Preserving Composition and Crystal Structures of Chemical Compounds in Atomic Embedding

Yifan Ding Daheng Wang Tim Weninger Meng Jiang
Department of Computer Science & Engineering
University of Notre Dame
Notre Dame, IN, USA
{yding4,dwang8,tweninge,mjiang2}@nd.edu

***Abstract*—Representation learning is popular for its power of learning latent feature vectors (i.e., embeddings) to represent data units from a complex type of data (e.g., languages, networks, behaviors). The embeddings preserve specific structure and thus improve the performance of predictive models. In this work, we develop a new representation learning method in the *chemistry* domain. Given a large set of compounds of inorganic crystals, the method learns the embeddings of atoms so that the predictive models can place them into the periodic table correctly. Our method preserves not only the compounds' *compositions* but also their *structures* such as crystal system, point group, and space group. Experiments demonstrate the effectiveness of the proposed method, compared to the state-of-the-art method (in PNAS 2018). One interesting result is that given 20 atoms with known positions in the periodic table, our method can achieve an accuracy of 0.70, while the baseline makes only 0.54, on filling the remaining 14 hidden atoms into the table. This shows that the atomic embeddings we generated preserve useful information and can be extended for scientific exploration.**

I. Introduction

Since an unprecedentedly big amount of data in chemistry and materials science become available, the potential use of data science in the fields have been brought into attention [1], [2], [3]. Statistical learning methods are being used for battery materials discovery [4], solid catalysts [5], and computational material design [6]. The fundamental idea is that the properties of atoms/elements could be represented and learned from materials databases.

A recent work published in the Proceedings of the National Academy of Sciences of the United States of America (PNAS) [7] represented the composition information of chemical compounds as an atom-environment matrix and used Singular Value Decomposition (SVD) [8] to find the feature vector of atoms (as the left singular vectors). Results showed that the feature vectors are useful on predicting the formation energy of elpasolites [9]. Meanwhile, in the community of data science and machine learning, representation learning has been getting more and more attention. Tons of research works demonstrate the effectiveness of being used as a unsupervised feature extraction tool. Basically, it learns low-dimensional vectors of the data units (called *embedding*) from different types of data sets, such as text embeddings [10], [11], network embeddings [12], [13], [14], [15], [16], knowledge graph embeddings [17], [18], [19], spatiotemporal embeddings [20], and behavior embeddings [21]. One of the advantages of the representation learning algorithms is that it can preserve multiple kinds of information (if specified) from the data in the embedding vectors, which could be applied to the data-driven atomic feature learning task.

In this work, we investigate the important role of structure information in describing the properties of the chemical compounds. The top part of Figure 1 shows two kinds of information of chemical compounds, composition (on the left) and crystal structures (on the right). We find that different chemical compounds may have the same composition but totally different crystal structures. For example, the *Wurtzite* and *Sphalerite* are of the same composition ("$\boldsymbol{ZnS}$") but different crystal systems: *Wurtzite* is *hexagonal* and *Sphalerite* is *cubic*. This is actually common in crystal chemistry. Moreover, there is a taxonomy of crystal structures which has three levels: crystal systems, point groups, and space groups [9]. The number of systems/groups of the levels are 7, 32, 230, respectively. We focus on solving two problems: (1) preserving both the composition and crystal structure information of chemical compounds in the embeddings of atoms, (2) investigating which granularity of the crystal structure generates the most effective embeddings.

As shown in Figure 1, we employ the network embedding algorithm [16] to learn atomic embeddings by preserving the composition and crystal structure information. The idea is to structure chemical compounds into a bipartite graph between atom nodes and environment nodes. The environment includes two components. One is the composites in the compound. For example, *Perovskite* is known as $\boldsymbol{CaTiO_3}$: if the atom is $\boldsymbol{Ca}$, the composites environment is $\mathbf{1}\ \boldsymbol{TiO_3}$. "$\mathbf{1}$" is for the count of atom $\boldsymbol{Ca}$ in the compound and "$\boldsymbol{TiO_3}$ is for the other atoms and their counts. Similarly, if the atom is $\boldsymbol{Ti}$, the composites environment is $\mathbf{1}\ \boldsymbol{CaO_3}$; if the atom is $\boldsymbol{O}$, the composites environment is $\mathbf{3}\ \boldsymbol{CaTi}$. The other environment is the crystal structures, as shown as a hierarchy at the top right of the figure. *Perovskite* is located at the leaf node of the path "orthorhombic" (crystal system) – "mmm" (point group) – "Pnma" (space group).

In the present work we describe a **composition-structure convolution** to represent the environment nodes. Basically, an environment is recognized as a pair of the composition environment and crystal structure environment such as "[$\mathbf{1}\ \boldsymbol{TiO_3}$,

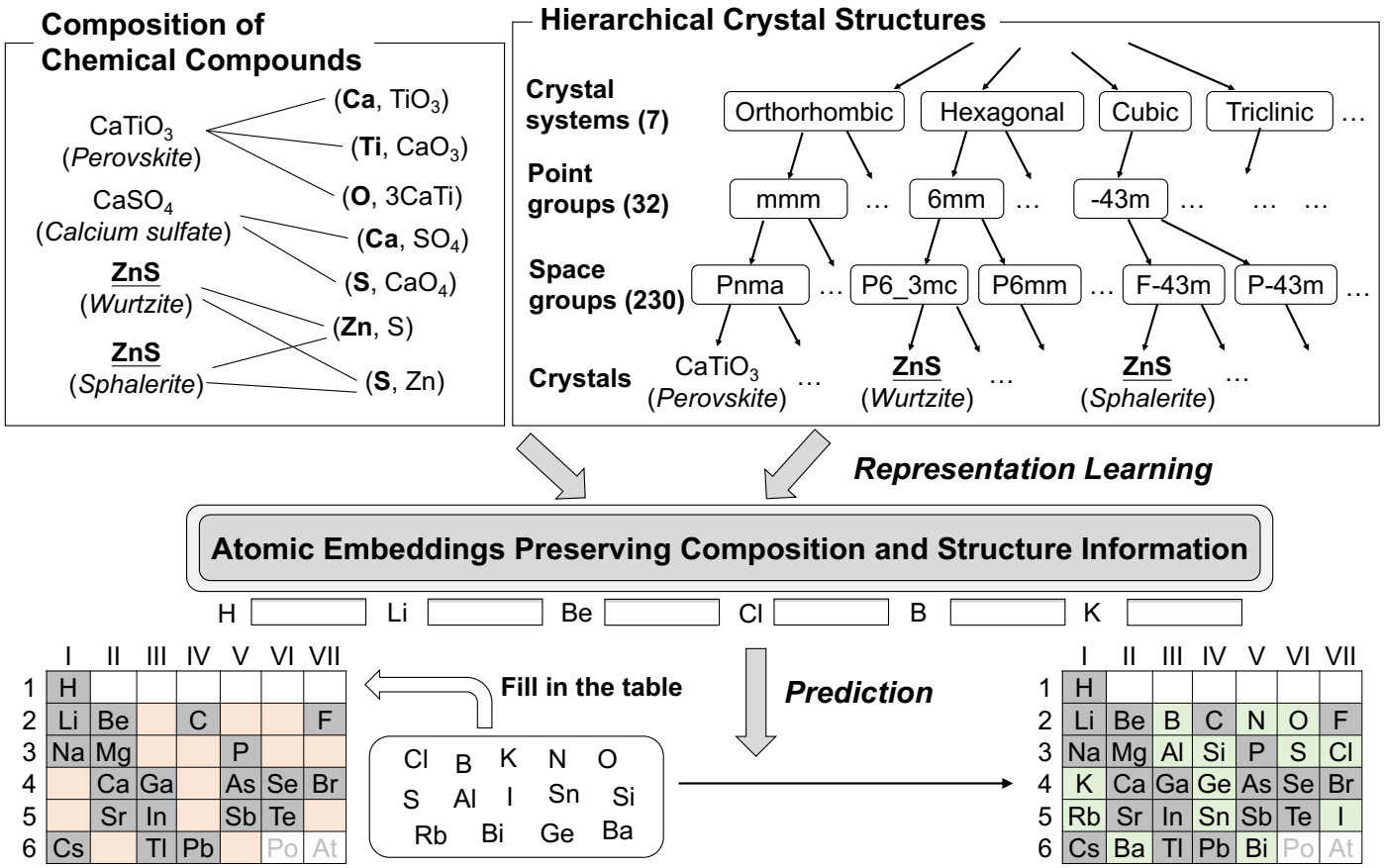


Fig. 1. Representation learning in Chemistry: We proposed a new representation learning method that can learn embedding vectors of atoms from both (1) composition of chemical compounds and (2) hierarchical crystal structures and deployed the method for periodic table prediction (given the correct positions of 20 atoms, place the rest 14 atoms in the slots).

"Pnma"] and "[$\mathcal{S}$, "P6_3mc"] (when we choose the level of space group). This significantly increases the number of unique environments from 83,743 to 132,517 (as given in Figure 3 later) and the atomic embeddings are more effective than those learned from the composition information only.

We evaluate the atomic embeddings on two tasks. The first task is to predict the atom's position in the periodic table [22], [23]. It can also be considered as a table filling task. As shown at the bottom of Figure 1, suppose we have the information of 20 atoms' positions in the table. The task is to fill the 14 remaining atoms to correct positions. This is a non-trivial task. It requires the numerical representations of the atoms to embed their nature properties from chemical compound data because the ground-truth table actually placed the atoms based on their natures (i.e., atomic particle structures) [24], [25]. The second task is standard and has been adopted by the PNAS work [7]. It is to predict the formation energy of elpasolite crystals by feeding the embeddings into a two-layer neural network. This is an important task in material discovery.

Experimental results show that on both tasks, the atomic embeddings that preserves both composition and crystal structure information performs better than those that preserve one kind only. Given 20 atoms in the periodic table, our embeddings can achieve an accuracy of 0.70 on average, while the baseline makes 0.54, on filling the 14 hidden atoms into the table. As the example given in Figure 1, our embeddings hit 10 among 14, while the baseline makes only 5. We found that the space group (bottom level) performs the best in the first task, while the crystal system (top level) performs the best in the second task.

Our contributions can be summarized as follows:

- We introduce a taxonomy of crystal structures for the representation of chemical compounds. The taxonomy has three levels: crystal systems, point groups, and space groups. They play essential roles: different chemical compounds may have the same composition but different structures.
- We propose composition-structure convolution to represent the associations between atoms and environments in the chemical compounds as a bipartite network. We employ state-of-the-art network embedding algorithms to learn effective atomic embeddings.
- Experimental results demonstrate the effectiveness of the atomic embedding. We evaluate it on a new task and a standard task. The new task is to fill the periodic table given partial placement of the atoms.

The remainder of this paper is organized as follows. Section II reviews the related works. Section III defines the problem. The proposed approach is given in Section IV. Section V shows and discusses the experimental results, and Section VI concludes the study.

II. Related Work

In this section, we review three related topics: representation Learning, atomic embedding and periodic table.

A. Representation Learning

Representation learning methods have been widely used for unsupervised feature extraction. It is to learn low-dimensional embedding vectors from complicated types of data. The embeddings preserve specific structures inside the data to improve the performance of predictive models. The WORD2VEC [10] and GLOVE [11] methods learn word embeddings (and later extended to phrase, sentence, paragraph embeddings, and document embeddings) from natural language texts in an unsupervised way: one is to train the vectors for predicting the missing word in a certain context; the other is to recover the co-occurrences of words in a large amount of text data. Network embedding methods such as DEEPWALK [12], LINE [13], HEBE [14], and many others [16], [26], [27], [28], [29], [30], [15], [31] are able to extract effective features for multiple network-oriented tasks such as node classification, clustering, or outlier detection. To facilitate question-answering and recommender systems, representation learning for knowledge graphs is gaining more and more attention [17]. Effective answers and recommendations were generated through embedding learning of entities and relations [18], [19]. Some other kinds of embeddings are also popular. For example, learning spatiotemporal factors in dynamic data can improve the applications of smart city [20]. Learning representations of resources, plans, and goals in human behaviors can provide reliable support to the decision-making process [21].

B. Atomic Embedding

A general idea of predicting chemicals' properties is to decompose them into atoms' inherent property (descriptor) and interactions between them. However, some inherent properties of atoms are hard or even impossible to gain. Atomic embedding, which can either be modified directly from atomic properties or generated from other observed data, is potentially a convenient and powerful atomic descriptor [32]. Researchers have applied atomic embedding in potential energy surface [33], classification in superheavy elements [34], and drug discovery [35].

C. Periodic Table

In 1869, the well-known Russian chemist Dmitri Mendeleev published a periodic table arranging the elements in order of relative atomic mass. Mendeleev came up the physical and chemical properties of elements are related to their atomic mass in a "periodic" way known as periodic trends by modern chemistry. Another contribution was to predict the undiscovered elements in the correct location in his periodic table.

The modern periodic table arranges elements by the number of electrons [22], [23]. In this paper, we focus on the 34 main group elements. Electron arrangement of each element is represented by the position in the table: the row number (1-6) represent the total layers of electrons while the column (I-VII) represent the number of electrons in the outermost layer.

III. Problem Definition

In this section, we formally define the composition and crystal structure information of chemical compounds in a mathematical way. Then, we define the problem of atomic embedding learning with the requirement of preserving the two kinds of information. Finally, we define the periodic table filling task.

Suppose $\boldsymbol{\mathcal{X}}$ is the set of elements/atoms. We denote the chemical compound by $\boldsymbol{C}$. For example, an atom could be $\boldsymbol{Ca}$, $\boldsymbol{Ti}$, or $\boldsymbol{O} \in \boldsymbol{\mathcal{X}}$. The chemical compound $\boldsymbol{C}$ could be $\boldsymbol{CaTiO_3}$. We define a function as the count of an atom $\boldsymbol{X} \in \boldsymbol{\mathcal{X}}$ in the chemical compound $\boldsymbol{C}$:

$$n_{\boldsymbol{C}}(\boldsymbol{X}) : \boldsymbol{\mathcal{X}} \to \mathbb{N} = \{0, 1, 2, \dots\}. \tag{1}$$

For example, we have

$$n_{\boldsymbol{CaTiO_3}}(\boldsymbol{Ca}) = 1, \tag{2}$$
$$n_{\boldsymbol{CaTiO_3}}(\boldsymbol{Ti}) = 1, \tag{3}$$
$$n_{\boldsymbol{CaTiO_3}}(\boldsymbol{O}) = 3. \tag{4}$$

We denote the set of atoms in the chemical compound $\boldsymbol{C}$ as

$$A(\boldsymbol{C}) = \{\boldsymbol{X} \in \boldsymbol{\mathcal{X}} \mid n_{\boldsymbol{C}}(\boldsymbol{X}) > 0\}. \tag{5}$$

The composition information of an atom $\boldsymbol{X}$ in the context of chemical compound $\boldsymbol{C}$ is:

$$Compo(\boldsymbol{X}, \boldsymbol{C}) = [n_{\boldsymbol{C}}(\boldsymbol{X}), \{\boldsymbol{X}' : n_{\boldsymbol{C}}(\boldsymbol{X}')\}_{\boldsymbol{X}' \in A(\boldsymbol{C})}^{\boldsymbol{X}' \neq \boldsymbol{X}}]. \tag{6}$$

For example, when $\boldsymbol{C}$ is $\boldsymbol{CaTiO_3}$ and $\boldsymbol{X}$ is Ti, we have the composition information of atom Ti in $\boldsymbol{CaTiO_3}$ below:

$$[1, \{\boldsymbol{Ca} : 1, \boldsymbol{O} : 3\}],$$

which can be written as

"$\boldsymbol{1CaO_3}$."

Note that the other atoms (denoted by $\boldsymbol{X'}$) were in alphabetical order. We denote the set of all possible composites as:

$$\mathcal{C} = \{Compo(\boldsymbol{X}, \boldsymbol{C}), \forall \boldsymbol{X}, \forall \boldsymbol{C}\}. \tag{7}$$

The crystal structure information of chemical compound $\boldsymbol{C}$ at level l is denoted by

$$Struct(l, \boldsymbol{C}) \in \mathcal{S}_l, \tag{8}$$

where the level of crystal structures $l \in \{$"$crystal\ systems$", "$point\ group$", "$space\ group$"$\}$. For example, we have the values for each level as below:

$$\mathcal{S}_{\text{"}crystal\ systems\text{"}} = \{\text{"}hexagonal\text{"}, \text{"}cubic\text{"}, \dots\}, \tag{9}$$
$$\mathcal{S}_{\text{"}point\ group\text{"}} = \{\text{"}mmm\text{"}, \text{"}6mm\text{"}, \text{"}-43m\text{"}, \dots\},$$
$$\mathcal{S}_{\text{"}space\ group\text{"}} = \{\text{"}Pnma\text{"}, \text{"}P6\_3mc\text{"}, \text{"}P6mm\text{"}, \dots\}.$$

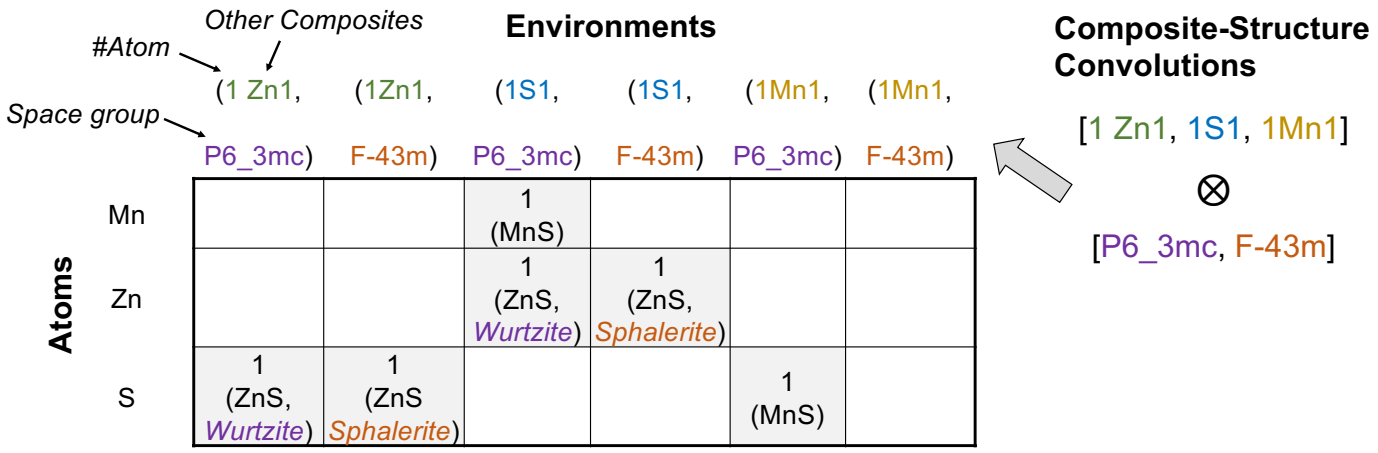


Fig. 2. We propose "composite-structure" convolutions to represent the environments of an atom in the crystal. The composite part includes (1) the count of the atom and (2) other composites (other atoms and their counts).

The size of the levels is

$$|\mathcal{S}_{\text{“crystal systems”}}| = 7, \quad (10)$$
$$|\mathcal{S}_{\text{“point group”}}| = 32, \quad (11)$$
$$|\mathcal{S}_{\text{“space group”}}| = 230. \quad (12)$$

Given a database of chemical compounds, we transform the composition and crystal structure information into *an atom-environment association network* $G = (\boldsymbol{\mathcal{X}}, \mathcal{V}, E_{\boldsymbol{\mathcal{X}}\times\mathcal{V}})$, where $\mathcal{V}$ is the set of all possible environments. Now we can formally define the problem.

Problem 1 (Atomic Embedding Learning): **Given** the composition information $Compo(\boldsymbol{X}, \boldsymbol{C})$ and the crystal structure information $Struct(l, \boldsymbol{C})$, for a chemical compound $\boldsymbol{C}$, a specific atom $\boldsymbol{X}$, and a structure level l, (1) **construct** the atom-environment association network $G = (\boldsymbol{\mathcal{X}}, \mathcal{V}, E_{\boldsymbol{\mathcal{X}}\times\mathcal{V}})$, and (2) **learn** a mapping function:

$$\mathbf{x} = f(\boldsymbol{X}) : \boldsymbol{\mathcal{X}} \rightarrow \mathbb{R}^d \quad (13)$$

where d is the number of dimensions. The function f generates the low-dimensional feature vectors (i.e., embeddings) of atoms $\mathbf{x}$ preserving the composition and crystal structure information.

Problem 2 (Periodic table filling task): For a selected number $n(2-32)$, n randomly selected main group elements are removed from the original table, the task is to utilize the data set(described in Data Description of the Experiment section) to fill the removed elements into the table correctly.

IV. The Proposed Approach

In this section, we first present a novel formulation of the atom's environments (including both composition and crystal structure information) in the chemical compounds and construct an atom-environment network from the data. Then, we briefly introduce the atomic embedding learning method. Finally, we present the table filling method.

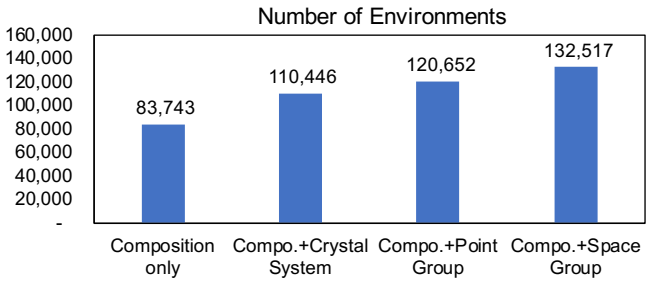


Fig. 3. With structure information being represented via Composite-Structure Convolutions, the number of environments becomes bigger (58% more); and the lower-level crystal structure has more environment.

A. Composite-Structure Convolutions for Atom-Environment Network Construction

Figure 2 presents the idea: Given the composite space $\mathcal{C}$ and the l-level crystal structure space $\mathcal{S}_l$, the environment is defined as the convolution of the composite and the structure, or say, it is a combination of both environment factors. For a specific chemical compound $\boldsymbol{C}$, a specific atom $\boldsymbol{X}$, and a specific structure level l, the environment is written as

$$v(\boldsymbol{X}, \boldsymbol{C}, l) = [Compo(\boldsymbol{X}, \boldsymbol{C}), Struct(l, \boldsymbol{C})]. \quad (14)$$

The set of possible environments only includes the environment nodes that are associated with at least one compound:

$$\mathcal{V} = \{v(\boldsymbol{X}, \boldsymbol{C}, l)\}_{(\boldsymbol{X},\boldsymbol{C},l)} \subset \mathcal{C} \otimes \mathcal{S}_l. \quad (15)$$

As the given example in Figure 2, suppose we have three chemical compounds: $\boldsymbol{MnS}$, $\boldsymbol{ZnS}$ (*Wurtzite*), and $\boldsymbol{ZnS}$ (*Sphalerite*). The atoms are $\boldsymbol{Mn}$, $\boldsymbol{Zn}$, and $\boldsymbol{S}$. Based on Eq.(6), we have three composites:

$\mathbf{1}\boldsymbol{Zn}\mathbf{1}$, $\mathbf{1}\boldsymbol{S}\mathbf{1}$, $\mathbf{1}\boldsymbol{Mn}\mathbf{1}$.

$\boldsymbol{MnS}$ and $\boldsymbol{ZnS}$ (*Wurtzite*) belong to the same space group "P6_3mc". $\boldsymbol{ZnS}$ (*Sphalerite*) belongs to another space group "F-43m". So we have the space group-level structures below:

"P6_3mc" and "F-43m".

The composite-structure convolutions generate $6 = 3 \times 2$ environment candidates as shown as the column names in Figure 2.

We build the network G for each structure level l. The weight of the edge from an atom $\boldsymbol{X} \in \boldsymbol{\mathcal{X}}$ to an environment $v \in \mathcal{V}$ is:

$$a(\boldsymbol{X}, v) = |\{\boldsymbol{C} | v(\boldsymbol{X}, \boldsymbol{C}, l) = v\}|. \tag{16}$$

If $a(\boldsymbol{X}, v) = 0$ for any atom $\boldsymbol{X}$, then we exclude the environment node v from the network. We use the adjacency matrix $\mathbf{A}$ to represent the network. For example, if all the value s on the column ($\mathbf{1}\boldsymbol{Mn}\mathbf{1}$,"F-43m") in the matrix are zero, we delete this column.

Figure 3 shows that if we use the composition only, we have 83,743 unique environment nodes, or say, valid columns; and if we use composition-crystal structure convolutions, for the levels, "crystal systems", "point groups", and "space groups", we have 110,446 (+31.9%), 120,652 (+44.1%), 132,517 (+58.2%) unique environments, respectively. The increase brings a more concrete description of the contexts of atoms in the chemical compounds.

Now the network (as well as the adjacency matrix) has been constructed. The next step is to learn the atomic embedding.

B. Network Representation Learning for Atomic Embeddings

We employ NETMF, a general framework to explicitly factorize the closed-form matrices that the skip-gram powered network embedding algorithms such as DEEPWALK [12], LINE [13], and NODE2VEC [36] aim to implicitly approximate and factorize [16]. In the work of Qiu *et al.*, the authors provided theoretical results concerning these network embedding algorithms. Their experiments demonstrate that NETMF improves the performance relatively by up to 50% over DEEPWALK and LINE. Here are the details of our implementation and deployment of the NETMF algorithm. It has three steps.

Step 1: Given the matrix $\mathbf{A}$, calculate the normalized graph:

$$\hat{\mathbf{A}} = {\mathbf{D}_{row}}^{-1/2} \mathbf{A} {\mathbf{D}_{col}}^{-1/2}, \tag{17}$$

where $\mathbf{D}_{col} = diag(\mathbf{A}^{\mathrm{T}}\mathbf{e})$ is the diagonal matrix with column sum of $\mathbf{A}$; $\mathbf{D}_{row} = diag(\mathbf{A}\mathbf{e})$ is the diagonal matrix with row sum of $\mathbf{A}$.

Then we use eigen-decomposition to find the eigenvectors ($\mathbf{U}_h$) and eigenvalues ($\mathbf{\Lambda}_h$):

$$\hat{\mathbf{A}} \approx \mathbf{U}_h \mathbf{\Lambda}_h {\mathbf{U}_h}^{\mathrm{T}}. \tag{18}$$

Step 2: We would like to generate a DEEPWALK matrix:

$$\mathbf{M} = \frac{vol(G)}{bT} \left(\sum_{r=1}^{T} (\mathbf{D}^{-1}\mathbf{A})^r \right) \mathbf{D}^{-1}, \tag{19}$$

where $vol(G) = \sum_i \sum_j A_{ij}$ is the volumn of the weighted graph G, b is the number of negative samples, T is the context window size in the skip-gram model.

We approximate $\mathbf{M}$ with

$$\hat{\mathbf{M}} = \frac{vol(G)}{b} \mathbf{D}^{-1/2} \mathbf{U}_h \left(\frac{1}{T} \sum_{r=1}^{T} \Lambda_h^r \right) {\mathbf{U}_h}^{\mathrm{T}} \mathbf{D}^{-1/2}. \tag{20}$$

Then make all the entries in $\hat{\mathbf{M}}$ at least one: $\hat{\mathbf{M}} = max(\hat{\mathbf{M}}, 1)$.

Step 3: We use rank-d approximation by SVD:

$$log\hat{\mathbf{M}} \approx \mathbf{U}_d \mathbf{\Sigma}_d {\mathbf{V}_d}^{\mathrm{T}}. \tag{21}$$

Use $\mathbf{U}_d \sqrt{\mathbf{\Sigma}_d}$ as the embedding vectors of atoms.

C. Similarity Ranking in Table Filling

We apply the ranking method in the table filling task based on the atomic embedding. For a candidate element and a candidate position in the table, we calculate the *summation* of the similarities between the embedding of the candidate element and embeddings of all the known neighbor elements of the candidate position. We use *Pearson correlation* in the calculation and fill the table with the highest *summation* (see Figure 4 and Task 1-3).

V. EXPERIMENTS

In this section, we introduce the data set and competitive methods. Then we do multiple tasks. For each task, we present task description, evaluation method, and result analysis.

A. Experimental Settings

1) Data Description: We obtain the inorganic crystal data from the Materials Project, a materials genome approach to accelerating materials innovation [37]. The data set has 83,990 chemical compounds, including information of formula, energy, and crystal structures. The number of unique atoms is 87. We find 2.8 unique atoms per chemical compound on average.

In the evaluation phase, we focus on the *main-group elements*. They are the 34 nonradioactive elements of the following groups:

- Hydrogen ($\boldsymbol{H}$),
- Alkali metals ($\boldsymbol{Li}$, $\boldsymbol{Na}$, $\boldsymbol{K}$, $\boldsymbol{Rb}$, $\boldsymbol{Cs}$),
- Alkaline earth metals ($\boldsymbol{Be}$, $\boldsymbol{Mg}$, $\boldsymbol{Ca}$, $\boldsymbol{Sr}$, $\boldsymbol{Ba}$),
- Boron group ($\boldsymbol{B}$, $\boldsymbol{Al}$, $\boldsymbol{Ga}$, $\boldsymbol{In}$, $\boldsymbol{Tl}$),
- Carbon group ($\boldsymbol{C}$, $\boldsymbol{Si}$, $\boldsymbol{Ge}$, $\boldsymbol{Sn}$, $\boldsymbol{Pb}$),
- Pnictogens ($\boldsymbol{N}$, $\boldsymbol{P}$, $\boldsymbol{As}$, $\boldsymbol{Sb}$, $\boldsymbol{Bi}$),
- Oxygen group ($\boldsymbol{O}$, $\boldsymbol{S}$, $\boldsymbol{Se}$, $\boldsymbol{Te}$),
- and Halogens ($\boldsymbol{F}$, $\boldsymbol{Cl}$, $\boldsymbol{Br}$, $\boldsymbol{I}$).

We obtain the elapsolites with their formation energy from a previous work [38]. It includes 10,556 elpasoltes. Because we only evaluate the embeddings of main-group elements, we have 5,628 elpasoltes for experiments and analysis.

2) Competitive Methods: The major baseline of atomic embedding is produced by the PNAS work [7]. It uses only the composition information and thus named as "Composition only" in performance comparisons.

When we reproduced The major baseline of atomic embedding is produced by the PNAS work [7]. It uses only

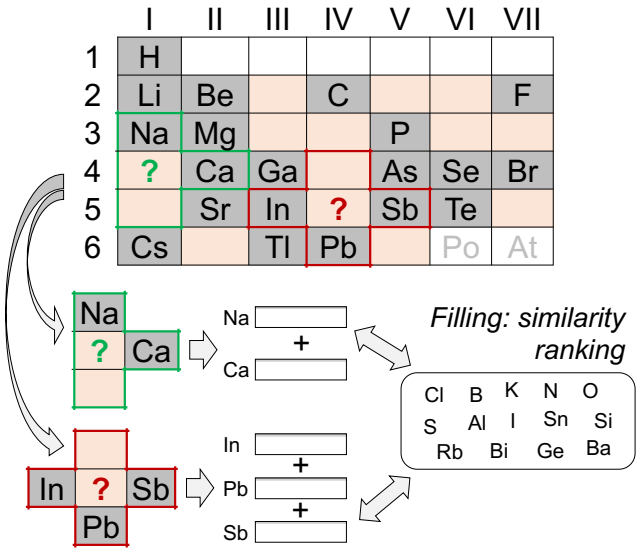


Fig. 4. When predicting an atom's neighbors (Task 1), predicting an atom's position in the table (Task 2), or filling an atom into the table (Task 3), we rank and select the proper atom or position of the highest summation of similarities between the atom candidate's embedding and its known neighbors' embeddings. We tried "average" and found the performance did not change much. We give two examples in the figure.

the composition information and thus named as "Composition only" in performance comparisons.

When we reproduced the results in [7], surprisingly we found out that random embedding (named as "Random") that was generated from a uniform distribution performs not much worse than "Composition only". Therefore, we also use it as one of the baselines to see how much "chemical semantics/properties" are successfully preserved in the embeddings.

Our proposed learning method generates atomic embedding for each level of crystal structures. We name them as " Compo.+Crystal System", "Compo.+Point Group", and "Compo.+Space Group".We set the number of dimensions d as 40 for all the embeddings.

B. Results on Filling the Periodic Table

In this section, we apply the atomic embedding to three tasks and compare their performances. We apply the same ranking method for all the evaluated embeddings. we rank and select the proper atom or position of the highest *summation* of similarities between the atom candidate's embedding and its known neighbors' embeddings.

We use *Pearson correlation* between the atom candidate's embedding and neighbor atoms' embeddings to assess the atom being filled into this position.Figure 4 illustrates the idea of table filling on all the three tasks. Specifically for Task 1 & 2, we assume that all the neighbors are available, while for Task 3, the assumption may not hold because one or more neighbors may be unknown.

1) Task 1: Neighbor-based Atom Prediction:

TABLE I
RESULTS ON TASK 1 (NEIGHBOR-BASED ATOM PREDICTION): THE ATOM EMBEDDINGS WITH COMPOSITION AND SPACE GROUP INFORMATION PERFORMS THE BEST ON TOP-k PRECISION.

	Prec@1	**Prec@3**	**Prec@5**
Random	0.0000	0.0294	0.0882
Composition only [7]	0.4118	0.5294	0.7941
Compo.+Crystal System	0.4412	0.7647	0.8529
Compo.+Point Group	0.4706	0.7647	**0.8824**
Compo.+Space Group	**0.5588**	**0.8235**	**0.8824**

a) Task description: For each of the main-group element $\boldsymbol{X}$, we assume that we are only given the neighbors of its position on the periodic table and asked to find this element out from $34 - n$ element candidates, where n is the number of neighbors:

- $n = 1$: $\boldsymbol{X}$ includes $\boldsymbol{H}$ (1);
- $n = 2$: $\boldsymbol{X}$ includes $\boldsymbol{F}$, $\boldsymbol{I}$, $\boldsymbol{Cs}$, and $\boldsymbol{Bi}$ (4);
- $n = 3$: $\boldsymbol{X}$ includes $\boldsymbol{Li}$, $\boldsymbol{Be}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{N}$, $\boldsymbol{O}$, $\boldsymbol{Na}$, $\boldsymbol{Cl}$, $\boldsymbol{K}$, $\boldsymbol{Br}$, $\boldsymbol{Rb}$, $\boldsymbol{Te}$, $\boldsymbol{Ba}$, $\boldsymbol{Tl}$, and $\boldsymbol{Pb}$ (15);
- $n = 4$: $\boldsymbol{X}$ includes $\boldsymbol{Mg}$, $\boldsymbol{Al}$, $\boldsymbol{Si}$, $\boldsymbol{P}$, $\boldsymbol{S}$, $\boldsymbol{Ca}$, $\boldsymbol{Ga}$, $\boldsymbol{Ge}$, $\boldsymbol{As}$, $\boldsymbol{Se}$, $\boldsymbol{Sr}$, $\boldsymbol{In}$, $\boldsymbol{Sn}$, and $\boldsymbol{Sb}$ (14).

b) Evaluation methods: We use Precision@1, Precision@3, and Precision@5 to evaluate the performance. As the smallest number of atom candidates is $34 - 4 = 30$, it is not easy to correctly predict every element with the neighborhood information only.

c) Experimental results: Table I shows the performance of the atomic embeddings on Task 1. We have the following observations. *First*, all the embeddings that combines composition and crystal structure information perform better than the baseline (Composition only). Compo.+Space Group performs the best, achieving a Prec@1 of 0.5588, a Prec@3 of 0.8235, and a Prec@5 of 0.8824. It improves relatively by **+35.7%**, **+55.6%**, and **+11.1%** over the baseline, respectively. *Second*, the Random embedding performs rather poorly, showing that the atomic embeddings preserve atomic information and become effective in filling the periodic table. *Third*, the highest Prec@1 of 0.5588 means that the best embedding put the correct answer at the top of list for only 19 positions among 34. The number of hits becomes as big as 28 when we use Prec@3. This means that the atomic embeddings can find the small set of best candidates. Some atoms have similar properties thus making it difficult to pick the correct answer.

2) Task 2: Neighbor-based Position Prediction:

a) Task description: For each main group element $\boldsymbol{X}$, we want to predict the correct position in the periodic table. For each candidate position, all the neighbor elements are given. Again, we use *Pearson correlation* and *summation* of similarities.

b) Evaluation methods: As the same as Task 1, we use Precision@1, Precision@3, and Precision@5 to evaluate the performance.

TABLE II
RESULTS ON TASK 2 (NEIGHBOR-BASED POSITION PREDICTION): THE ATOM EMBEDDINGS WITH CRYSTAL STRUCTURE INFORMATION PERFORMS BETTER THAN THE EMBEDDINGS W/O IT.

	Prec@1	**Prec@3**	**Prec@5**
Random	0.0000	0.0588	0.0882
Composition only [7]	0.3529	0.5588	0.7647
Compo.+Crystal System	0.5000	**0.7059**	0.7941
Compo.+Point Group	**0.5294**	**0.7059**	0.7941
Compo.+Space Group	0.5000	**0.7059**	**0.8235**

c) Experimental results: Table II shows the performance of the atomic embeddings on Task 2. We have the following observations. *First*, all the embeddings that combines composition and crystal structure information perform better than the baseline (Composition only). It is hard to tell which level of the crystal structure performs the best because their performances are close. Generally, the proposed embeddings achieve a Prec@1 of 0.5294, a Prec@3 of 0.7059, and a Prec@5 of 0.8235. It improves relatively by **+50.0%**, **+26.3%**, and **+7.7%** over the baseline, respectively. *Second*, the Random embedding performs rather poorly, showing again that the atomic embeddings do preserve the atoms' properties.

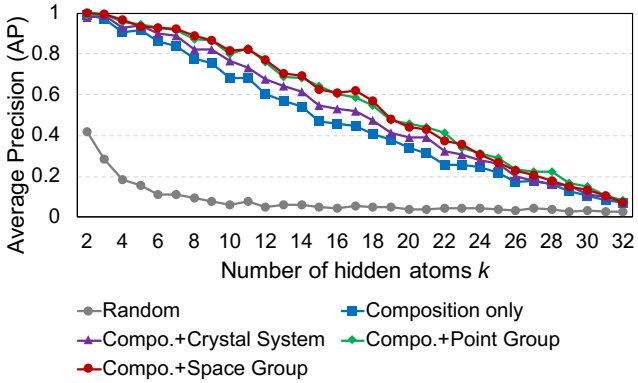
3) Task 3: Table Filling:

a) Task description: Given the "complete" periodic table of 34 main-group elements, we randomly hide k atoms and place them back to the slots. We use the same method to fill the table. At the beginning, we have k empty positions and k atoms to fill. It is an iterative process (k iterations). For each iteration, we calculate the *Pearson correlation* for each atom candidate and each empty position's neighbor atoms. We choose the pair of the highest correlation: we make a decision to fill the position with the corresponding atom. Then we remove the position and atom from the candidate sets. The table is completed after k trials.

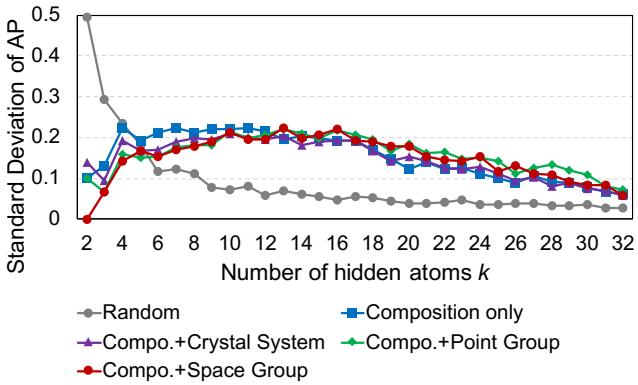
b) Task description and evaluation methods: Given a specific number k (number of hidden atoms), we do 100 trials: For each trial, we hides k atoms, fills them back into the table, and evaluate the accuracy. We use two metrics to evaluate the performance:

- Average Binary Precision (ABP): For each trial, only when all the placements are correct, i.e., all the k atoms are placed at the correct positions, we count a precision of 1 for this trial. ABP is the average score of the precision of all the trials.
- Average Precision (AP): We calculate a precision for each trial and then report the average score. The precision for each trial is calculated as the correctly-positioned atoms over k. It could be a number between 0 and 1.

Higher ABP or higher AP means better performance. Note that ABP is not bigger than AP, so it is difficult to obtain a high ABP. When $k = 2$, AP is the same as ABP.



(a) Compo.+Point/Space Group perform the best on Average Precision.



(b) All the embeddings except Random have very similar standard deviation.

Fig. 5. Results on Task 3 (Table Filling) when the number of hidden atoms k varies: The atom embeddings with both composition and crystal structure information generate the best Average Precision and reasonable standard deviation.

c) Experimental results: We vary the number of hidden atoms k from 2 to 32 and report the results in Table III, Figure 5, and Figure 6.

Table III shows the ABP and AP scores of the atomic embeddings under different settings of k. We give the results when k is 2, 3, 5, 8, 10, and 14. We have the following observations.

First, all the embeddings that combines composition and crystal structure information perform better than the baseline (Composition only). Compo.+Point Group performs the best when k is 2, 3, or 5 (small). Compo.+Space Group is the best when k is 8, 10, or 14 (big). The embedding with structure information significantly improves the AP relatively by **+1.1%**, **+2.4%**, **+3.3%**, **+14.0%**, **+20.0%**, and **+28.4%**, when k is 2, 3, 5, 8, 10, and 14, respectively. Clearly, when k is bigger, the improvement is more significant: when k is bigger, the table filling problem becomes more challenging because fewer atoms are known.

Second, the Random embedding performs poorly, showing that the atomic embeddings did preserve important information of the atoms and become effective in filling the periodic table.

TABLE III

RESULTS ON TASK 3 (TABLE FILLING): THE ATOM EMBEDDINGS WITH COMPOSITION PLUS SPACE GROUP INFORMATION OFTEN PERFORMS THE BEST. FOR EACH TRIAL, WE HIDE k ATOMS AND FILL THEM BACK TO THE TABLE. AVERAGE BINARY PRECISION (ABP) COUNTS 0 OR 1 FOR EACH TRIAL: ONLY WHEN ALL THE ATOMS ARE PLACED CORRECTLY, IT COUNTS 1; OTHERWISE 0. AVERAGE PRECISION (AP) COUNTS THE PRECISION FOR EACH TRIAL, SAY, THE NUMBER OF ATOMS CORRECTLY PLACED OVER k. (HIGHER ABP/AP MEANS BETTER PERFORMANCE.)

# Hidden atoms k	2	3		5		8		10		14	
	ABP	**ABP**	**AP**	**ABP**	**AP**	**ABP**	**AP**	**ABP**	**AP**	**ABP**	**AP**
Random	0.4367	0.1081	0.2716	0.0100	0.1540	0.0000	0.0971	0.0000	0.0600	0.0000	0.0588
Composition only [7]	0.9786	0.9333	0.9537	0.8200	0.9140	0.3900	0.7769	0.2000	0.6800	0.0500	0.5413
Compo.+Crystal System	0.9840	0.9504	0.9658	**0.8700**	0.9380	0.4900	0.8233	0.3100	0.7630	0.0400	0.6124
Compo.+Point Group	**0.9893**	**0.9661**	**0.9767**	**0.8700**	**0.9440**	0.6100	0.8698	0.4500	0.8010	**0.1900**	0.6823
Compo.+Space Group	**0.9893**	**0.9661**	0.9765	0.8500	0.9310	**0.6700**	**0.8860**	**0.4800**	**0.8160**	**0.1900**	**0.6951**

Third, the AP is as high as 0.8160 when $k = 10$; the AP is 0.6951 when $k = 14$. Our proposed atomic embeddings can accurately fill the atoms in the table without prior knowledge (e.g., the electrons arrangement of elements).

Figure 5 presents the curves of Average Precision (AP) vs. the number of hidden atoms k: (a) is for the mean value of AP scores on the 100 trials; (b) is for the standard deviation.

From Figure 5(a) we have the following observations. *First*, the Random embedding performs poorly (the grey line). *Second*, it matches our intuition that the performance would be worse when k becomes bigger. *Third*, the embeddings that combines composition and crystal structure information (purple, green, and red) perform better than Composition only (blue). *Lastly*, we find that Compo.+Point Group and Compo.+Space Group have similar performances.

As shown in Figure 5(b), the standard deviations of the atomic embeddings are similar with each other. It is around 0.2 when k is between 5 and 20, which is a bit high. This shows that the 34 main-group atoms have different difficulty levels of being learned into numerical representations and filled into the periodic table.

A case study: Let's look at a specific case when k is 14. So given 20 atoms and their positions, we use the atomic embeddings to put the remaining 14 atoms back into the table.

Figure 6 shows (a) the ground truth and (b–f) how the table-filling results of different atomic embedddings. The Random embedding fills only one element ***Ge*** correctly. The baseline method (Composition only) [7] fills 5 elements correctly among 14. When the top level of crystal structures, i.e., crystal system, is considered in the environment, the ***S*** and ***I*** are correctly positioned, making the number of correct placements 7. When the second level (point group) is considered, another pair of elements, ***B*** and ***N***, are correctly placed in the table. The number of correct placements increases to be 9. Finally, the third level (space group)-based atomic embedding fills 10 positions correctly!

It is also interesting to look at the remaining wrong prediction by Compo.+Space Group in this case: (1) the pair of ***K*** and ***Ba*** and (2) the pair of ***Al*** and ***Ge***. Due to inert pair effect, the properties of ***Tl*** are similar with ***K***, so ***K*** was placed as the neighbor of ***Tl***. ***Al*** and ***Ge*** are on the diagonal, which often indicates similar properties.

TABLE IV

RESULTS ON TASK 4 (ELPASOLITE FORMATION ENERGY PREDICTION): THE ATOM EMBEDDINGS WITH COMPOSITION AND CRYSTAL SYSTEM INFORMATION PERFORMS THE BEST ON MEAN ABSOLUTE PRECISION (MAP). (SMALLER MAP MEANS BETTER PERFORMANCE.)

	MAE: Mean $\pm$ Std (eV/atom)
Random	0.13310 $\pm$ 0.00874
Composition only [7]	0.12182 $\pm$ 0.00816
Compo.+Crystal System	**0.11833 $\pm$ 0.00582**
Compo.+Point Group	0.11915 $\pm$ 0.00610
Compo.+Space Group	0.12198 $\pm$ 0.00958

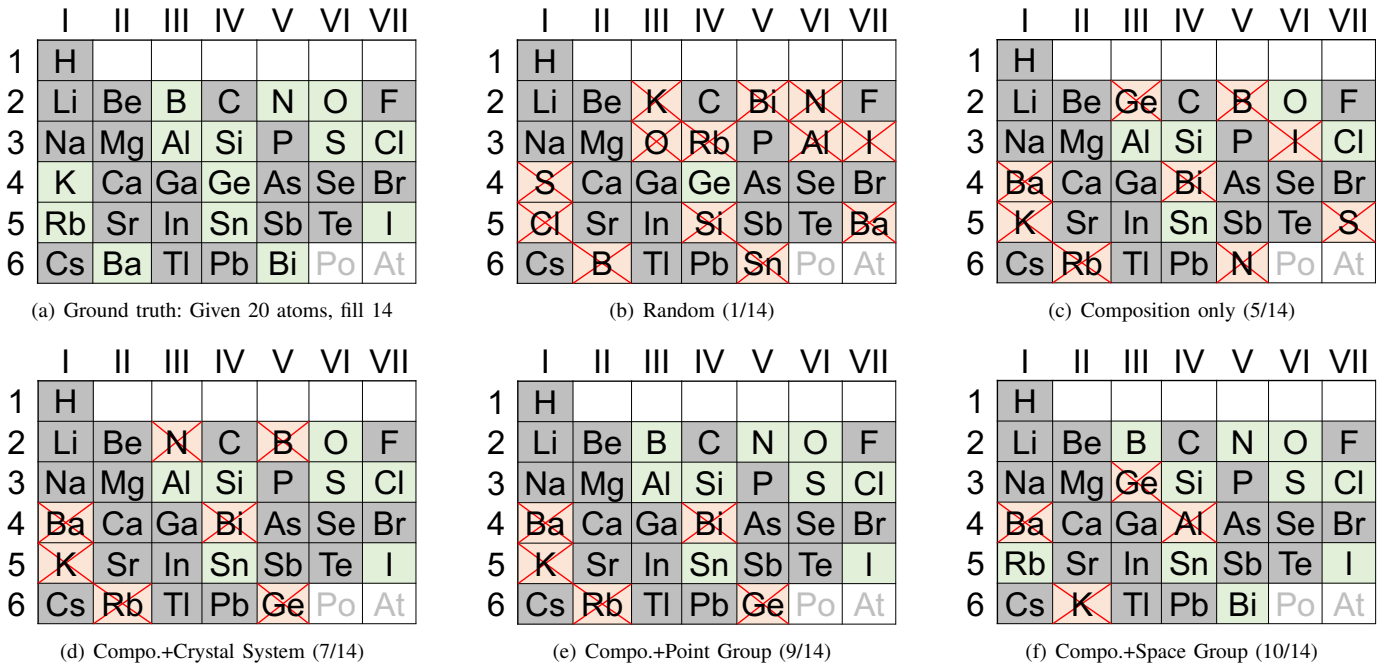
C. Results on Materials Discovery

In this section, we apply the atomic embeddings to a standard task that has been used in [7] and compare their performances.

1) Task 4: Elpasolite Formation Energy Prediction: In this section, we describe the task, introduce evaluation methods, and give and analyze the results.

a) Task description: Elpasolite has the form of $\boldsymbol{ABC_2D_6}$. We assign each elpasolite a feature vector by concatenating the embeddings of all four elements (***A***, ***B***, ***C*** and ***D***). Here ***A***, ***B***, ***C*** and ***D*** are all main-group elements. The task is to train a predictive model for predicting the formation energy of elpasolites. The embedding of the elapsolite is put into a two-layer neural network trained by the numerical label, i.e., formation energy. In the neural network, there are 10 neurons in the first layer and one neuron in the second layer. Rectified gated linear unit (ReLU) is put between two intermediate layers as activation function.

b) Evaluation methods: We implement hold-out and split the set of elapsolites into the training set (80%), validation set (10%), and testing set (10%). We use mean absolute error (MAE) as both loss function and monitor of validation set. When the error of validation set is not decreasing, we stop our training process and calculate the MAE of test set. We implement the neural network model in Keras [39]. The number of epochs is set as 1000. The batch size is set as 32. The learning rate of the Adam optimizer is 0.01 and the



(a) Ground truth: Given 20 atoms, fill 14

(b) Random (1/14)

(c) Composition only (5/14)

(d) Compo.+Crystal System (7/14)

(e) Compo.+Point Group (9/14)

(f) Compo.+Space Group (10/14)

Fig. 6. Results on Task 3 (Table Filling): The atom embeddings with both composition and crystal structure information fill the periodic table more accurately.

decay in training is 0.0001. Mean absolute error(MAE) and standard deviation(Std) of test set are utilized to evaluate the performance.

c) Experimental results: Table IV shows the performance of the atomic embeddings on Task 4. We have the following observations. *First*, the Compo.+Crystal System performs the best: it makes the smallest MAE on both mean value and standard deviation. The other two proposed embeddings perform a bit worse but not too much. *Second*, the baseline (Composition only) performs not bad either. Surprisingly, we observe that the random embedding can also generate a good performance. The reason is that the neural network model learns to predict the formation energy from not only the embedding of atoms but also the compositions of the elpasolites. Though the embeddings distribute randomly, the elpasolite composition contributes to the prediction with essential information. We conclude with the importance of using Task 1–3 (the Periodic Table Filling) for evaluation: Formation energy prediction cannot directly evaluate the usefulness of the atomic embeddings.

VI. Conclusions

In this paper, we applied representation learning algorithms to the *chemistry* domain. Given a large set of chemical compounds, the algorithms learns the embeddings of atoms for interesting tasks such as filling the atoms into the periodic table and predicting the formation energy of elpasolite. One of our primary contributions is that we reveal the important role of crystal structure information in the atomic embeddings. Our algorithms preserve not only the compounds' composition but also the crystal structures such as the crystal system, point group, or space group. Through an extensive set of experiments we demonstrated the effectiveness of the proposed algorithms. One interesting result was that given 20 atoms in the periodic table, our method could achieve an accuracy of 70%, while the baseline achieved only 54% accuracy for filling 14 hidden atoms into the Periodic table.

Different from traditional networks, crystals in three-dimensions contain intrinsic hierarchical symmetric elements: crystal systems, point group, and space group. In future work, we intend to move beyond treating the symmetric elements as extra attributes utilizing existing methods. Instead, we intend to embed the symmetric elements inside the embeddings with some pattern extraction methods or neural network related techniques. We can gain better and more universal atomic/elemental embeddings to design and predict materials.

Another future goal is to implement the constraints related to symmetric properties directly in the loss function. Current methods tend to sample input to the neural network or design target functions to approximate constraints. We may be able to design innovative models to jointly learn other useful embeddings from symmetric elements and the crystal properties.

Acknowledgements

We thank Shou-Cheng Zhang for his innovative work in this field of study.

REFERENCES

[1] N. N. Greenwood and A. Earnshaw, "Chemistry of the elements," 1984.

[2] L. M. Ghiringhelli, J. Vybiral, S. V. Levchenko, C. Draxl, and M. Scheffler, "Big data of materials science: critical role of the descriptor," *Physical review letters*, vol. 114, no. 10, p. 105503, 2015.

[3] A. Agrawal and A. Choudhary, "Perspective: Materials informatics and big data: Realization of the "fourth paradigm" of science in materials science," *Apl Materials*, vol. 4, no. 5, p. 053208, 2016.

[4] B. Kang and G. Ceder, "Battery materials for ultrafast charging and discharging," *Nature*, vol. 458, no. 7235, p. 190, 2009.

[5] J. K. Nørskov, T. Bligaard, J. Rossmeisl, and C. H. Christensen, "Towards the computational design of solid catalysts," *Nature chemistry*, vol. 1, no. 1, p. 37, 2009.

[6] S. Curtarolo, G. L. Hart, M. B. Nardelli, N. Mingo, S. Sanvito, and O. Levy, "The high-throughput highway to computational materials design," *Nature materials*, vol. 12, no. 3, p. 191, 2013.

[7] Q. Zhou, P. Tang, S. Liu, J. Pan, Q. Yan, and S.-C. Zhang, "Learning atoms for materials discovery," *Proceedings of the National Academy of Sciences*, vol. 115, no. 28, pp. E6411–E6417, 2018.

[8] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[9] F. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, "Crystal structure representations for machine learning models of formation energies," *International Journal of Quantum Chemistry*, vol. 115, no. 16, pp. 1094–1101, 2015.

[10] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[11] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[12] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 701–710.

[13] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[14] H. Gui, J. Liu, F. Tao, M. Jiang, B. Norick, and J. Han, "Large-scale embedding learning in heterogeneous event data," in *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 2016, pp. 907–912.

[15] Y. Shi, H. Gui, Q. Zhu, L. Kaplan, and J. Han, "Aspem: Embedding learning by aspects in heterogeneous information networks," in *Proceedings of the 2018 SIAM International Conference on Data Mining*. SIAM, 2018, pp. 144–152.

[16] J. Qiu, Y. Dong, H. Ma, J. Li, K. Wang, and J. Tang, "Network embedding as matrix factorization: Unifying deepwalk, line, pte, and node2vec," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*. ACM, 2018, pp. 459–467.

[17] Z. Wang, J. Zhang, J. Feng, and Z. Chen, "Knowledge graph embedding by translating on hyperplanes." in *AAAI*, vol. 14, 2014, pp. 1112–1119.

[18] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion." in *AAAI*, vol. 15, 2015, pp. 2181–2187.

[19] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2d knowledge graph embeddings," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[20] W. Liu, Z. Li, and X. Tang, "Spatio-temporal embedding for statistical face recognition from video," in *European Conference on Computer Vision*. Springer, 2006, pp. 374–388.

[21] D. Wang, M. Jiang, Q. Zeng, Z. Eberhart, and N. V. Chawla, "Multi-type itemset embedding for learning behavior success," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 2397–2406.

[22] P. Levi, *The periodic table*. Everyman's Library, 1996, vol. 218.

[23] E. R. Scerri, *The periodic table: its story and its significance*. OUP USA, 2007.

[24] B. J. McFarland, *A World from Dust: How the Periodic Table Shaped Life*. Oxford University Press, 2016.

[25] S. E. Ahnert, J. A. Marsh, H. Hernández, C. V. Robinson, and S. A. Teichmann, "Principles of assembly reveal a periodic table of protein complexes," *Science*, vol. 350, no. 6266, p. aaa2245, 2015.

[26] D. Wang, P. Cui, and W. Zhu, "Structural deep network embedding," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 1225–1234.

[27] X. Wang, P. Cui, J. Wang, J. Pei, W. Zhu, and S. Yang, "Community preserving network embedding," in *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[28] D. Zhu, P. Cui, D. Wang, and W. Zhu, "Deep variational network embedding in wasserstein space," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 2827–2836.

[29] D. Zhu, P. Cui, Z. Zhang, J. Pei, and W. Zhu, "High-order proximity preserved embedding for dynamic networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 11, pp. 2134–2144, 2018.

[30] J. Ma, P. Cui, X. Wang, and W. Zhu, "Hierarchical taxonomy aware network embedding," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 1920–1929.

[31] Z. Zhang, P. Cui, X. Wang, J. Pei, X. Yao, and W. Zhu, "Arbitrary-order proximity preserved network embedding," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 2778–2786.

[32] I. Abarenkov, M. Boyko, and P. Sushko, "Embedding and atomic orbitals hybridization," *International Journal of Quantum Chemistry*, vol. 111, no. 11, pp. 2602–2619, 2011.

[33] J. Behler and M. Parrinello, "Generalized neural-network representation of high-dimensional potential-energy surfaces," *Physical review letters*, vol. 98, no. 14, p. 146401, 2007.

[34] S. Gong, W. Wu, F. Q. Wang, J. Liu, Y. Zhao, Y. Shen, S. Wang, Q. Sun, and Q. Wang, "Classifying superheavy elements by machine learning," *Physical Review A*, vol. 99, no. 2, p. 022110, 2019.

[35] Z. Quan, X. Lin, Z.-J. Wang, Y. Liu, F. Wang, and K. Li, "A system for learning atoms based on long short-term memory recurrent neural networks," in *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2018, pp. 728–733.

[36] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 855–864.

[37] A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. a. Persson, "The Materials Project: A materials genome approach to accelerating materials innovation," *APL Materials*, vol. 1, no. 1, p. 011002, 2013. [Online]. Available: http://link.aip.org/link/AMPADS/v1/i1/p011002/s1&Agg=doi

[38] F. A. Faber, A. Lindmaa, O. A. Von Lilienfeld, and R. Armiento, "Machine learning energies of 2 million elpasolite (a b c 2 d 6) crystals," *Physical review letters*, vol. 117, no. 13, p. 135502, 2016.

[39] F. Chollet *et al.*, "Keras," https://keras.io, 2015.