	512		After 7th
Fully-connected Layer	256	ReLU	8th to 10th
Drop-out (0.7)	256		After 10th
Fully-connected Layer	128	ReLU	11th to 13th
Drop-out (0.8)	128		After 13th
Fully-connected Layer	64	ReLU	14th to 15th
Fully-connected Layer	32	ReLU	16th
Fully-connected Layer	1	Linear	17th



Are CBFV features based on domain knowledge better?

Integrating Materials and Manufacturing Innovation
https://doi.org/10.1007/s40192-020-00179-z

TECHNICAL ARTICLE

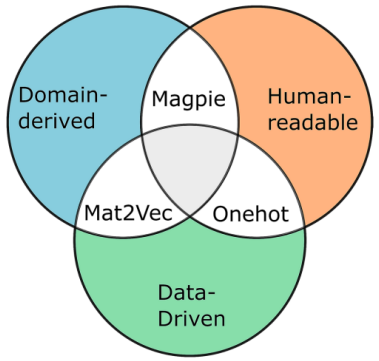
Is Domain Knowledge Necessary for Machine Learning Materials Properties?

Ryan J. Murdock[1] · Steven K. Kauwe[1] · Anthony Yu-Tung Wang[2] · Taylor D. Sparks[1]

Received: 4 June 2020 / Accepted: 9 July 2020

Abstract
New featurization schemes for describing materials as composition vectors in order to predict their properties using machine learning are common in the field of Materials Informatics. However, little is known about the comparative efficacy of these methods. This work sets out to make clear which featurization methods should be used across various circumstances. Our findings include, surprisingly, that simple fractional and random-noise representations of elements can be as effective as traditional and new descriptors when using large amounts of data. However, in the absence of large datasets or for data that is not fully representative, we show that the integration of domain knowledge offers advantages in predictive ability.

Graphical abstract



Keywords Materials informatics · Machine learning · Featurization · Descriptors · Neural networks

✉ Taylor D. Sparks
sparks@eng.utah.edu

[1] Materials Science and Engineering Department, University of Utah, Salt Lake City, UT 84109, USA

[2] Technische Universität Berlin, Fachgebiet Keramische Werkstoffe/Chair of Advanced Ceramic Materials, 10623 Berlin, Germany

Introduction

In Materials Informatics (MI), composition-based machine learning (ML) entails the creation of a composition-based feature vector (CBFV) that represents materials based on expertly curated element properties. Traditionally, descriptive statistics (average, range, sum, and variance) regarding

Published online: 27 August 2020

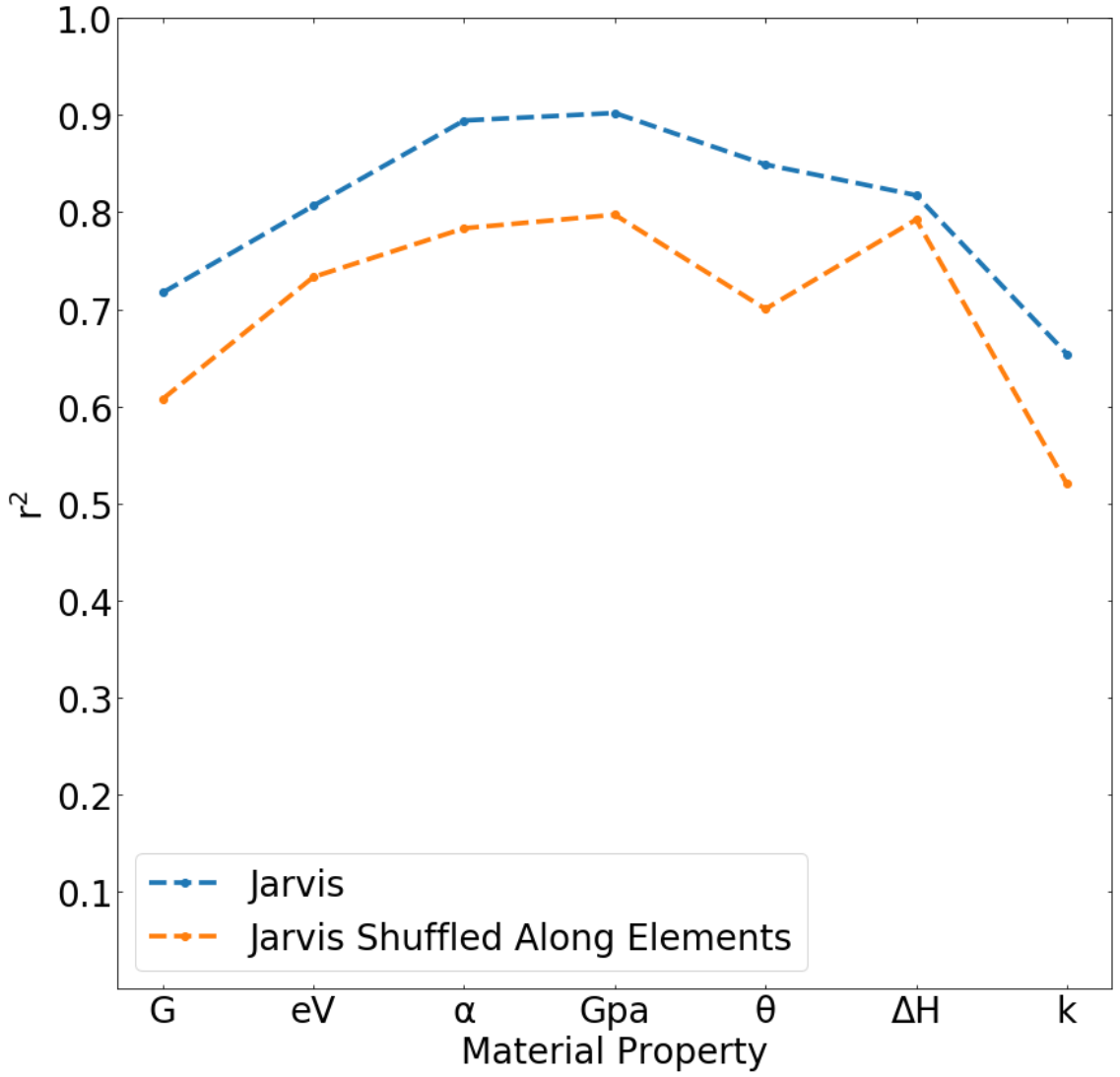
Consider JARVIS featurization code from NIST

1	element	Number	MendeleevNumber	AtomicWeight	MeltingT	Column	Row	CovalentRadius	Electronegativity
11	Ne	10	99	20.1791	24.56	18	2	58	1.63
12	Na	11	2	22.98976928	370.87	1	3	166	0.93
13	Mg	12	68	24.305	923	2	3	141	1.31
14	Al	13	73	26.9815386	933.47	13	3	121	1.61
15	Si	14	78	28.0855	1687	14	3	111	1.9
16	P	15	83	30.973762	317.3	15	3	107	2.19

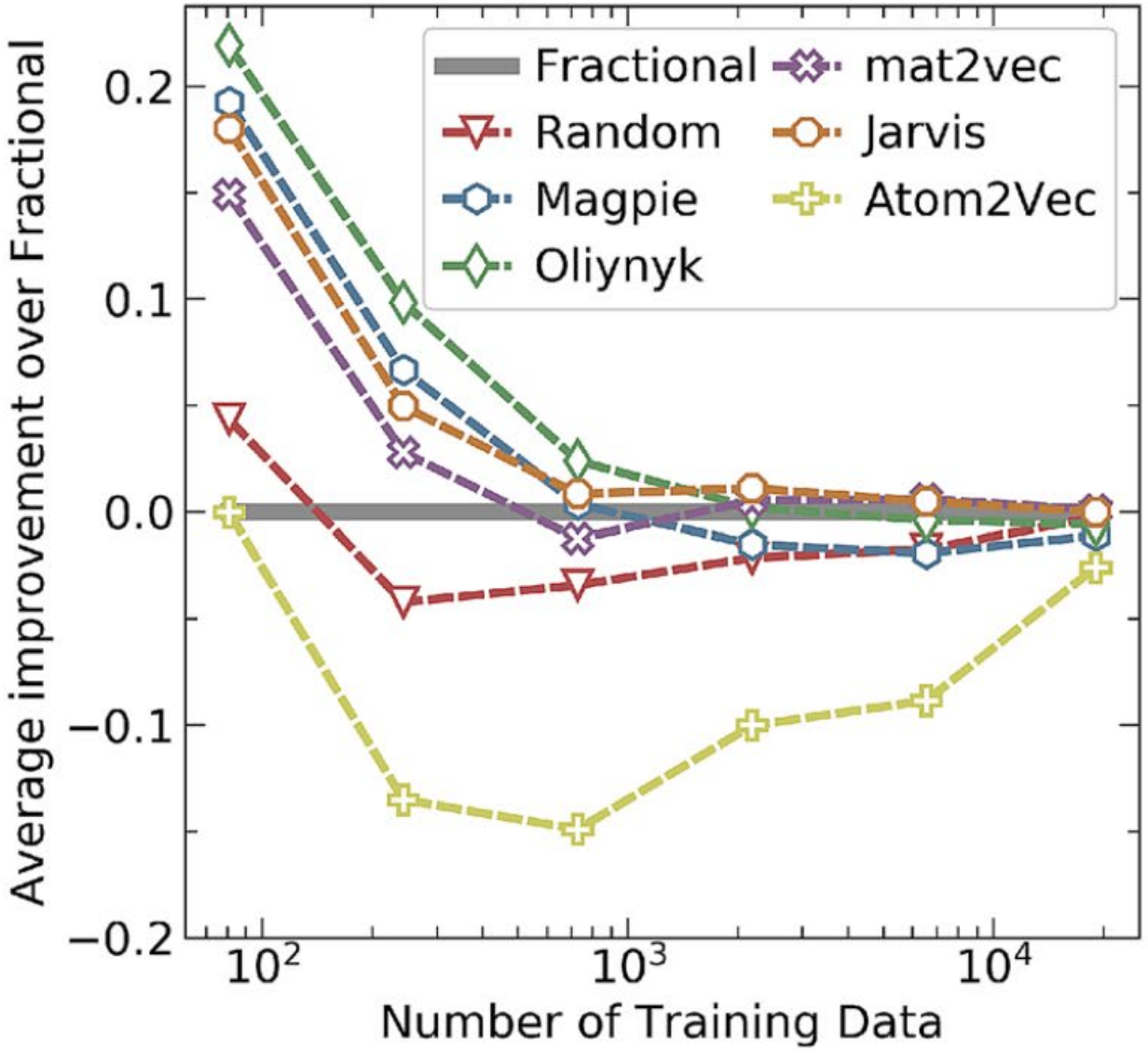
1	element	Number	MendeleevNumber	AtomicWeight	MeltingT	Column	Row	CovalentRadius	Electronegativity
11	Ne	10	99	20.1791	24.56	18	2	58	1.63
12	Na	11	2	22.98976928	370.87	1	3	166	0.93
13	Mg	12	68	24.305	923	2	3	141	1.31
14	Al	13	73	26.9815386	933.47	13	3	121	1.61
15	Si	14	78	28.0855	1687	14	3	111	1.9
16	P	15	83	30.973762	317.3	15	3	107	2.19

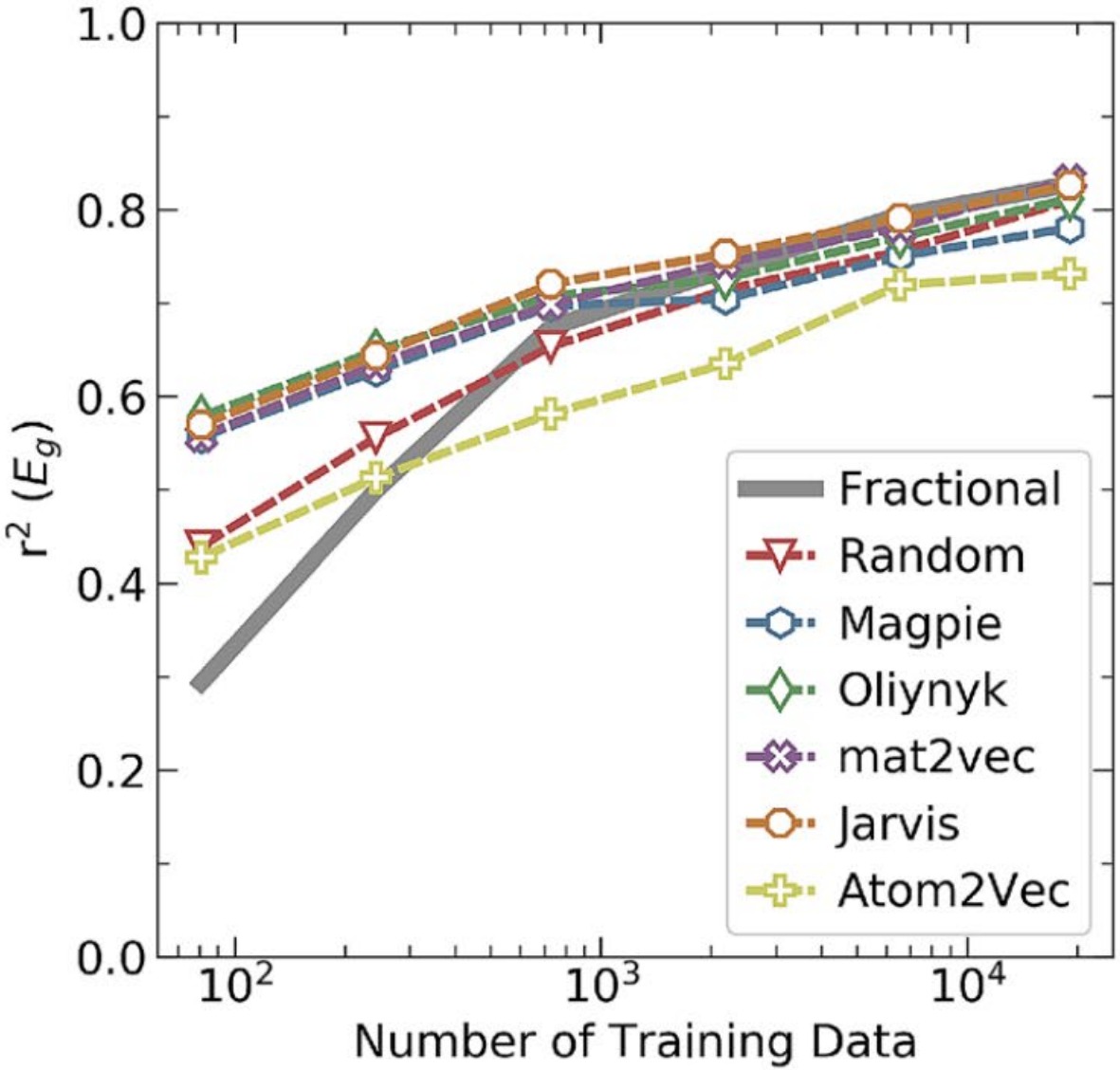
Will JARVIS break if we swap some rows?

JARVIS with random shuffling does worse (sanity check!)



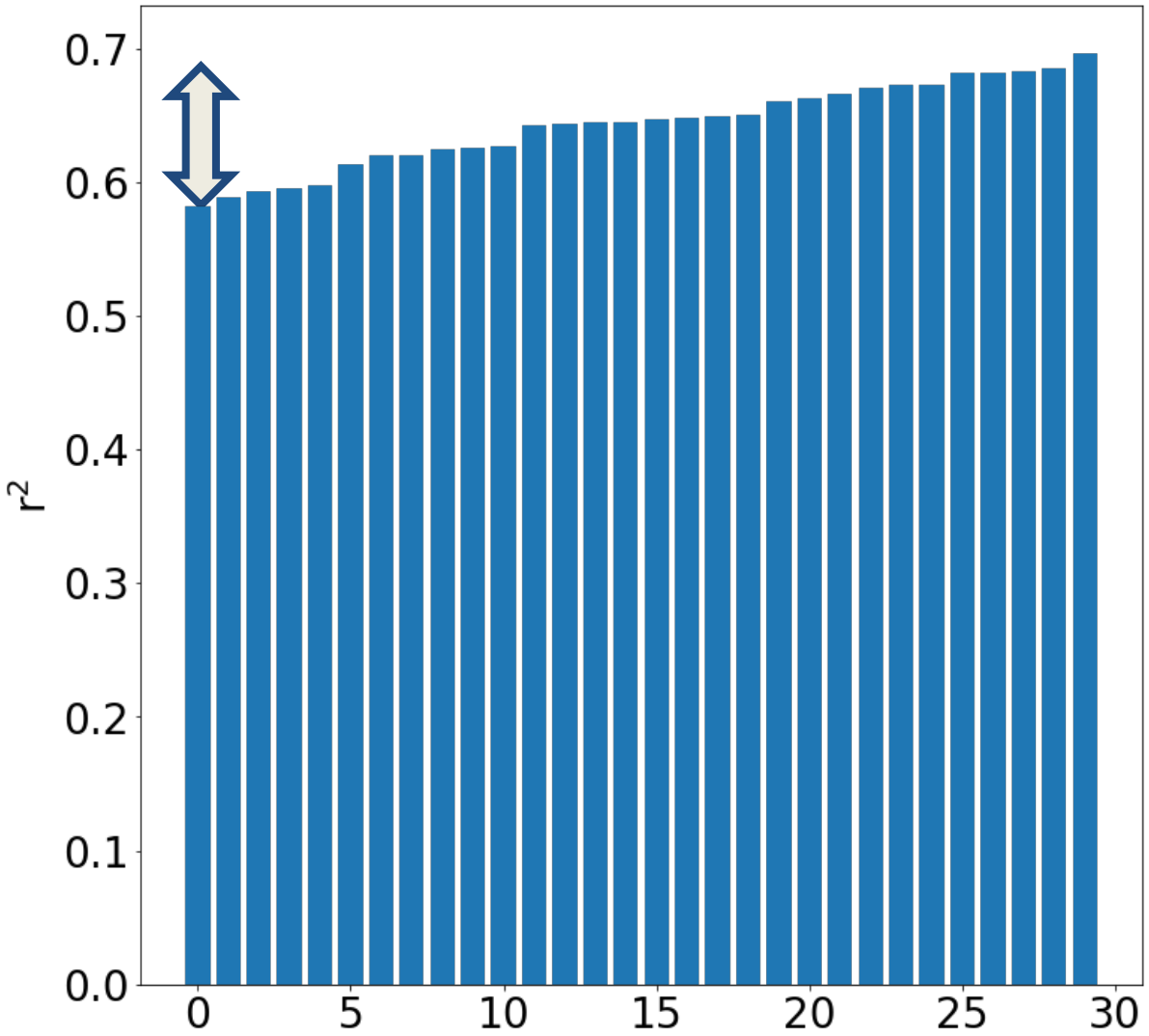
Domain knowledge helps a bit, but not much and primarily in low data limit



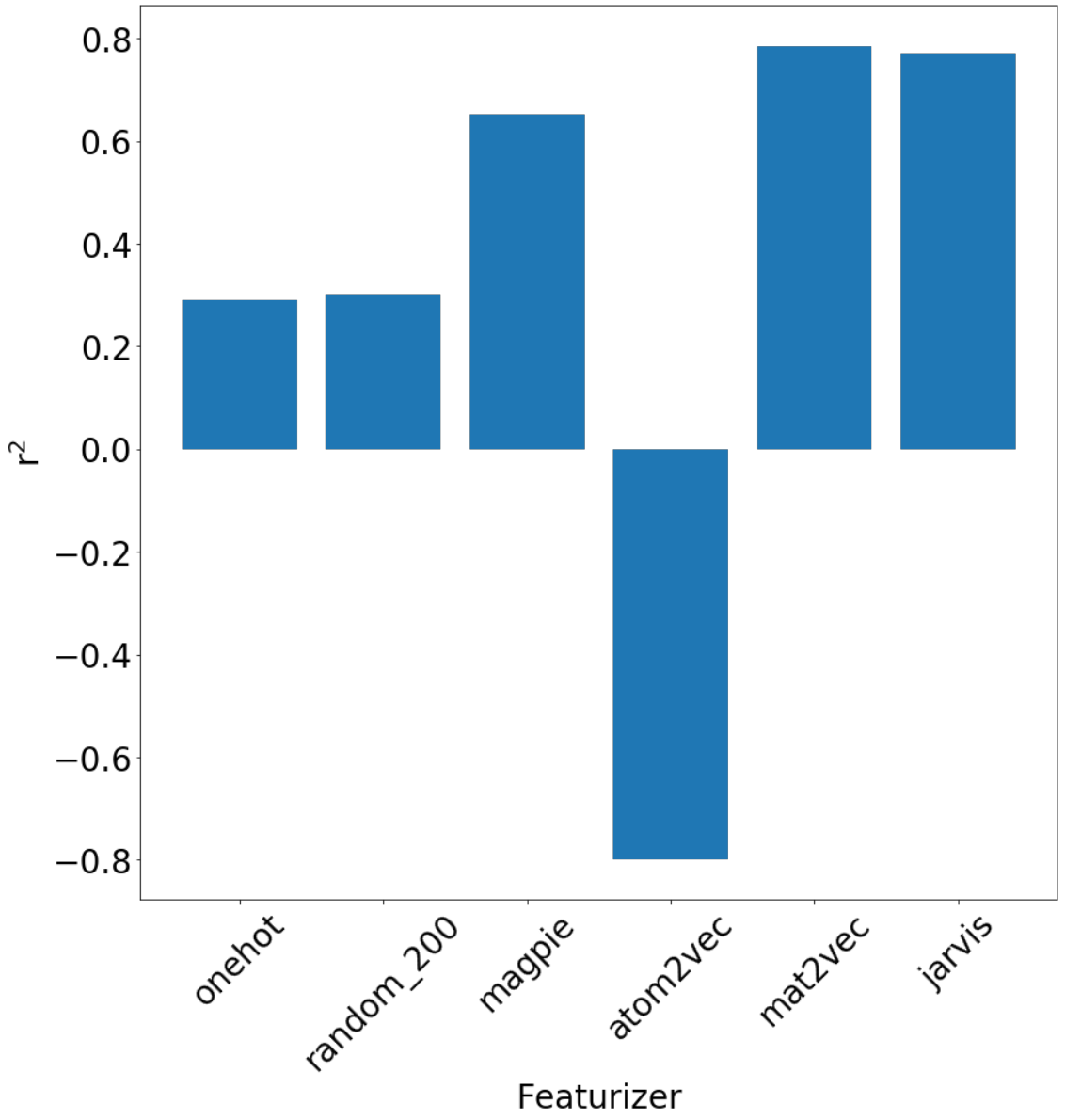


Most feature comparisons are not very meaningful!

Over 15% variation just from selecting from your randomly splitting cross-validation set!



Feature engineering may help us extrapolate



Our CBFV package makes it very easy to compare many different CBFV techniques in python

CBFV 1.0.1

```
pip install CBFV
```

Latest version

Released: about 17 hours ago

Tool for quickly creating a composition-based feature vector

Navigation

- Project description
- Release history
- Download files

Project links

- Homepage

Statistics

GitHub statistics:

- Stars: 3
- Forks: 1
- Open issues/PRs: 2

Project description

https://pypi.org/project/CBFV/

CBFV Package

Tool to quickly create a composition-based feature vectors from materials datafiles.

Installation

The source code is currently hosted on GitHub at: https://github.com/kaaiian/CBFV

Binary installers for the latest released version are available at the Python Package Index (PyPI)

```
# PyPI
pip install CBFV
```

Making the composition-based feature vector

The CBFV package assumes your data is stored in a pandas dataframe of the following structure:

Structure-based feature vector