博士論文

Material Property Predictions and Discovery Using a Novel Descriptor "Elemental Fingerprints" with Neural Networks

（新規記述子「元素フィンガープリント」を用いたニューラルネットワークによる材料物性の予測及び材料探索）

Jaekyun Hwang

黄　濈鈞

Contents

Chapter 1 Machine learning and materials informatics

This dissertation aims to develop a novel descriptor for the application of machine learning to materials science and engineering and to test the effectiveness and usefulness of this new descriptor. In this introduction chapter, we explore how machine learning (or deep learning, artificial intelligence), actively applied to various issues in a wide variety of fields with the recent advancement of computer technology and information science, can be applied to materials science [1-5].

From simple prediction problems starting with linear regression, to complicated problems like detecting a person in an image and playing the game of Go, machine learning has shown processing capability that goes beyond human ability even in the fields where machines traditionally could not surpass humans, and continues evolving. Many of these outstanding machine learning achievements are made by a model known as artificial neural network [6], which is conquering complicated tasks, one after another, that could not be done by traditional models that can be explained and understood by humans. At the same time, there are active movements in various fields to utilize machine learning techniques like neural network in order to overcome limitations.

Machine learning is also being tried out in the field of materials science, from simple prediction of physical property to interpretation of experimental results and suggestion of candidates for materials discovery. Among them, we would like to focus on the prediction of materials properties (e.g., stability, strength, conductivity). However, there are many obstacles to overcome in order to combine actual materials research and development of materials with machine learning. Many approaches to solve this problem are often valid only to a limited extent, so in this dissertation, we would like to study and propose a more comprehensive method (prediction model and descriptor).

The following Subchapter 1.1 briefly introduces groundbreaking machine learning studies. Though they are not directly related to this study, we introduce them in order to enhance our understanding of the development of various related technologies. In Subchapter 1.2, we introduce several studies that apply machine learning to materials science in various ways. In

Subchapter 1.3, we will introduce the descriptors for materials that are used for machine learning. A descriptor for materials means the method of converting the information of materials into input features for machine learning models. It is very important to use efficient descriptors because machine learning results using the same information can vary greatly depending on the descriptors. In Subchapter 1.3, we will investigate the advantages and disadvantages of various descriptors, including the descriptors used in the study introduced in Subchapter 1.2. In Subchapter 1.4, we point out the remaining limitations for incorporating machine learning into the research and development of materials. Then we investigate the conditions that the descriptors should satisfy to apply machine learning to the materials research and development more generally.

1.1 Recent progress of machine learning

Machine learning, in short, is a generic term for technologies that bring out hidden relationships among data to extract various previously unknown information. Machine learning is a research field that has been spotlighted as a novel methodology made up of the accumulated knowledge of artificial intelligence and advanced computer technology. Image classification is one of the fields which is receiving great attention because of high performance of deep learning. This field has a wide range of applications, from classifying the hand-written zip code to recognizing traffic signs for automatic driving. In the field of image classification, before the advent of deep learning, there had been attempts to create a classification model for the pictures of hand-written digits. These attempts had been made by using NIST or MNIST dataset [7]. Then, the deep learning made a breakthrough. Recognition error is about 7.6% when using a pairwise model, one of the classical classification models. Using deep learning, Yann et al. succeeded in achieving much better performance. They achieved a recognition error of 3.4% in 1990 using an early version of the neural network model [8], and 0.7% in 1998 using the neural network model named LeNet [9]. In 2012, Dan et al. achieved an error rate of 0.23% with a model called MCDNN [10], enabling them to distinguish numerical photographs at a similar level to humans for the first time. One of the reasons why this performance has attracted attention is that the deep learning model finds and

learns important features in the image on its own, rather than using designated features for traditional models. One of the reasons why this research was so remarkable is that it presented a model to find important features from data on its own, in contrast to the conventional model that learns a set of features specified by humans.

Both models of LeNet and MCDNN were constructed using a deep convolutional neural network (CNN), which is an effective neural network for two or higher-dimensional datasets [11]. For example, a single image's width and height consist of small units called pixels, and each pixel is made of three-color channels: red, green, and blue. Each pixel has a strong correlation with its surrounding pixels; therefore, instead of viewing each pixel as an individual data, a "filter" of a certain form can be applied to surrounding pixels to extract the features of the image. This process of modifying features of a function by applying another function is called convolution. In CNN, convolution is simultaneously used with a technique called pooling or stride to collect and strengthen image features. Since then, most machine learning techniques related to image recognition have been developed based on CNN. The advancement of technology has allowed CNN models to train on a larger number of images with higher accuracy.

From the above example, many researchers realized that if they can transform certain information into images in a natural way, they can conveniently conduct machine learning by using CNN. One major example is the game of Go. Programs that analyze abstract strategy games such as Go and Chess used alpha-beta pruning algorithm to analyze the possible moves until recently [12,13]. With the increase of computing power, these programs have become more and more powerful. However, alpha-beta pruning is less powerful as the number of cases increases. Therefore, the computer could defeat the world champion in chess, but could not defeat even an amateur in the game of Go, where the number of cases is much larger [14].

DeepMind Technologies Limited has been working on mastering the game of Go to overcome this situation. In 2015, they had developed a Go program known as AlphaGo, which won the first game against a Go professional player. They improved the AlphaGo model and announced AlphaZero in 2018, which does not require any human background knowledge. The new Go program outperformed both humans and other computer programs in the abstract strategy game field, and their achievements seemed unattainable in the past [15,16]. AlphaZero was designed to recognize a 19-by-19 Go board as a 19x19 pixel image and applied CNN to the pixel image. The AlphaZero attracted much attention, because it shows that the reinforcement learning

only using the game rules as input is better than the traditional alpha-beta algorithm and the combination of supervised learning and reinforcement learning using the records of past games between human players. This means that a model mimicking insight or intuition, which was considered to belong to a human domain, even though in a limited area. For example, a human Go player considers about 100 possible next moves, whereas traditional programs attempt to consider as many possibilities as possible based on fast computation powers, such as about over 10 million possible next moves. However, AlphaZero searches for only about 10,000 possible next moves, yet it makes much more effective choices than the traditional programs.

Another field in which machine learning has achieved excellent results is machine translation, in which computer calculations are used to translate one human natural language to another. The field began with a rule-based machine translation and has expanded to include many different methods. When using traditional methods other than deep learning such as the rule-based translator, the performance was limited. As machine learning technology has developed, machine translation has become useful in real life, such as travel, although it is not yet perfect. In 2016, the Google developed a neural machine translation system and got several interesting research results [17,18]. Google's model was different from the ones mentioned above in several aspects. One of the most significant differences is the use of natural human language as an object of learning. Since there is no way to know the word sequence length of each sentence in advance, it uses recurrent neural networks, a model with no limitation in the input length. Simply speaking, the recurrent neural networks receives one input (in this case, a sub-word), outputs a certain calculation result, and uses that output as an input again. It means recurrent neural network can have an arbitrary number of time steps. By adding this and several other new approaches, Google's model could learn many languages simultaneously. Using the training results of grammars and vocabularies of several language pairs, it can translate a novel language pair better than existing methods. By analyzing such results, machine learning is shown to have capability to create its own universal language that interconnects different languages together. This capability has high potential in other applications as well, such as preserving languages that are in danger of extinction due to a steadily decreasing number of users.

In this subchapter, we have explored various machine learning trends. Though they may seem irrelevant to the study of materials science, it is important to remain open to new research and explore the possibilities of combining them with the current materials research.

1.2 Machine learning for materials research

It is said that the history of material development began with the first use of tools by humans. Various advanced technologies of the time were utilized to develop new materials. Data-driven science, referred to as the fourth paradigm in scientific research after three paradigm shifts of experiment, theory, and computational modeling and simulation, is being actively attempted in various ways in the research and development of materials. This subchapter examines machine learning techniques that enhance the performance of some class of tasks by applying statistical methods in data-driven science. We examine several examples of potential energy surface prediction, the discovery of the descriptor that describing the material properties, and the method of predicting the material properties directly from the material compositions. A brief discussion on the application of machine learning in material development is also presented.

1.2.1 Prediction of potential energy surface

Potential energy surface (PES) describes the energy change with parameters such as the position of atoms in a material system [19]. It is used to evaluate the atomic arrangement and stability of a material, or to simulate chemical reactions of materials. PES can be calculated quite accurately owing to the recent advancement of computational modeling and simulation.

However, because an accurate PES calculation requires high computational cost, methods to quickly estimate PES using empirical potential were also studied. Starting from the classic Lennard-Jones potential [20] method with a small number of parameters, various empirical potentials have been proposed. Potentials containing comparatively more terms and parameters, such as the reactive force field (ReaxFF) [21], has been proposed to describe more complex situation accurately. The parameters of an empirical potential are estimated through experimental data or a small number of relatively accurate first-principles calculation data. Calculations using empirical potential requires smaller computational cost than first-principles calculations. Thus, it enables quick calculations of the energy states of atoms in complex systems. However, it has the drawback of relatively lower accuracy.

In recent years, machine learning-based potentials such as neural network potential (NNP) and Gaussian approximation potential (GAP) were developed. They draw attention as they enable faster PES estimation than first-principles calculations and more accurate than traditional empirical potentials. These machine learning-based potentials enable efficient material simulations based on rapid and accurate PES predictions. Through these methods, new approaches of materials interpretation can be achieved by enabling simulations that were difficult previously. To construct NNP or GAP, it is necessary to select a descriptor to express the atomic arrangement in a numerical form. Some of the descriptors that can be used to construct NNP or GAP are introduced with other descriptors in subsequent Subchapter 1.3.

1.2.2 Discovery of important factors

One easy approach to understand a material is to observe the changes in its properties when the material composition is changed. However, it is difficult to derive clear correlations behind such changes in most cases. Therefore, it often requires a lot of trial and error to understand new materials, which can cause delays in material development. Machine learning can facilitate the process of material research and development by revealing missed correlations and minimizing trial and error.

Balachandran et al. conducted recursive partitioning analysis to derive factors to better describe the stability of oxide perovskite [22], which had previously been described using the Goldschmidt tolerance factor [23]. As a result, they obtained a classification tree diagram that defines the phase stability of the oxide perovskite. They revealed that the most important factor to characterize the phase stability is the ideal A-O bond distance based on the bond-valence model ($d_{A\text{-}O}$), followed by the tolerance factor.

However, not all material properties can be explained by simple factors such as the above $d_{A\text{-}O}$. Ghiringhelli et al. proposed a method to generate a descriptor for material properties that cannot be explained with existing methods [24]. They generated candidate descriptors with mathematical combinations from the primary features of the elements constituting the material. Then, they presented an iterative approach to evaluate and derive the valid descriptors. Through this method, Ghiringhelli et al. were able to discover equations that explains the simulation results.

For example, they discovered the following equation for the formation energy difference ΔE between two crystal structures of octet binary compounds from two elements, A and B.

$$\Delta E = 0.117 \frac{\mathrm{EA(B)} - \mathrm{IP(B)}}{r_p(\mathrm{A})^2} - 0.342 \text{ (eV)} \tag{3-1}$$

Here, $\mathrm{EA(B)}$ and $\mathrm{IP(B)}$ are electron affinity and ionization potential of element B, respectively, and $r_p(A)$ is p orbital radius of element A. This equation has not been explained by the existing laws of physics. These methods are expected to accelerate our understanding and discovery of materials.

1.2.3 Prediction of materials properties

Another widely attempted method in materials research using machine learning is the construction of models for direct prediction of the material properties using the information of atomic arrangements and/or chemical composition as input. In particular, the prediction of material properties using only the raw material compositions (such as mole fraction) has been attempted over the years because compositional information can be obtained much more easily than structural information. For example, Winkelmann and Schott proposed an additive model for predicting the thermal conductivity of glass materials in 1893 [25]. This model employs a linear equation having the mole fraction of each raw material constituting the glass as the terms. However, the prediction models used in the research and development of modern glass materials have not advanced significantly from the initial additive model. One such model is the equation proposed by Fluegel in 2007 to predict the viscosity of glass [26]. Fluegel tried to improve the accuracy of the existing equation by adding terms that describe the product of the mole fractions of different raw materials to the multiple linear equation. Fluegel's equation again uses mole fractions of raw materials, excluding SiO_2 as the terms. Fluegel's equation has achieved sufficient accuracy for practical glass material development in consequence of big data and data preprocessing. However, his addictive model has a disadvantage that it cannot effectively predict the materials not included in the database, and no addictive model has been able to solve this weak point yet.

There are also several studies using classic predictive model in the prediction of formation enthalpy, one of the important properties of material. Among them, one of the studies by Ali et al. in 2007 predicts formation enthalpy using only compositional descriptors such as the number of oxygen atoms [27]. They used a linear regression similar to the addictive model. However, the prediction error of this model is 0.6 eV/atom, which is impractical. As an advanced prediction model using machine learning, the ElemNet was proposed by Jha et al. [28]. It employs a deep neural network, a deep learning model that can provide high accuracy, while using only the mole fractions of each element as the input features. Using the ElemNet model, Jha et al. succeeded in predicting the formation enthalpy of crystalline solids composed of various elements with mean absolute error of 50 meV/atom.

1.2.4 Discovery of new material using a Bayesian optimization

When we are trying to develop new material, predictions are often insufficient. Bayesian optimization can be useful in this case [29-32]. Seko et al. try to find a crystalline compound with low lattice thermal conductivity (LTC) for thermoelectric materials with high conversion efficiency [33]. In their study, by calculating the LTC of 101 compounds with known LTC experimental values using first-principles calculations, they verified the accuracy of the simulation. They then attempted to explore the compounds with low LTC among 54,779 candidates using as few simulations as possible from Bayesian optimization.

Among the top 221 materials proposed by the Bayesian optimization, they conducted LTC simulations on only nine of them, considering the high computational costs of first-principles simulation. As a result, the material with the lowest LTC was $PbRb_4Br_6$ with 0.08 W/mK, and further obtained two additional candidate materials, K_2CdPb and $Cs_2[PdCI_4]I_2$, which both had LTC of less than 0.5 W/mK while having electronic band gap energy between 0.1 eV and 1 eV according to the DFT-PBE formalism calculations. However, in this research, materials with a high probability of improvement were calculated only once based on the existing data, meaning that a full Bayesian optimization was not conducted. Nevertheless, considering that the LTC of the champion training material (PbSe) is 0.96 W/mK, success in finding a material having an LTC of 0.08 W/mK after the first try is worth paying close attention.

1.3 Material descriptors for machine learning

Some of the techniques introduced in Subchapter 1.1 are also being tried out in the materials research. The most significant difference between machine learning in the materials research and that of the image recognition and abstract strategy game is the insufficiency of curated data for machine learning in materials research. For example, in image recognition, a large database such as ImageNet is widely used as a benchmark for technological verification. The lack of large databases causes difficulty in testing machine learning techniques in materials research.

While many papers report the physical or chemical properties of materials, they arrange numerous variables, including experimental condition, process, result, etc. in their own way. As a result, in machine learning in materials research, only a small number of data are often available when a researcher tries to test his/her novel method. On the other hand, unlike experiments, independent variables are often well established in the first-principles calculations. Therefore, many tests are done by applying machine learning technology to first-principles calculation results.

When machine learning is used in a situation where the number of data and/or independent variables is limited, descriptors specially designed for the situation are often used. However, suppose there is a general descriptor that can be used in a wide range of situations in addition to specially designed descriptors. In that case, machine learning would be applied more efficiently in materials research. The general descriptor proposed in this study aims a descriptor that can be used generally regardless of the type of material or the type of research method, i.e. experiment or simulation.

In this subchapter, we examine how input data are prepared in previous studies from the view point of descriptors, and explore the current limitations of machine learning techniques in materials research.

Subchapters 1.3.1 to 1.3.3 introduce three descriptors that can be used when the atomic structure is known. These descriptors can play an important role in first-principles simulations on nanostructures. However, information about atomic structures is not necessarily sufficient for machine learning, because they are often obtained from experiments while it is difficult to make materials with arbitrary structures experimentally. In most cases, the input data or controllable

variables that can be used in machine learning are the experimental conditions such as temperature and pressure, and the composition of the materials.

Subsequently, starting with Subchapter 1.3.4, we will introduce descriptors that can be used when the information obtained from experiments is limited. In Subchapter 1.3.9, we summarize the descriptors introduced so far, and discuss the desirable characteristics that a newly developed should have under the current situation.

1.3.1 Atom-centered symmetry functions

The atom-centered symmetry functions proposed by Behler et al. are descriptors proposed to create an accurate potential energy model by quantifying the environment of each atom [34]. This descriptor is proposed for machine learning potential that aims to have the advantages of both density functional theory (DFT) [35,36] and empirical potential.

First-principles calculations based on DFT are widely utilized in materials science research to simulate various material properties [37,38]. However, DFT calculations generally have a complexity of $O(N^{2\sim3})$, which means that the simulation time is proportional to the square to cube of the number of atoms (or electrons) within the model. From this complexity problem, conducting the large size of simulation requires a high computational cost. Therefore, the empirical interatomic potential is often used in such situations. Empirical interatomic potential allows for even billion-atom MD simulations [39], but the energy surfaces of empirical potential are much less accurate than those of DFT calculations.

To compensate for these shortcomings, machine learning potential aims to create a potential energy prediction model that uses machine learning to achieve high computational speed and higher accuracy than traditional empirical potentials. However, prediction models adopting the Cartesian coordinates of atomic positions as the input has poor accuracy and scalability. That is, when the Cartesian coordinates are adopted, it is difficult to guarantee the invariance under the translation and rotation of the system in machine learning, and to apply machine-learned models to systems having different number of atoms from those of systems used in the training.

Atom-centered symmetry functions proposed by Behler et al. are designed to solve these problems by many-body functions that represent the surrounding environment of each atom in a

system, that is, the information of its neighboring atoms within a cutoff radius. Behler et al. proposed a way to predict the total energy of a system from its atomic arrangement by high-dimensional neural network potential (HDNNP). They expressed the total energy as the sum of atomic energies and suggested that each atomic energy can be predicted using a neural network.

Bose et al. calculated the atomic energy of gold atoms in various structures through 9,972 DFT calculations [40]. They trained an HDNNP using the calculation data and compared its prediction with ReaxFF, one of the traditional empirical potentials. They achieved an energy prediction error of 17 meV/atom using the HDNNP compared to 136 meV/atom using the ReaxFF, which shows the potential of NNP.

1.3.2 Smooth Overlap of Atomic Positions

Another descriptor that quantifies information of the three-dimensional atomic structure is Smooth Overlap of Atomic Positions (SOAP) proposed by Bartok et al. [41]. They offered a method for calculating the smooth similarity kernel to evaluate neighboring elemental environments. Similarity kernel refers to a method of expressing the similarity of three-dimensional structure through the kernel. The "smooth" in SOAP indicates that the similarity kernel is smoothly expressed using the Gaussian function. In other words, SOAP is a descriptor in which the information of atoms around an atom is expressed by the sum of Gaussians.

To create SOAP, the same information is required as the atom-centered symmetry functions discussed in the previous subchapter, i.e., the structural input data such as the crystal structure and the coordinates of atoms. Bartok et al. attempted to verify the effectiveness of SOAP using 8,000 tight-binding molecular dynamics calculations for Si clusters. They showed that the Gaussian approximation potential (GAP) [42] could be trained using SOAP. This GAP predicts the energy and the force of Si atoms within the error of 15 meV/atom and 0.22 eV, respectively. From these results, Bartok et al. claim that SOAP could well represent the dissimilarity of the atomic structures.

1.3.3 Crystal graph

Both Atom-centered symmetry functions and SOAP reconstruct information of the three-dimensional atomic structures using functions (symmetry functions and Gaussians, respectively). On the other hand, Xie et al. suggested a way to represent the atoms and their bondings by graph theory [43]. They made a crystal graph in which an atom is represented by a node and an atomic bonding by an edge. They also proposed a way to train this graph to predict material properties by convolutional neural networks (CNN).

This method cannot reflect the information of minute atomic displacement if the atomic bonds have remained intact. However, it can still distinguish between the two cases where two atoms are swapped in the crystal and between the two cases where the crystal structures are changed while the chemical formula remains the same. They showed that the crystal graph could predict several material properties obtained from the Materials Project database, which is one of the major databases of DFT calculation results.

As a result, for instance, the prediction error of the crystal graph is 0.054 GPa for bulk moduli, where the DFT calculation error compared with experiments is 0.050 GPa. In the case of formation energy, the prediction error is 39 meV/atom. This value is slightly higher than the prediction errors of NNP or GAP introduced in Subchapters 1.3.1 and 1.3.2, 6 meV/atom and 15 meV/atom, respectively. However, crystal graph has the merit of being able to predict the properties of the material from any combination of elements through a single model.

1.3.4 Pure elemental property

As simple and intuitive descriptors that can describe a material, the properties of the elements constituting the material or combinations of such properties can be used. There may be various methods for selecting a descriptor depending on the target materials and target properties to be described.

For example, Yi et al. proposed a model to evaluate the crystal stability by training using the DFT formation energies of the Garnet structures which can be represented by the chemical formula of $C_3A_2D_3O_{12}$ [44]. They used the electronegativity and the ionic radius of the three

elements C, A and D as descriptors. They succeeded in predicting the formation energy of the garnet structure with an error of 9 meV/atom with the neural network based on the rectified linear units.

However, this type of descriptor is only valid for a predetermined family of materials (in this case, a garnet structure that can be represented by the chemical formula of $C_3A_2D_3O_{12}$) and cannot be directly used for the training or prediction of other materials.

On the other hand, the advantage of this type of descriptors is that they could be highly interpretable. For example, Balachandran et al. proposed a classification model that describes the formability of perovskite oxides, which can be represented by ABO_3 [22]. Previously, the Goldschmidt tolerance factor [23] calculated using ionic radii was known to explain this, however, it was not perfect. According to the decision tree model by Balachandran et al., one factor is more critical than the Goldschmidt tolerance factor in determining whether perovskite structure can exist stably. The factor is the ideal A–O bond length (d_{A-O}). For example, if the d_{A-O} is less than 2.45 Å, the stable perovskite is never made regardless of the B element. However, setting up this type of interpretable descriptor (like the tolerance factor) often requires a deep understanding of the material and its operating mechanisms.

1.3.5 Non-linear combinations of primary features

The interpretable descriptor introduced in Subchapter 1.3.4 can provide logical reasoning for prediction results and can also significantly impact other areas through new understandings obtained from the logical reasoning. However, generating an interpretable descriptor requires a deep understanding of the materials and properties to be described or involves a lot of trial and error. Ghiringhelli et al. provided a solution to overcome this problem based on their study of the choice of descriptor [24].

In the paper, they tried to predict the stability of two octet AB-type binary compound structures, zinc blend and rocksalt. They tried to find the best descriptor that can predict the energy difference between the two structures calculated by DFT. To create candidate descriptors, 23 primary features were prepared from the properties of each element (A, B) and the properties of dimers that two elements could form. Then, they make more than 10,000 complex descriptors by

combining primary features and mathematical operations such as absolute difference, squares, and exponential. Subsequently, they identified the best combination of descriptors that predicts the energy difference between the two structures. As a result, the prediction error of the energy difference using only one descriptor is 0.14 eV, which correctly predicts 74 crystal structures, out of 82 binary octet compounds.

The descriptor proposed in the paper was not a physical quantity that can be understandable from existing physical laws. In 2019, Bartel et al. proposed a new tolerance factor τ based on the method described above to replace the existing Goldschmidt tolerance factor t, which explains the stability of the perovskite (ABX_3) structures [23]. τ is more accurate than t. Furthermore, by analyzing each term in τ, it shows that the stability of BX_6 octahedra and the oxidation state of A are important in determining the stability of the perovskite structure. This analysis illustrates how we can understand the mechanisms of a target property through a data-driven approach.

1.3.6 Molar concentrations

One of the most intuitive material descriptors is the molar concentrations or the presence of elements (or raw materials) in target materials. This method is discussed in the prediction of material properties using chemical composition introduced in Subchapter 1.2.3. Depending on the experiments, the proportion of each element or raw material that makes up the material can be used as a descriptor. Alternatively, more simply, the presence or absence of each element can be expressed as binary.

Models using this simple descriptor often successfully predict the material properties. Recently, one research group proposed the artificial neural network (ANN) model named ElemNet, which uses only the molar concentration as input [28].

ElemNet is an ANN model with 94 input features corresponding to the molar ratio from H to Pu. This study succeeded in predicting formation enthalpy within the mean absolute error of 50 meV/atom by using more than 200,000 first-principles calculation data and 17 hidden layer model. This is slightly less accurate than the prediction error of 39 meV/atom performed by the crystal graph and CNN model introduced in Subchapter 1.3.3. However, considering that the

information of atomic structures was not used or required in ElemNet, this result of ElemNet is encouraging.

Based on this result, they attempted to create a convex hull diagram of the binary and ternary systems that were not used for training. As a result, they were able to obtain the most similar formation enthalpy distribution to the actual DFT calculation results compared to the existing models. However, neither their ElemNet nor existing model has been able to accurately predict stable compositions. This failure suggests that machine learning model can be vulnerable to unseen data that are not used for training.

1.3.7 Elemental attributes

Properties of the material other than our target one can be used as an extra descriptor to predict the target property of a material. For example, we can use the density of a material as one of the descriptors to predict the thermal conductivity of a material. However, it is often difficult or expensive to obtain this type of information for many materials. On the other hand, the properties of elements or raw materials are relatively small in number and are often easy to obtain. For example, densities of more than 200,000 materials are difficult to obtain, while those of 94 pure elements is relatively easy to obtain.

To use this kind of information as efficiently as possible, Ward et al. proposed a new descriptor named elemental attributes [45]. They first collected 23 properties of pure elements (e.g., atomic number, electronegativity, etc.) from hydrogen to plutonium. Subsequently, the mean, minimum, maximum, deviation, and mode weighted by the mole fraction for each property can be calculated for each material. Then we obtain a total of 115 attributes that describe one material from the calculation of five statistics for 23 properties. They made models to predict material properties such as formation energy and bandgap energy using the elemental attributes. The reported prediction error for the formation enthalpy was 88 meV/atom, which is less accurate than that of ElemNet [28] despite the use of additional information about elemental properties in the elemental attributes compared with ElemNet. One reason for this poorer accuracy of the elemental attributes could be that the five statistics of elemental properties fail to catch some important information in the distribution of elemental properties.

1.3.8 B-fingerprint & D-fingerprint

B-fingerprint and D-fingerprint are the descriptors proposed by Isayev et al. to investigate the similarities among materials [46]. Both are built on electronic band structures calculated within DFT. This subchapter will introduce the B-fingerprint first.

B-fingerprint divides a certain energy region (e.g., ±10 eV from the Fermi level) into several parts (bins), then count the number of bands in each energy zone at the targeted symmetry point in k-space. For example, in the case of KNO_3 (ICSD 384), there is one energy band between −10 eV and −7.5 eV, three bands between −7.5 eV and −5.0 eV, and no energy band between −5.0 eV and −2.5 eV, and so on. Then, the B-fingerprint from eight bins for KNO_3 can be represented as [1, 3, 0, 5, 0, 2, 0, 3]. The total length of the fingerprint depends on the number of symmetry points being investigated and the number of bins. If we use 32 bins for a cubic lattice with four high symmetry points, the length of the whole fingerprint will be 128.

D-fingerprint is a descriptor in which the same procedure as the above is applied to the density of states instead of energy bands. Isayev et al. draw a material cartogram (a kind of map) using these two fingerprints, and attempted to predict the critical temperature of superconductors. They claimed that mapping using these fingerprints could be useful for material explorations. In addition, though not strongly asserted in the paper, fingerprints seem to work as descriptors for the prediction of material properties.

1.3.9 Summary of material descriptors

So far, we have reviewed several descriptors available in the field of materials science. Table 1-1 categorizes these descriptors based on several limitations in use that we pay attention in this study. Each descriptor has its own characteristics. Some descriptors are superior in specific situations, but difficult to apply to other situations. Other descriptors are generally applicable, but the performance may not be very good.

As previously explained, we would like to develop a novel descriptor that can be made from only the compositional information of elements or raw materials of which a material consists. We also want to make the novel descriptor applicable to any type of material. Such a new

descriptor can be useful in various situations. Hence, this study intends to discuss the methods of maximizing the use of compositional information. At the same time, we aim to make it flexible and expandable so that it can be customized depending on the materials and research topics.

Descriptor name	Can descriptor be used without three-dimensional structural information (crystal structure, atomic configurations, etc.)?	Can descriptor be used for a wide range of materials?	Can a pretrained model be applicable to a wide range of materials?
Atom-centered symmetry functions	X	O	X
Smooth Overlap of Atomic Positions	X	O	X
Crystal graph	X	O	O
Pure elemental property	O	X	X
Non-linear combinations of the primary features	O	△*	X
Molar concentrations	O	O	O
Elemental attributes	O	O	O
B-fingerprint & D-fingerprint	X	O	O

Table 1-1. The classification of the eight descriptors introduced in the previous subchapters based on limitations in use. (* Only for up to binary systems)

1.4 Objectives of the present study

Up to now, we have reviewed how much machine learning related technologies are used in materials science and engineering research nowadays. In Subchapter 1.2, we reviewed how machine learning can be used in material research and development. In Subchapter 1.3, we investigated the descriptors for machine learning models, which is the subject of this study. As shown in Table 1-1, descriptors can be classified into two categories: descriptors that can be made only with compositional information and those that requires structural information such as atomic configuration. If the compositional descriptor and the structural descriptor can be used together, more accurate training may be possible based on more information.

However, in testing material properties through actual experiments, it is difficult to obtain information such as atomic structures. The measurements of material properties require synthesis of real materials with controlling the composition of the raw materials and experimental conditions. Furthermore, although accurate prediction may be facilitated by adopting more descriptors such as simple properties of a target candidate material as shown in Subchapter 1.3.2, use of such descriptors is unavoidably limit the usefulness of prediction itself because the material properties of a new material that has not been tested yet require many trials and errors of synthesis of real materials or many simulations.

In certain situations, the independent variables we can use are just chemical formulas or compositions of raw materials and experimental conditions. Therefore, in the present study, we investigate machine learning models that can predict the target material properties and search materials by taking advantage of such limited information as much as possible. To do so, we propose a novel descriptor, elemental fingerprints, that generates input features for machine learning. The elemental fingerprints are designed to be applied to various situations by combining the characteristics of existing material descriptors and thus are expected to contribute widely to various material research and development.

The following Chapter 2 explains neural networks that are used to predict and explore materials in the present study and adversarial training that is one of the techniques for a more robust neural network model. Chapter 3 describes the elemental fingerprints proposed in this study. Chapters 4 through 6 examine how the elemental fingerprints and neural network model can be

applied to various materials and situations and discuss their usefulness in real material research and development. Chapter 7 gives the conclusions.

Chapter 2 Artificial neural networks

Artificial neural network (ANN) refers to a numerical calculation model that simulates biological brain structures and working principles. In a neural network of a biological brain, countless neurons are connected like a complex net with synapses. In addition, our brain learns new information by changing connections between neurons according to external stimuli. Brain's capability to have various network structures depending on the connection strength between neurons is called brain plasticity. ANN began with a study to simulate the process of learning information and memory of the brain neural network. It mimics the learning of the brain with updating the connections between artificial neurons (nodes). Although there are no clear classification criteria, dozens of ANN models have been proposed at present based on the node structure and the way of network connection. Among them, we will discuss multilayer perceptron in Subchapters 2.1 to 2.3, which is the first deep neural network that opened a new era of deep learning. From here on, ANN refers to the multilayer perceptron unless otherwise noted. Subchapter 2.4 introduces the method to make a more robust and accurate ANN model and explores its potentials.

2.1 Neural network model as a universal approximator

It can be considered that the beginning of the artificial neural network is perceptron proposed by Rosenblatt in 1958 [47]. In a perceptron, like connections of neurons, the value of the output node is described by giving weights to the values of multiple input nodes. Here, the nodes are connected through an activation function, which is a non-linear function. The activation function mimics the neural transmission mechanism that does not respond to stimuli weaker than a certain strength. For example, output of the step function is 0 when the input is less than the threshold and 1 when larger than the threshold. However, unlike real neural transmission, it is not easy to obtain a clue

for updating the ANN if it uses a step function. Therefore, an activation function like the sigmoid function, which is continuous and differentiable, is used instead. Recently, an activation function that has a sharp point, like a rectified linear unit, is widely used because it has been revealed that no problem arises even when there are a finite number of non-differentiable points [48]. A perceptron that describes one output for n inputs can be expressed as follows:

$$y = \varphi(\sum_{i=1}^{n} w_i x_i + b) \tag{2-1}$$

Here, w_i is a weight for each input, b is a bias, and φ is an activation function. One weight is required for each neuron-neuron connection, and the bias is a parameter that shifts the sum of the weighted values. However, the perceptron cannot simulate a non-linear objective function. One typical example is the XOR problem shown in Table 2-1 with two independent variables (or inputs) x_1, x_2.

Suppose that in a single layer perceptron, the output is 0 when the sum of x_1 and x_2 is 0, and output is 1 when the sum is 1. Then, when the sum is 2 (where both x_1 and $x_2$is 1), there is no way of making the output 0 again. In this case, our perceptron can explain only three-quarters of the data. A solution to this problem is unexpectedly simple: a hidden state is added between the input node and the output node. Here, the concept of the layer is adopted to simplify the model. An input layer is a set of input nodes, and an output layer is a set of output nodes. Furthermore, suppose a layer is connected in the following way. A node inside a layer is connected to all nodes of the following layer, but the nodes within the layer are not connected with each other. This layer is called a fully connected layer. A layer that exists between the input layer and the output layer is called a hidden layer.

When a perceptron has more than one hidden layer, it is called a multilayer perceptron, and the XOR problem can be solved by using one hidden layer having two neurons [49]. Fig. 2-1 shows an example of a multilayer perceptron that can solve the XOR problem. For the activation function φ, a step function, which outputs 0 for an input of less than 0.5 and outputs 1 for an input of greater than or equal to 0.5, is used to simplify the model.

Inputs		Output
x_1	x_2	y
0	0	0
0	1	1
1	0	1
1	1	0

Table 2-1. Truth table of XOR problem

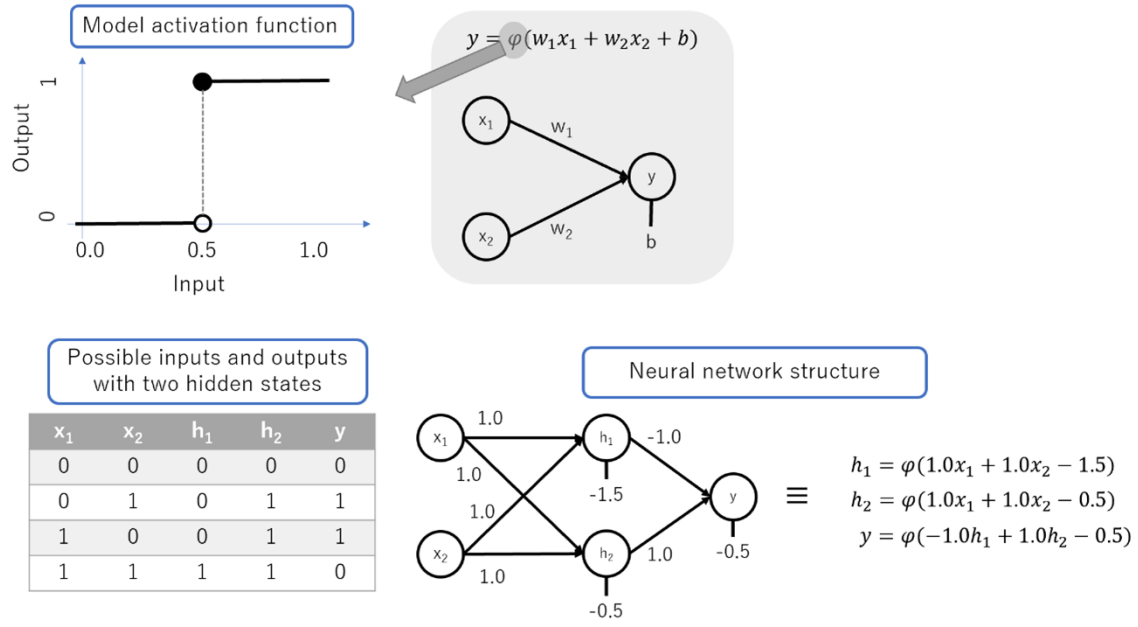


Figure 2-1. One possible network structure that can solve the XOR problem. h_1 and h_2 represent two hidden nodes. A Model activation function is undifferentiable for simplicity. Neural network structure with six weights and three biases also can be expressed by a set of three equations.

Furthermore, Cybenko proved that a multilayer perceptron using a finite number of neurons and the sigmoid activation function could approximate any type of continuous objective function in 1989 [50]. In 1991, it was shown that the above statement can be universalized to any type of activation function [51]. It is called the universal approximation theorem, and ANN is called a universal approximator. However, there are practically better networks than the multilayer perceptron depending on the respective characteristics of data. In many cases, it is beneficial to use two or more hidden layers though a single hidden layer is theoretically enough.

2.2 Backpropagation and optimization algorithms

Making the ANN model accurately describe a certain objective function is equivalent to solving the optimization problem of weights and biases in the model. Here, the weights and biases are called parameters, in particular, trainable parameters to emphasize that they are the parameters to be optimized. However, the ANN model with nonlinear activation function has no solution that can be expressed by a finite series of analytical functions. The most widely used optimization method in such a situation is the error backpropagation algorithm developed by Rumelhart et al [52]. First, if an ANN model has not been trained at all, all parameters are randomly initialized before the training begins. Backpropagation is the name given to an algorithm that updates the trainable parameters in the opposite direction for the calculation of output from the inputs called forward propagation. In the backpropagation, we have to evaluate how accurately the ANN model predicts the objective function. For this purpose, a loss function is introduced which evaluates the overall performance of ANN by adding all the prediction errors for all training data. Mean squared error loss is one of the most widely used loss functions in numerical predictions. Mean squared error loss was also used to minimize prediction errors of material properties in this study. In this case, the loss function L for n inputs can be calculated as follows:

$$L = \frac{1}{n}\sum_{i=1}^{n}(f_i - y_i)^2 \tag{2-2}$$

On the other hand, when a calculation result of the ANN model is a probability such as classification problems, squared error loss cannot distinguish cases of slightly incorrect classification and totally incorrect classification. For example, in a case of classifying two classes A and B, if the probability of being class A is predicted as 60% while it is actually B, it should have a smaller loss function value than the prediction of the probability of class A being 90%. In such cases, for example, cross-entropy loss that evaluates the degree of overlap between true distribution and estimated distribution is used as the loss function instead of the mean squared error loss.

Many ANN models use differentiable function for nonlinear activation functions, while the activation functions with the non-differentiable points are increasingly used in recent years as mentioned in Subchapter 2.1. In the latter cases, functions that do not lose continuity are adopted, so it is possible to obtain a numerical differentiation instead of an analytic tangent line. Consequently, in both cases differentiation can be used to predict the change in the loss function when the parameter is slightly changed. The main strategy of error backpropagation is to update a parameter little by little in the direction of decreasing the loss function. Now, a problem of increasing the performance of the ANN model consisted of N trainable parameters is changed into a problem of searching global minimum from differentiable N-dimensional space.

Many of the optimization algorithms used in this situation are based on the gradient descent method. It is an iterative algorithm that determines the next point (N-dimensional parameters) by evaluating the gradient or slope of the loss function. After we update the loss function, we recalculate the gradient again. Here, if the change of parameters is too large, then the minimum point will be passed over. In this case, the update of parameters is too hasty, which is expressed as “the training rate is too high.” If the change is too small (i.e., if the training rate is too low), then it will take too much timestep to reach the minimum point.

One of the simplest approaches of this method is the steepest descent algorithm that updates parameters toward a direction of the steepest gradient of loss. However, this algorithm is known to have a problem of being stuck at a local minimum or unable to escape from a saddle point. Some studies proposed a method called stochastic approximation [53,54] that assigns random fluctuation and inertia to the motion of an observation point by reflecting the previous update to the next update. Various algorithms adopting these techniques in a different way, often together. At present, one of the widely used optimization algorithm showing excellent

performance in various situations is Adaptive Moment Estimation (Adam) [55]. Adam is an algorithm calculates two types of moments to the motion of an observation point, and the training rate is self-regulated according to moments (this behavior is called adaptive). This study also used Adam as an optimizer.

2.3 Overfitting and regularizations

The optimization algorithms explained in the previous subchapters have all evolved so as to reduce the prediction loss as quickly as possible. Loss is a measure of the accuracy of training in machine learning. If the prediction of a machine learning model does not clearly describe the actual data because the loss is too large, it is called underfitting. However, paradoxically, we should also take note of a phenomenon called overfitting that can occur when the loss is too low [56].

Figure 2-2 explains the underfitting and overfitting by assuming that polynomial regression is performed on the simulated data set. Here, we have a single input variable with a single output variable, and there is some noise in the data. The noise here is likely to occur for the data set in real situations due to observation errors or features that cannot be explained by the current model. Figure 2-2 (a) shows a case where the model is so simple that the effect of the input variable is not trained enough; in other words, underfitting. In contrast to this, Figure 2-2 (c) and (d) show very low loss by increasing the degree of a polynomial in order to enhance the accuracy of the model. However, the models in these cases focus too much on each data point, and do not reflect the overall trend. Thus, the prediction values of the models greatly fluctuate up and down. Therefore, it is considered that overfitting is obviously occurring in the models.

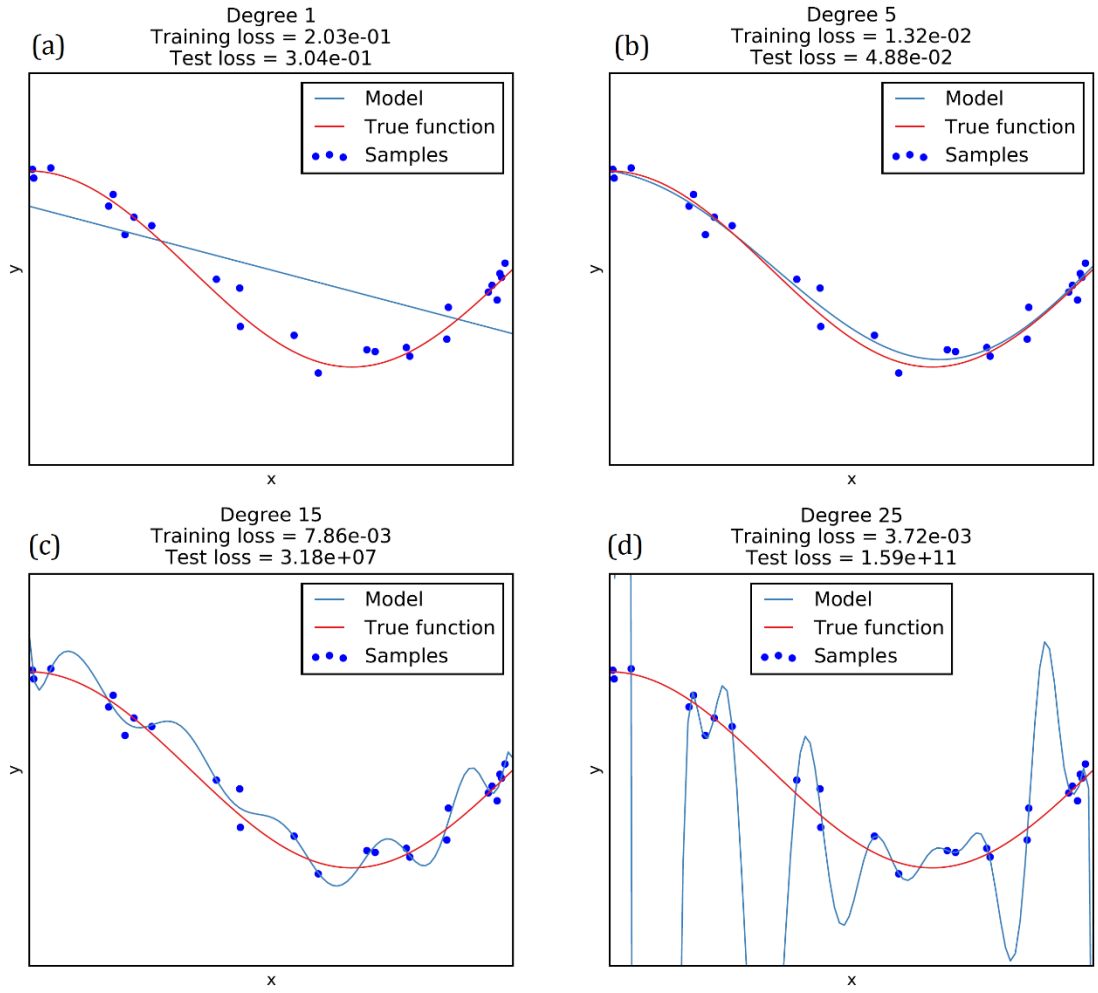


Figure 2-2. Demonstration of polynomial regression of different degrees. Training data is 1-D model situation with a random noise.

2.3.1 Training, validation, and test

One of the methods that quantitatively monitor whether overfitting has occurred in the machine learning model is to divide the data set into a training set and the test set. For example, if there are 100 data points in the database, we can use 80 of them to train a machine learning model. After model training, we calculate both the training loss and test loss. Here, the latter is evaluated by predicting the remaining 20 data points using the trained model. As seen in Figure 2-2(c) and (d), the training loss is relatively low, but the test loss is very high, which can be considered as the evidence of overfitting.

Then, how can we prevent overfitting? A straightforward way that can be applied to the existing prediction model without model modification is early stopping. In this method, we continuously monitor the possibility of overfitting during the training of the model and stops training in the middle if overfitting occurs. To this end, we have to prepare a validation set separately that is not directly used for training but is referred to as creating the optimal model. When the model is updated through the optimization algorithms previously discussed in Subchapter 2.2, both the training loss and validation loss can be simultaneously evaluated. Only the training loss will be optimized for parameter updates. The validation loss will also be reduced in the beginning as seen in the progress of model from Figure 2-2 (a) to (b). However, if the model is sufficiently trained, the validation loss can increase in the middle of the training. Because the loss tends to decrease with small oscillations by the nature of optimization algorithms, training should not stop when the validation loss increases only once. We can save the model in the middle, and if the increase of validation loss becomes statistically significant, then we retrieve the past model. However, there may be a concern that the verification results may vary depending on the data used for verification. Therefore, another method called cross-validation is also often used together [57]. It repeatedly validates the model several times to ensure that all data is included once in the validation set.

2.3.2 Regularization

There are several ways to prevent overfitting by using a technique called regularization, which adds a penalty term to the loss function. The basic concept of regularization is as follows.

The more the optimization algorithm updates the parameters inside the model to reduce the loss, the more the overfitting is likely to occur because parameters become increasingly complex. So, the penalty term is used to increase the loss again depending on the parameter complexity so that the loss does not become too small.

Two methods are mainly adopted for this [58,59]. One of them, L1 regularization, suppresses the complexity of the parameters by removing the less important parameters or input features. This is based on the idea that a simpler model is less vulnerable to overfitting. In the L1 regularization, a penalty is calculated from the absolute value of the weights. It makes a model sparse, that is, many of the weights in the model become zero. This can suppress overfitting by reducing unnecessary terms when there is a lot of redundant information or sparseness in the data. However, applying the L1 regularization to complex objective functions may not be suitable because there is a risk of missing important information.

On the other hand, the L2 penalty calculates a penalty on the sum of the square of all weights. This penalty makes sure that there are no unnecessary features in the model, yet the model does not rely too much on specific features. That is, it suppresses overfitting by forcing all weights to contribute to the model approximately to the same degree.

The following Equation (2-3) shows the application of the L1 regularization and L2 regularization to the mean squared error loss of a model consisting of m parameters θ, respectively.

$$L_1 = \frac{1}{n}\sum_{i=1}^{n}(f_i - y_i)^2 + \lambda\sum_{j=1}^{m}|\theta_j|$$
$$L_2 = \frac{1}{n}\sum_{i=1}^{n}(f_i - y_i)^2 + \lambda\sum_{j=1}^{m}{\theta_j}^2 \quad (2\text{-}3)$$

Here, λ is called penalty size or a regularization factor, and this value can be selected depending on the purpose of training, data size and characteristics. Figure 2-3 (a) shows how to select the optimal parameter when the penalty increases as the loss is reduced. Figure 2-3 (b) and (c) shows a schematic diagram of selecting the optimal parameters using the L1 regularization and L2

regularization, respectively, in a model situation where there are two parameters in the model. The weight of one parameter in the L1 regularization becomes zero, indicating that the model is simplified by one dimension from 2-D to 1-D.

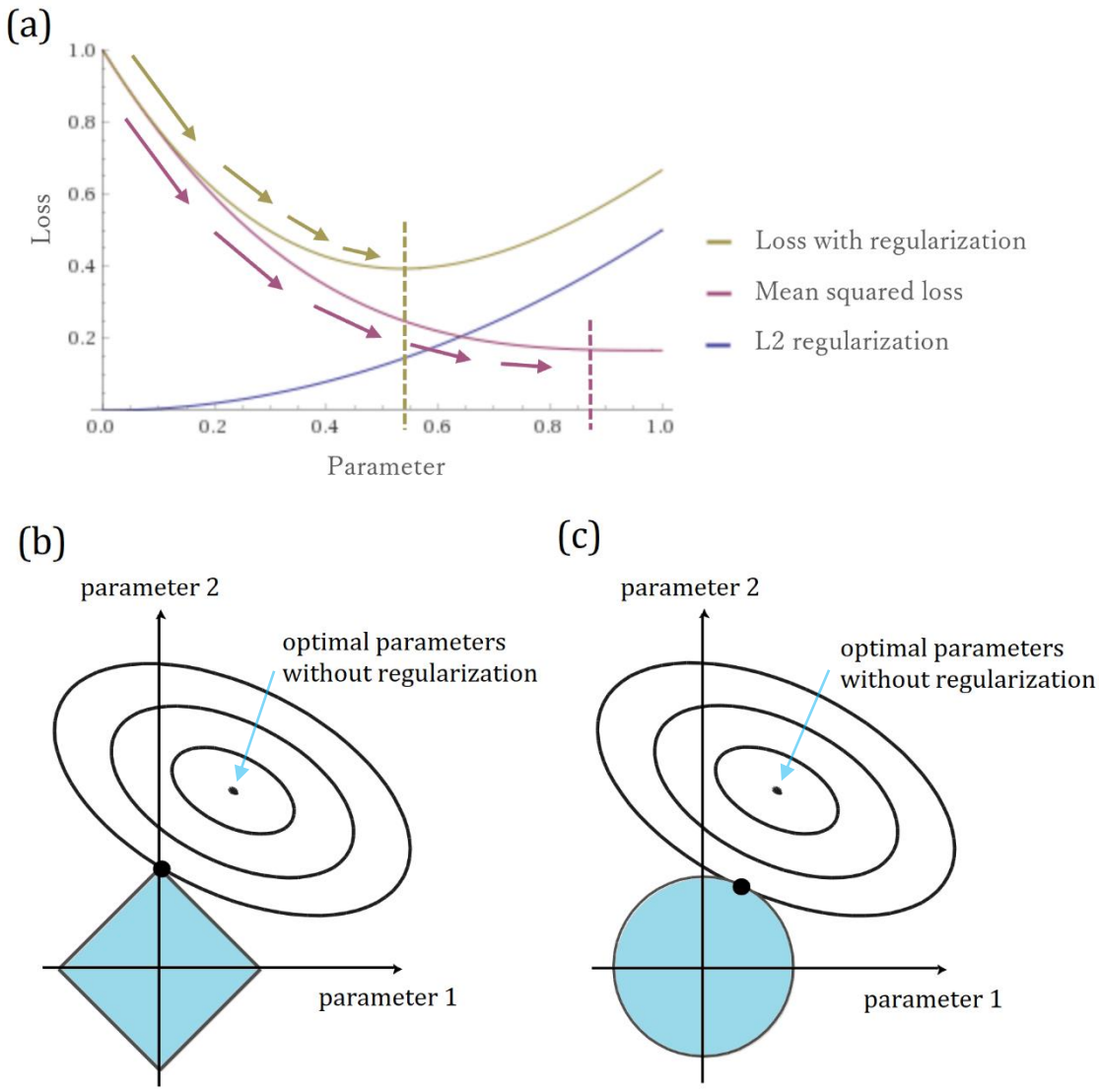


Figure 2-3. (a) Schematic diagram of parameter optimization without and with the regularization. The regularization prevents the parameter from becoming too large. Schematic representation of the mechanisms of (b) the L1 regularization and (c) the L2 regularization.

2.4 Adversarial training for robust model

We cannot guarantee that the model trained using the technique presented in Subchapter 2.3 is also valid for untrained data. As a well-known example of image recognition, Google has warned that even examples that differ slightly from correctly classified examples can be misclassified [60]. Eykholt et al. demonstrated that by adding a physical perturbation such as putting stickers on a stop sign, the stop sign could be recognized as a Speed Limit 45 sign [61]. In this way, somebody could deliberately attack traffic signs so that the deep learning model used for autonomous driving is forced to misclassify these signs. Many researchers now agree that we should prepare for these malicious attacks. These attacks are designed to find a combination of noises to maximize the probability that the classification model misclassifies input data by adding noise to original data. The data created in this way is called an adversarial example.

Therefore, adversarial training was proposed to train both original data and adversarial examples in order to train a model that can defend against arbitrary adversarial attacks. Although adversarially trained models are not perfect, they are known to be more accurate in the prediction of untrained data and more robust to adversarial attacks.

However, it required a lot of computational costs to create adversarial examples during model training. Goodfellow et al. proposed the fast gradient sign method (FGSM) to create adversarial examples quickly [60]. They noticed that the gradients should be calculated when training a model with backpropagation. Originally, gradients are used to update parameters to minimize loss, but inversely, they used gradients to add the component in the direction of maximizing loss to the input x. They created adversarial examples by giving the sign of gradients in the direction of increasing loss and proved that FGSM could be used to attack models quickly but effectively. As an example, they showed that the image of a panda could be misclassified as a gibbon (long-armed ape).

2.4.1 Adversarial training for better prediction performance

Adversarial training is actively discussed in the field of image recognition, but theoretically, it can be applied to numerical prediction models in the same way without any modification. Lakshminarayanan et al. noticed that the adversarial training of prediction models enabled smooth and robust predictive estimates [62]. This is because the training of adversarial examples forces the output of prediction models not to change drastically for small changes in input data. While regularization introduced in Subchapter 2.3 avoids overfitting by suppressing model parameters to prevent sudden fluctuations in predicted values, adversarial training achieves a similar effect by training with adversarial examples.

Another technique used to achieve better results than a single model is ensemble, that is, the use of multiple models. Since an ensemble of completely identical prediction models is meaningless, each model in the ensemble needs to be different from each other to some degree. This can be achieved by the adversarial training, but also by other techniques. Monte Carlo dropout, proposed by Gal & Ghahramani in 2016, is an effective strategy for the ensemble of ANN models [63]. Dropout refers to a method of constructing a network without using some neurons chosen randomly (as many as the number specified by the hyperparameter called dropout rate) in each training cycle of the ANN model. A typical dropout neural network computes the output values using all neurons in the inference process (after training is completed). In Monte Carlo dropout, an ensemble is composed of networks obtained by applying the dropout technique to the trained model even in the inference process.

In several demonstrations, however, Lakshminarayanan et al. showed that neural network ensembles created by adversarial training outperformed the ensembles of networks by standard training and ensembles of Monte Carlo dropout.

2.4.2 Uncertainty of neural network and Bayesian optimization

Lakshminarayanan et al. also claimed that adversarial training could make the evaluation of the predictive uncertainty from neural network models possible. This evaluation is one of the yet unsolved problems [64,65]. In their demonstration using a standard training technique of neural

networks, prediction of training data gives high accuracy, while the prediction of unseen examples gives consistently incorrect results. On the other hand, neural network models trained through adversarial training gives dispersed prediction results for unseen examples. Based on this demonstration, they claimed that the ensemble of neural networks could achieve higher prediction accuracy than single network models due to the distributed prediction results. Simultaneously, prediction uncertainty can be evaluated from the variance of the prediction results.

The prediction uncertainty of neural networks cannot be accurately evaluated. However, by an ensemble of M networks, the predicted value (average) and its uncertainty (variance) can be approximated as follows.

$$\begin{aligned} \mu_*(x) &= \frac{1}{M}\sum\nolimits_m \mu_{\theta_m}(x) \\ \sigma_*^2(x) &= \frac{1}{M}\sum\nolimits_m \left(\sigma_{\theta_m}^2(x) + \mu_{\theta_m}^2(x)\right) - \mu_*(x) \end{aligned} \tag{2-4}$$

There is a prediction model that can quantitatively evaluate the uncertainty of a prediction: Gaussian process. Bayesian optimization can be performed by using the Gaussian process because the Gaussian process can represent the Bayesian posterior distribution. Bayesian optimization is an algorithm that efficiently explores the optimal case among various possibilities, but it could not be performed with the neural network model [31,32]. In this situation, Lakshminarayanan et al. argued that the neural network ensemble created by adversarial training could simulate the Bayesian posterior distribution.

In their paper, however, it was not clear that the neural network ensemble could actually perform Bayesian optimization. We think that this approach can be very useful in material discovery based on the prediction of material properties, which usually have only a small number of training data with a high cost of additional observation. Therefore, in a case study in Chapter 5, we examine how effectively Bayesian optimization using the neural network ensemble can perform material discovery.

Chapter 3 Elemental fingerprints as novel materials descriptors

It is necessary to choose descriptors, the ways to convert the material information into the numerical input features, to predict material properties using machine learning techniques. In this study, we focused on descriptors made from the chemical formula or the compositions of raw materials, and applicable to the prediction of material properties. These descriptors are called compositional descriptors.

Machine learning with compositional descriptors has the advantage of being widely applicable only with compositional information. Descriptors that can be used with limited information can be applied to various researches without prior knowledge of the material. Compositional descriptors have been widely used because they are intuitive and easy to understand. However, they have not been improved much from the mole fraction or raw materials ratio described in Subchapter 1.3. To maximize the information in the compositional descriptor, we try to develop a novel descriptor containing more information than the other descriptors. The elemental fingerprints proposed in this dissertation assume that the effect of the compositions of the material on the experimental results can be explained by the combination of the primary features of the raw materials or elements.

In Subchapter 3.1, we propose a new descriptor named ‘elemental fingerprints.’ Elemental fingerprints can be used in general situations, not in a limited family, and can be used with other descriptors if there is additional information available. We will also give a detailed explanation with a real example of how to generate elemental fingerprints. Subchapter 3.2 introduces the expected merits that can be obtained by using elemental fingerprints and how they have been validated in subsequent Chapters 4,5, and 6.

3.1 Elemental fingerprints from elemental property distributions

In this dissertation, we discuss a descriptor that can utilize compositional information more effectively, one of the most basic independent variables in material research and development. In Subchapter 1.3, we examined various compositional descriptors already in use. However, there are cases where it is difficult to apply them to various types of materials (1.3.4 & 1.3.5), just they use raw information as it is (1.3.6) or their performance is not as good as intended (1.3.7).

Considering the situation described in Subchapter 1.3, we propose a novel material descriptor named elemental fingerprints. In developing this new descriptor, we were inspired by two recently proposed descriptors. One is the elemental attributes set by Ward et al. [45] explained in Subchapter 1.3.7. We develop a new descriptor based on how the elemental attributes use the properties of the elements that make up the material. The other is the band-structure fingerprint (B-fingerprint) and density-of-states fingerprint (D-fingerprint) [46] explained in Subchapter 1.3.8. The reason why the new descriptor is named elemental "fingerprint" is that the process of generating elemental fingerprints is inspired by B- and D-fingerprints.

Now, in terms of our elemental fingerprints, we first collect the elemental properties of the pure element or the physical/chemical properties of the raw materials according to target material properties. Unlike the B-fingerprint, in which the high symmetry points are predetermined according to the crystal structure, the number of elemental properties can be flexibly selected according to the purpose of the study. Figure 3-1 shows an example of how to generate elemental fingerprints of ionic radius and melting point for $WMgO_3$.

In order to make elemental fingerprints, primary features of raw material or pure elements are required. In this process, researchers themselves gather and select which primary features are used to make the fingerprints according to the purpose of the study and the type of materials. For example, in the case of mixing oxides to make glass, we can use information about the raw oxides and their cations (Chapter 4). It is also possible to use pure elemental properties for materials composed of various elements (Chapters 5 and 6). Figure 3-1 shows an example of creating elemental fingerprints using the two primary features of the covalent radius and melting point of pure elements.

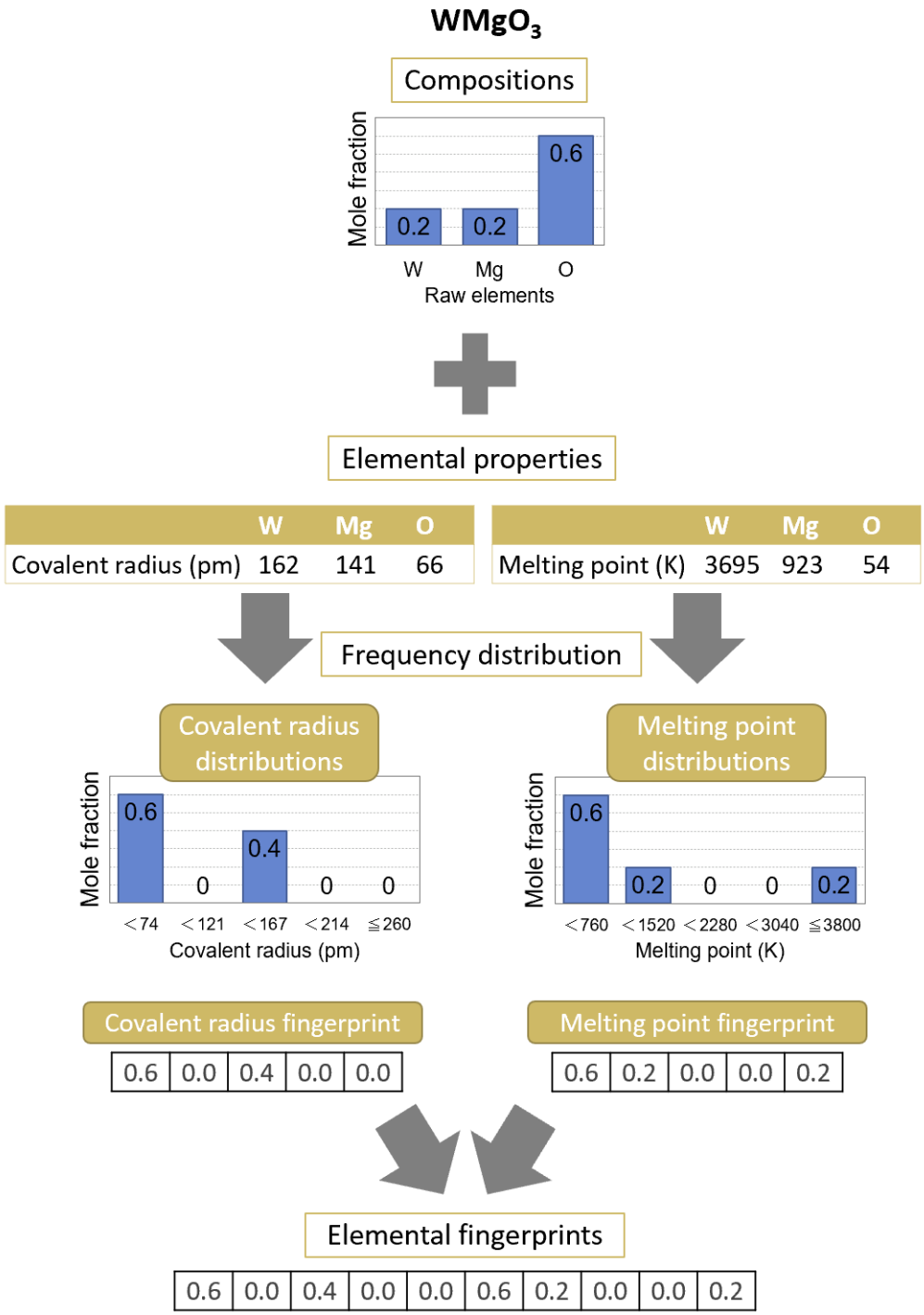


Figure 3-1. Schematic diagram of the generation of elemental fingerprints with two elemental properties.

At the top of Figure 3-1, we assumed that the target material is $WMgO_3$. From the chemical formula of this material, it can be seen that the elemental ratio of each element is 0.2 for W, and 0.2 for Mg, and 0.6 for O, respectively. In the case of synthesizing material from raw materials with some recipe, a mole fraction also can be calculated from the amount of the raw materials. In this example with $WMgO_3$, it is not easy to know the melting point of the target material. However, the melting points of the three elements that make up this material are widely known. Alternatively, even if other primary features are used, obtaining their information will be much easier than obtaining information on the material to be tested. The second row in Figure 3-1 shows the data on the melting point and covalent radius of three elements of W, Mg, and O.

The left-hand side of the third row shows the frequency distribution with five bins with information on the melting point. Oxygen has the lowest melting point and is located on the lowest side of the frequency distribution. Therefore, in the frequency distribution of the melting point, the first column contains the mole fraction of oxygen, 0.6. The melting point of Magnesium is in the middle and corresponds to the second column of the frequency distribution. The second column contains 0.2, which is the mole fraction of magnesium. Since no material has a melting point corresponding to the 3rd and 4th columns in the frequency distribution, the 3rd and 4th columns are zero. Finally, the highest melting point, that of tungsten, is located in the 5th column of the frequency distribution, and so this column has a mole fraction of tungsten, 0.2. As described above, the frequency distribution of the melting point is created. This process is repeated as many times as the number of primary features. At this process, two or more types of raw materials or elements may be located within the same frequency distribution class. For example, the covalent radius of tungsten and magnesium is not so different, so both elements are located in the 3rd column of the frequency distribution of the covalent radius. Therefore, the value of the third column of the frequency distribution, 0.4, is the sum of the molar fractions of tungsten and magnesium.

Elemental fingerprints are the descriptor for predicting or explaining certain experiments through the distribution of primary features of raw materials or pure elements. In the above example, elemental fingerprints assume the effect of the covalent radii of tungsten and magnesium on the experimental results is the same. However, by using various types of primary features, each raw material or element can be uniquely identified. Simultaneously, the effect of variation of the primary features on the results can be evaluated. Elemental fingerprints are

obtained by connecting the frequency distributions constructed as the above into a one-dimensional array. Therefore, the length of the entire fingerprints is the product of the number of primary features and the number of bins of the frequency distribution.

3.2 Expected merits of elemental fingerprints

The elemental fingerprints proposed in this dissertation assume that the effect of the composition of the material on experimental or simulation results can be explained by combining distributions of the primary features of the raw materials or elements. One of the consequences of this assumption is that even when predicting or evaluating a material containing an entirely new raw material, its properties can be predicted rather well through the distribution of primary features of raw materials. This is because elemental fingerprints do not directly use the compositional information, but convert it using primary features.

It is known that accurate prediction of untrained areas by a machine learning model is difficult. To investigate the first expected merit of fingerprints, we examine whether fingerprints can give a better prediction for data deviating from training data than other descriptors. In Chapter 4, we test how well prediction models trained on a limited set of data can predict the effects of raw materials that are never seen before.

The second consequence of the assumption used in the elemental fingerprints is that the researcher can create a descriptor optimized for respective cases by controlling the primary features of the fingerprints according to the research purpose or type of material. This point will also be demonstrated in this dissertation.

Chapter 4　Case study on oxide glass materials

In this chapter, we inspect the effectiveness and performance of elemental fingerprints for the viscosity (η) of oxide glass materials. A glass is an amorphous solid-state material obtained from rapid cooling from the high-temperature liquid state to a temperature below the glass transition temperature (T_g). The rapid cooling suppresses crystallization and causes a supercooled liquid state. The supercooled liquid becomes more viscous as the temperature is lowered. Understanding the behavior of its viscosity is important in the case of glass materials, and many studies have been conducted to obtain this [66-71].

First, a brief introduction about the viscosity of glass materials and previous studies on its behavior are discussed in Subchapters 4.1 to 4.3. Subchapter 4.4 examines the optimum settings and prediction performance of descriptors and prediction models. We tested prediction performance using not only fingerprints but also elemental attributes and molar fraction. We also compared five different prediction models, including ANN that are widely used for machine learning to select the best prediction model. In Subchapter 4.5, we experimented with extreme data separation, assuming a novel material search. That is, while the test dataset for investigation of the prediction performance is usually randomly divided from the entire dataset, in this subchapter, only data that do not contain a particular oxide are used for training, and data containing the oxide is used for testing. This is intended to verify whether fingerprints can predict properties of materials that have not yet been trained.

In this chapter, we call elemental fingerprints “cation fingerprints” because we use the physical properties of oxides and their cations as primary features for generating fingerprints.

4.1 Viscosity of glass materials

Viscosity is a quantity expressing the resistance to deformation of fluid. The fluid becomes more viscous as the temperature is lowered. When its viscosity becomes sufficiently high, materials may never crystalize because the higher viscosity suppresses the movement of atoms necessary for nucleation and growth. Therefore, the understanding of viscosity behavior is significant in the understanding of the mechanism of glass formation.

One of the important aspects of viscosity in the behavior of glass is its temperature dependence. To examine this behavior, the Angell plot is often used [69], in which the logarithm of η versus the inverse of temperature T multiplied by the glass transition temperature (T_g/T) is plotted. The Angell plots for various glass materials can be classified into three types, namely the 'strong,' 'fragile,' and 'intermediate' plots. The strong plot is almost a straight line, whereas the fragile plot is a convex curve. The intermediate type is in the middle of the strong and fragile plots. In the cases of the strong and intermediate types, which include most of the geochemical and metallic glasses, the Angell plot can generally be described by the following Vogel-Fulcher-Tammann (VFT) equation [71-73]:

$$\eta = \eta_0 \exp\left[\frac{DT_0}{T - T_0}\right] \tag{4-1}$$

Here, D is called the liquid's fragility, η_0 is the pre-exponential term corresponding to the viscosity at the infinite temperature, and T_0 is defined as the temperature at which the entropy of the liquid phase and that of the solid phase are identical [71,74]; that is, an ideal glass transition occurs. It is theoretically expected and experimentally confirmed that T_0 is somewhat lower than T_g. For the strongly fragile cases, such as fluoride glass, the Cohen-Grest (CG) model provides a better simulation [70,75,76].

The compositional dependence is another factor to be considered. Many theories used to explain this dependence of glass properties are based on the additive models of Winkelmann and Schott [25,77]. This method assumes that a glass is an ideal mixture of components. Then, each property of glass, e.g., heat capacity and T_g, is expressed as a linear function of the compositional ratio.

For these various analyses to describe viscosity, it is more effective to compare temperatures with specific viscosity rather than to compare viscosity at specific temperatures. Such a temperature is called isokom temperature. There are several isocom temperatures which especially attract attention. For example, the isokom temperature at $10^{12.0}$ Pa·s is called the annealing point, which refers to the temperature to relieve internal residual stresses of glass without shape change. The isokom temperature at $10^{6.6}$ Pa·s is called the Littleton softening point, which refers to the temperature at which glass deforms under its own weight. The isokom temperature at $10^{1.5}$ Pa·s is called the melting point, at which glass is fluid enough to be considered liquid. These three points are the isokom temperatures that are usually measured to describe the viscosity behavior of glass formation.

4.2 Existing models to predict the compositional dependence of viscosity

Many attempts have been made to explain the correlation among viscosity, temperature, and composition in a specific family of glass, such as soda-lime glass. For example, Fluegel proposed equations in terms of the mole fraction of raw materials and their products [26]. Note that the inclusion of the product of the mole fractions enables us to describe the nonlinearity not included in the Schott model within the multiple linear regression method. Specifically, Fluegel derived three equations to predict the three isokom temperatures mentioned in Subchapter 4.1. Note that it is more convenient to predict the isokom temperature than the viscosity at a certain temperature because the temperature for a particular viscosity can vary over a wide range, depending on the chemical composition. One of the Fluegel equations is presented here,

$$\begin{aligned} T_{6.6}(\mathrm{°C}) = 939 &+ 5.81\, C_{\mathrm{Al_2O_3}} - 4.37\, C_{\mathrm{B_2O_3}} - 0.174(\, C_{\mathrm{B_2O_3}})^2 + \cdots \\ &+ 0.320\, C_{\mathrm{B_2O_3}} \times C_{\mathrm{Na_2O}} + \cdots \\ &- 0.034\, C_{\mathrm{Al_2O_3}} \times C_{\mathrm{B_2O_3}} \times C_{\mathrm{Na_2O}} + \cdots \end{aligned} \tag{4-2}$$

Here, $C_{Al_2O_3}$ represents the mole fraction (mol%) of Al_2O_3. This equation consists of 16 terms representing mole fractions of raw materials (excluding SiO_2 as the remainder) and 22 terms representing the square and cubic terms, and the terms representing the products of the mole fractions of two and three raw materials. As the term selection and parameter optimization of the equation represent a type of machine learning, the Fluegel equation is valid only within a range of raw materials and their concentrations in which a sufficient number of data are available.

4.3 Glass materials databases

In the present study, our training target is the isokom temperature of glass at three viscosity points of $10^{1.5}$, $10^{6.6}$, and $10^{12.0}$ Pa·s. Two databases are suitable for the purpose of training the glass materials. One is SciGlass [78], which was used to select terms and optimize parameters in the Fluegel equations mentioned earlier. The other database is INTERGLAD [79], which was released earlier than SciGlass. It includes a regression analysis and glass design functions. Priven et al. claim that INTERGLAD is more suitable for commercial glass development owing to the 800,000 data records available on a variety of properties while SciGlass contains more detailed descriptions on synthesis procedures and original reference information [80]. However, both databases have recently been improved and there is no significant difference between them. Both have data about the material properties of more than 300,000 types of glass materials and have similar characteristics.

Note that the quality of data always presents an issue. Fluegel was conscious of this and used only calibrated data within a limited composition range to construct the Fluegel equation. An offset value was set for each reference (i.e., the data in the same reference have the same offset value), such that the same calibrated value would be obtained for the same compositions and conditions. This preprocessing increases the reliability of the model and eliminates the errors depending on experimenters. On the other hand, this approach cannot be applied to references that do not have enough data for mutual verification. Consequently, data of such references are discarded.

Because the purpose of this study is not the construction of a predictor with high accuracy but rather the verification of our novel descriptor, we used INTERGLAD without any preprocessing treatment or screening of data. In this study, data of raw oxide materials are also necessary to generate cation fingerprints. We collected data of seven properties (density, formation enthalpy, formation entropy, melting point, number of coordination, ion radius, and electronegativity) for 66 oxides and their cations from the materials handbook [81]. The number of isokom temperature data records for glass materials made from the combination of these 66 oxides is 1541, 12103, and 5636 at the three viscosity points, respectively.

4.4 Performance dependence on hyperparameters for fingerprints

Before performing machine learning, it is necessary to set hyperparameters. That is, in the case of ANN models, we must determine the numbers of neurons and hidden layers, the type of activation function, etc. In addition, in the case of machine learning via fingerprints, we have extra hyperparameters for the generation of fingerprints, that is, the choice of physical properties of raw materials and the number of bins (or the resolution of frequency distribution). These two hyperparameters are unique and critical only for the fingerprints. Thus, we examined the dependence of the machine learning results on the two hyperparameters and set their values. Table 4-1 shows the optimized results of the search space that was considered to find the best prediction model. We are using all available data from our database that mentioned at the end of Subchapter 4.3 for this optimization process.

Hyperparameter	Search space	Optimized value
Prediction model	Neural network, random forest, support vector machine, kernel ridge regression, linear ridge regression	Neural network
Activation function	Sigmoid, Tanh, ReLU	ReLU
Number of hidden layers	1 to 6	2
Number of neurons in each hidden layer	70, 100, 140, 200	140
L2 regularization scale	0, 10^{-5} to 10^{-2}	0.0001
Number of physical properties	1 to 7	7
Number of bins	5, 8, 10, 15, 18, 20, 24, 30, 50, 100	20

Table 4-1. Hyperparameter tuning results.

For the L2 regularization scale, the optimal value was 0.0001, which provided a prediction error of 33.0 °C. However, the improvement in the prediction accuracy was not significant compared to that of the 33.6 °C without regularization. This result suggested that overfitting is unlikely to occur because of the sufficient number of training data.

The hyperparameters are listed in the last two lines of Table 4-1. The numbers of the physical properties and the number of bins are hyperparameters specific to the cation fingerprints. We analyzed and examined the characteristics of the cation fingerprints in the following subchapters.

Note that aside from the ANN model, we also examined four other models, the random forest, support vector machine, kernel ridge regression, and linear ridge regression, for comparison. The optimization details of these four models are listed in the Table 4-2. In case all the data listed in the database are used, the well-tuned prediction models have small differences

in prediction errors (neural network: 33.0 °C, random forest: 33.1 °C, support vector regression: 33.9 °C, and kernel ridge regression: 34.2 °C) except for the linear ridge regression (76.1 °C). Note that the errors in this section are represented in terms of the root-mean-square (RMSE) of the randomly selected test sets.

Prediction models	Optimized hyperparameters	Root mean squared error (°C) of Isokom temperature prediction
Neural network	Number of hidden layers = 2 Number of neurons in each hidden layer = 140 Activation function = Rectified linear unit Regularization scale = 0.0001	32.95
Random forest	Number of trees = 100 Maximum number of features = 56 (40%) Maximum depth of tree = 50	31.40
Support vector regression	Kernel = Radial basis function Cost function = 2000 $\gamma = 5.0$	33.86
Kernel ridge regression	Kernel = Radial basis function Regularization scale = 0.0001 $\gamma = 1.0$	34.17
Linear ridge regression	Regularization scale = 0.0001	76.06

Table 4-2. Optimized hyperparameters and prediction accuracy of five regression models considered in this study. Training of prediction models other than the neural network was done by Scikit-learn [82].

4.4.1 Hyperparameters of fingerprints: number of bins

Figure 4-1 shows the dependence of the prediction performance on the number of bins for the isokom temperatures of $\log(\eta/(\mathrm{Pa}\times\mathrm{s}))=6.6$. Two parallel dashed lines represent the test set errors of prediction using molar ratio and elemental attributes as descriptors, respectively. In the test of the other two descriptors, the best ANN structures were the same as the ANN structure optimized by the cation fingerprints. The line graph shows the test set error as a function of the number of bins, when the cation fingerprints are used as a descriptor. As the number of bins increases, the resolution of distributions become finer and the test set error decreases initially and then increases. As seen in Fig. 4-1, 20 bins have the best prediction performance. Therefore, in this study, we decided to use 20 bins.

We speculated that in the limit of the number of bins, the prediction error of cation fingerprints converges to that of the molar ratio because of the following reason: Fingerprints are constructed by assigning a molar ratio of the raw material to each bin based on the distribution of the target raw property. In this process, as the number of bins increases, the average number of raw materials assigned to a bin decreases. Because the raw materials used in this study were 63 oxides, the average number of raw materials assigned to one bin was less than one in the cases of those that were larger than 63. As a result, the value of a bin is based on the molar ratio of one raw material, excluding the empty bins. Consequently, the prediction performance of the cation fingerprints and molar ratio was expected to be the same in the limit of the number of bins.

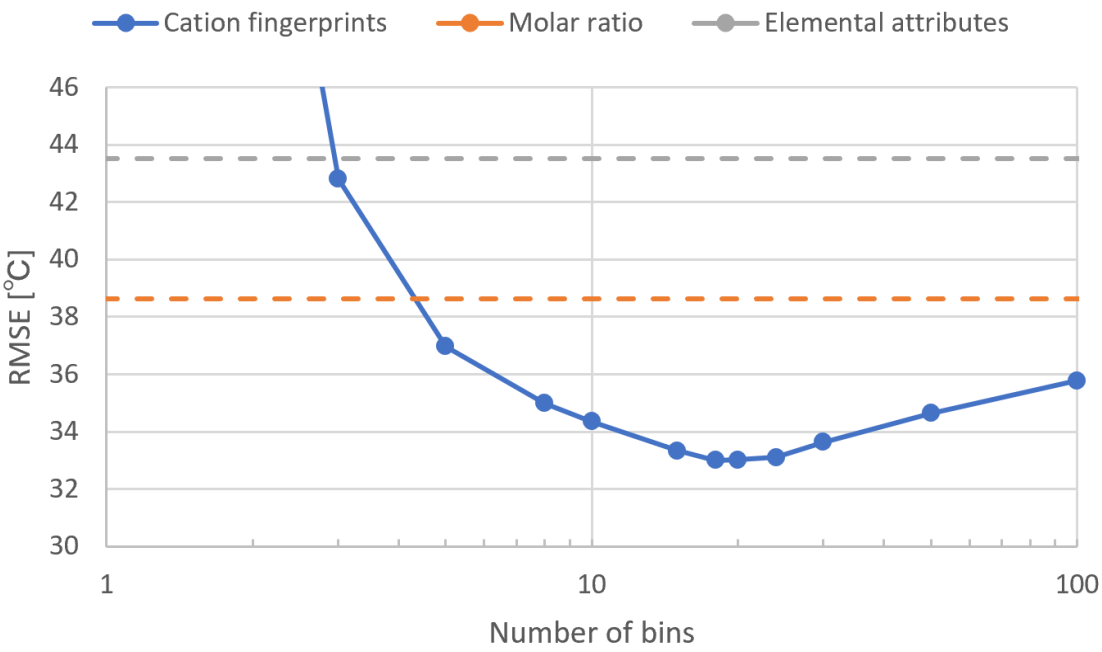


Figure 4-1. Prediction error of ANN models from different number of bins with seven physical properties. The two parallel dashed lines represent the prediction error from the elemental attributes and molar ratio, respectively.

4.4.2 Hyperparameters of fingerprints: selection of properties

For the number and selection of physical properties, the optimization on them can not only improve the training accuracy but also provide clues to determine which physical properties are important. While the ANN is one of the black box models, we can use the nature of fingerprints to indirectly evaluate the importance of each physical properties by controlling the properties used for machine learning. First, we tested seven sets of fingerprints in which each property was removed from the seven properties, respectively. No significant differences in the test error are seen among the seven sets.

Next, we examined the cases of one or two properties. The diagonal and off-diagonal terms in Table 4-3 show the test errors (RMSE) of the models with the one-property and two-property fingerprints, respectively. In Table 4-3, the seven physical properties are denoted by

capital letters, that is; A: electronegativity, B: coordination number, C: density, D: ionic radius, E: formation enthalpy, F: formation entropy, and G: melting point. Among the 28 combinations using the cation fingerprints in Table 4-3 (a), the combinations of the oxide density and melting points explain the viscosity behavior with the smallest prediction error of 41.6 °C. This is verifiable because a higher density or melting point of an oxide means that a higher temperature is required to achieve the same viscosity level. Using these analyses, the important physical properties explaining the isokom temperature can be estimated.

Figure 4-2 shows the dependence of the RMSE of the test prediction on the number of physical properties. The smallest and largest training errors within the same number of used properties, except for the case using all seven properties, are shown. As shown in the figure, the model becomes more stable and accurate when more physical properties are included. On the other hand, there was no significant decrease in the prediction performance despite some important properties missing. It can also be observed that there was no decrease in performance when relatively meaningless properties were added. Considering the above, we decided to generate cation fingerprints from frequency distributions of 20 bins for the seven physical properties of raw oxides.

We can also perform an evaluation similar to Table 4-3 (a) for the elemental attributes because the properties used to generate descriptors can be chosen. The results of the same analysis for the attributes are shown in Table 4-3 (b). Although the order of the RMSE among the 28 combinations was not exactly the same between Tables 4-3 (a) and (b), the isokom temperature was most accurately predicted when the oxide density or cation ionic radius was used with the oxide melting point in both cases.

However, a distinct difference was observed between the cation fingerprints and elemental attributes. In the prediction by cation fingerprints using only one physical property, the melting point (G) provided the lowest prediction error, whereas it provided the largest error in the prediction by elemental attributes using one physical property. This difference can be attributed to the fact that the fingerprint contains richer information than that of the set of statistical quantities used in the elemental attributes such as mean and variance.

In addition, in cases where two and seven properties were used, the prediction accuracy of the cation fingerprints was higher than that of the elemental attributes. From these results, it

can be concluded that the cation fingerprints were superior to the elemental attributes in both the prediction performance and explanation capabilities.

(a)

A	67.17						
B	54.41	143.63					
C	44.42	54.14	69.72				
D	43.20	45.58	45.02	74.99			
E	49.58	48.48	43.95	44.16	76.88		
F	49.45	67.30	42.56	50.16	59.29	106.02	
G	42.18	47.92	41.56	41.64	46.17	44.60	52.41
	A	B	C	D	E	F	G

(b)

A	156.44						
B	84.35	169.42					
C	78.66	69.52	110.12				
D	100.70	74.62	75.30	113.11			
E	83.84	74.09	85.68	79.41	180.71		
F	102.64	101.97	76.15	105.59	70.02	142.43	
G	75.31	86.71	70.62	68.87	75.37	78.51	180.71
	A	B	C	D	E	F	G

Table 4-3. Test set root mean squared error (°C) of ANN training from one (diagonal components) or two physical properties. Training was done by (a) cation fingerprints and (b) elemental attributes. Each physical property is indicated by one capital letter (A: Electronegativity, B: coordination number, C: density, D: ionic radius, E: formation enthalpy, F: formation entropy, and G: melting point).

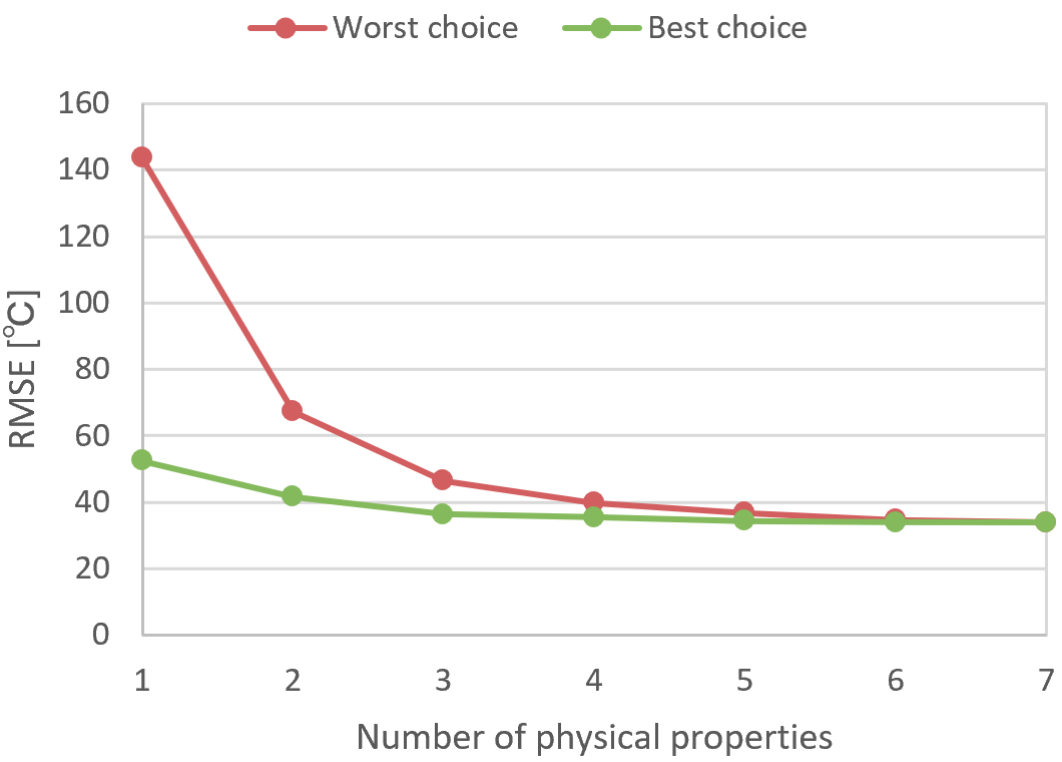


Figure 4-2. Test error of ANN training from fewer physical properties. Each line represents the worst and the best choices from each combination

4.5 Fingerprint as a descriptor for untrained novel materials

To demonstrate the advantage of the fingerprints, we tested a virtual extreme case of material search, which cannot be performed at all by the mole fractions. For this purpose, we set a specific target oxide in raw materials, and trained the ANN model using materials that do not contain the target oxide. Then, we tested the prediction accuracy of the glass materials, including the target oxide with the ANN model trained in this manner.

We searched the entire database for raw materials in which the percentage of glass containing those raw materials is between 10% and 30%. This range was determined based on the 20% random test set we examined earlier. This range allows us to conduct as many test sets as

possible while making it easier to compare results with the 20% random test set. In other words, we have to consider the trade-off between the small test set, which may not properly reflect raw material dependency, and the significant change in training size, which may influence ANN model performance. Table 4-4 shows the nine oxides that satisfy the above condition. In the case of SnO_2 for example, 1,307 glass materials contain SnO_2, but the remaining 10,796 do not contain SnO_2 at all. Then, the ANN model was trained by the 10,796 data points, and the isokom temperatures of 1,307 glass materials including SnO_2 were predicted.

Raw materials	MgO	Li_2O	ZrO_2	PbO	SrO	P_2O_5	TiO_2	Sb_2O_3	SnO_2
Test set size	3631	3044	2673	2593	2518	2036	1895	1775	1307
Training set size	8472	9059	9430	9510	9585	10067	10208	10328	10796

Table 4-4. Nine target materials with their respective test set size and training set size.

4.5.1 Hyperparameters for predicting untrained novel materials

The prediction error was large when we used the model settings discussed in Subchapter 4.4. Hence, we re-optimized the prediction model and investigated the requirements to explore novel combinations of materials which have not been examined yet. In Figure 4-3, we show the dependence of the averaged error over the nine test sets in Table 4-4 on the L2 regularization scale. The minimum prediction error is obtained when the regularization scale is 0.3. Although the final prediction error, 53 °C, is higher than the random prediction error of 39 °C, the result is meaningful because the material property is predicted from the training data without including the target raw materials. In addition, the prediction errors of elemental attributes and molar

concentrations under the same situation are 73 °C and 62 °C, respectively. That is, cation fingerprints show better performance than the other descriptors.

We attempted to determine in more detail how a prediction is possible in this extreme situation. As we can see from Figure 4-4, with the regularization scale, the training error continuously increases, whereas the test error decreases gradually until the regularization scale reaches 0.3, then slowly increases. This is typical behavior of the phenomena known as the tradeoff between overfitting and underfitting. Note that the value of the regularization scale of 0.3 at the test error minimum was higher than that of the optimized scale at 0.0001. This was conducted by randomly dividing the data sets. This may suggest that the present attempt is a situation where overfitting is likely to happen and is different from normal training and prediction. We also examined the effect of the number of bins and found that the optimal number of bins was 20, which was the same as in Subchapter 3.1 from the randomly divided data sets as shown in Figure 4-4.

In Table 4-5, we compared the performance of ANN with the other four prediction models as in Subchapter 3.1. Note that the hyperparameters of all models have changed to prevent overfitting. The results of the prediction errors for the random forest, support vector regression, kernel ridge regression, and linear ridge regression were 61.7 °C, 57.7 °C, 58.91 °C, and 77.0 °C, respectively. The prediction errors significantly increased in all models including the ANN model. Compared to the case in Subchapter 3.1, the difference in the predictive accuracy between the ANN and other models was slightly larger. In both cases, the ANN model had the best predictive performance. Therefore, it can be concluded that the ANN model fits well with the cation fingerprints for the prediction of material properties.

In Table 4-6, we show the dependence of prediction errors of respective target oxides and their average on the regularization scale. While some oxides (SnO_2, ZrO_2) are well predicted even without regularization, some others (P_2O_5, Li_2O) are poorly predicted, even with a large regularization scale. In particular, the error in the case of P_2O_5 is the largest. This large error may be attributed to the fact that phosphorous is a nonmetallic element and oxides of nonmetallic elements are rarely contained in this database, other than phosphorus pentoxide. Specifically, only 163 data records of SO_3 and 7 of SeO_2 are available as oxides of the nonmetallic elements.

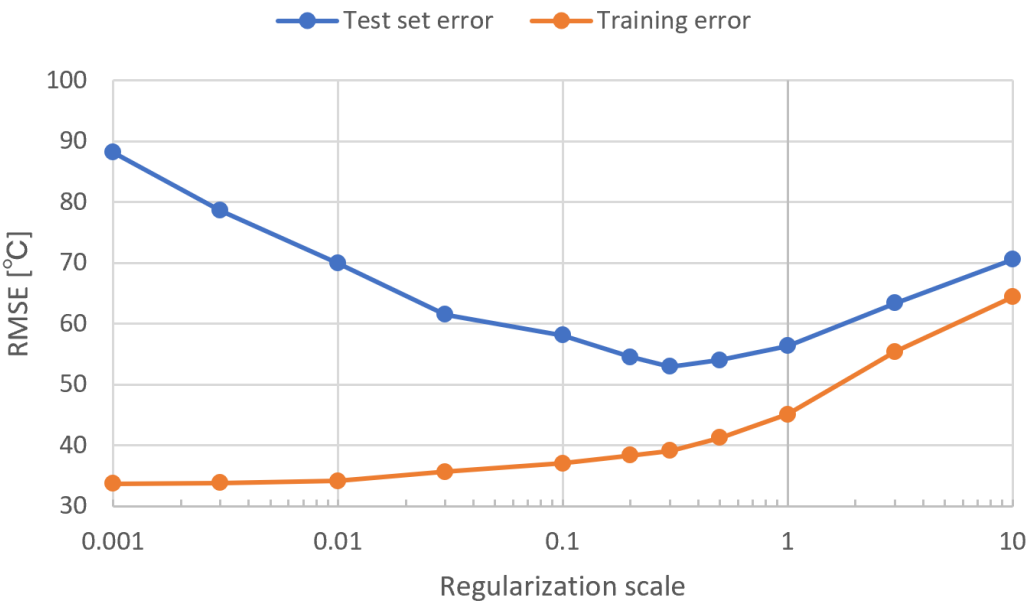


Figure 4-3. Penalty dependence of averaged root mean squared error of the nine test sets from Table 4-4.

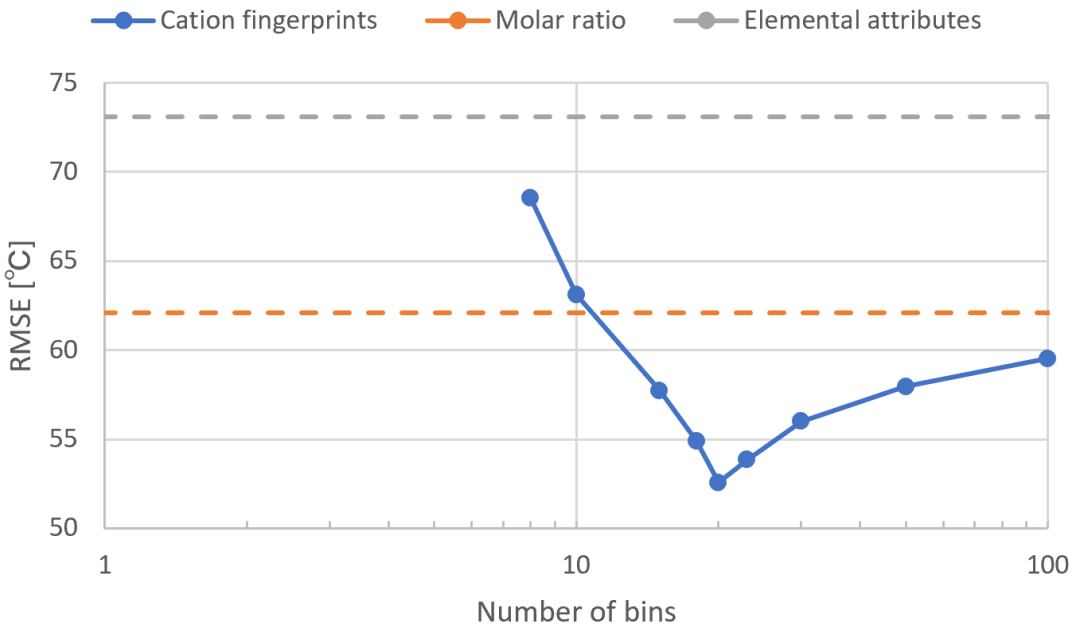


Figure 4-4. Dependence of prediction error of ANN models on the number of bins. Two parallel dashed lines represent the prediction errors from elemental attributes set and molar ratio, respectively.

Prediction models	Optimized hyperparameters	Root mean squared error (°C) of Isokom temperature prediction
Neural network	Number of hidden layers = 2 Number of neurons in each hidden layer = 140 Activation function = Rectified linear unit Regularization scale = 0.3	52.56
Random forest	Number of trees = 100 Maximum number of features = 28 (20%) Maximum depth of tree = 30	61.69
Support vector regression	Kernel = Radial basis function Cost function = 700 $\gamma = 0.5$	57.74
Kernel ridge regression	Kernel = Radial basis function Regularization scale = 0.2 $\gamma = 0.2$	58.91
Linear ridge regression	Regularization scale = 1.0	77.01

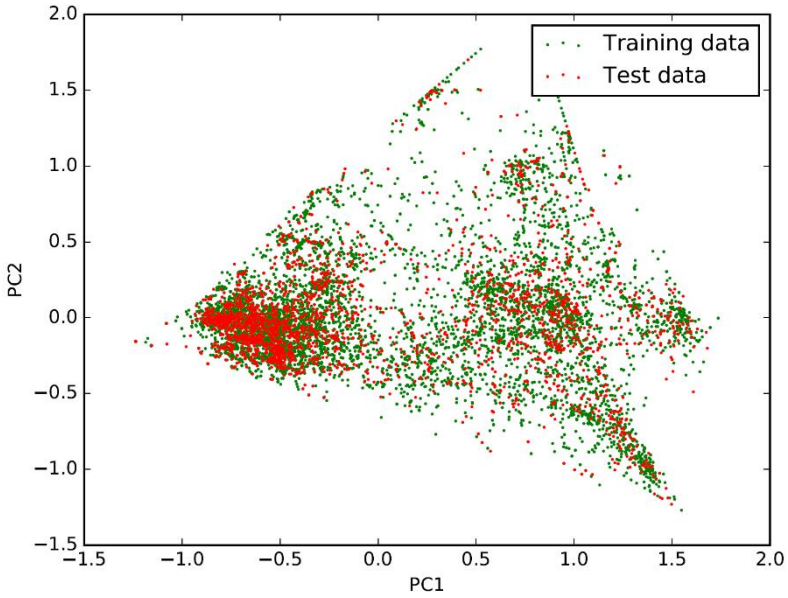
Table 4-5. Optimized hyperparameters and prediction accuracy of five regression models considered in this study. Training of the prediction models other than the neural network was done by Scikit-learn [82].

Raw materials	MgO	Li_2O	ZrO_2	PbO	SrO	P_2O_5	TiO_2	Sb_2O_3	SnO_2	Average
RMSE of test set (°C) with L2 = 0.0001	63.06	92.87	53.32	148.4	56.69	453.7	62.14	73.58	31.23	115.00
RMSE of test set (°C) with L2 = 0.001	57.92	80.36	52.67	129.3	55.13	271.9	52.94	62.45	30.56	88.13
RMSE of test set (°C) with L2 = 0.01	51.71	76.77	52.56	100.7	54.20	168.4	48.71	45.72	30.55	69.93
RMSE of test set (°C) with L2 = 0.3	48.68	64.50	48.61	60.89	54.81	81.91	47.34	39.74	30.27	52.97
RMSE of test set (°C) with L2 = 1.0	53.33	66.21	50.93	86.53	58.00	80.52	47.16	39.15	37.20	57.67

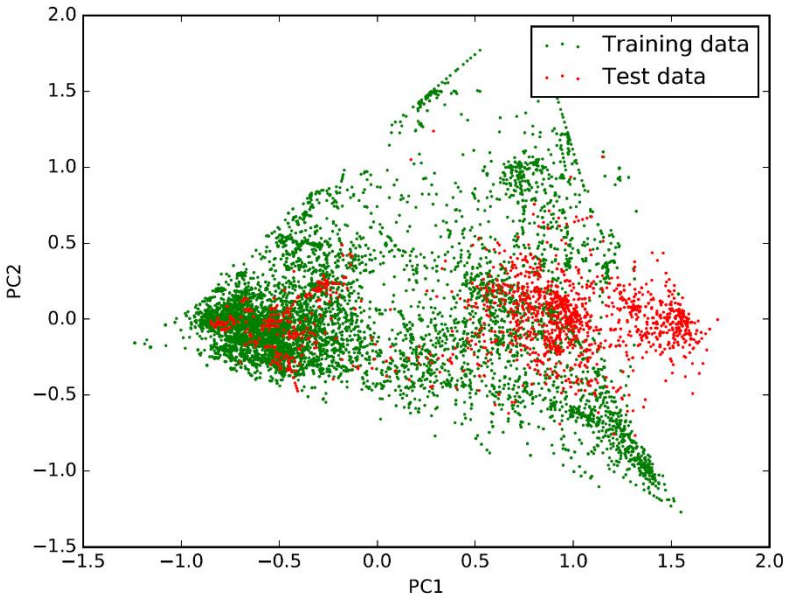
Table 4-6. Root mean squared error (RMSE) of test sets with various L2 regularization scales.

4.5.2 Principal component analysis of fingerprints

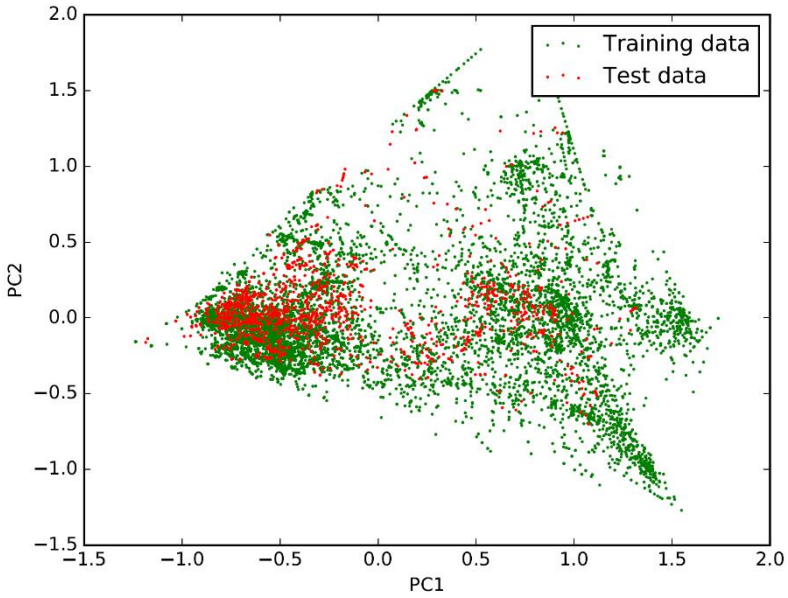
We performed principal component analysis (PCA) [83,84] to investigate the cause of large error in P_2O_5 and Li_2O. Figure 4-5 represents the distributions of the 1st and 2nd principal components in the case of (a) randomly divided test set, (b) P_2O_5 test set, (c) TiO_2 test set, and (d) Li_2O test set, respectively. As seen in Figure 4-5(b), in the case of the P_2O_5 test, many test materials (containing P_2O_5) are distributed in the right-hand area where no training data exist. However, in the case of Li_2O, which has the second largest prediction error, most of the test data are distributed in the region where the training data exist. A similar feature is seen in the TiO_2 case, which has moderate prediction error. This PCA analysis reveals why the prediction error of a test set is large in the case of P_2O_5. We can also understand why the material property can be predicted from the training data without including the target raw materials. However, we cannot determine why the error is the second largest in the case of Li_2O from this analysis. This may be because in the present PCA analysis, the dimension of fingerprints is reduced from 140 (seven properties by 20 bins) to 2, which may cause the loss of some information important to Li_2O.



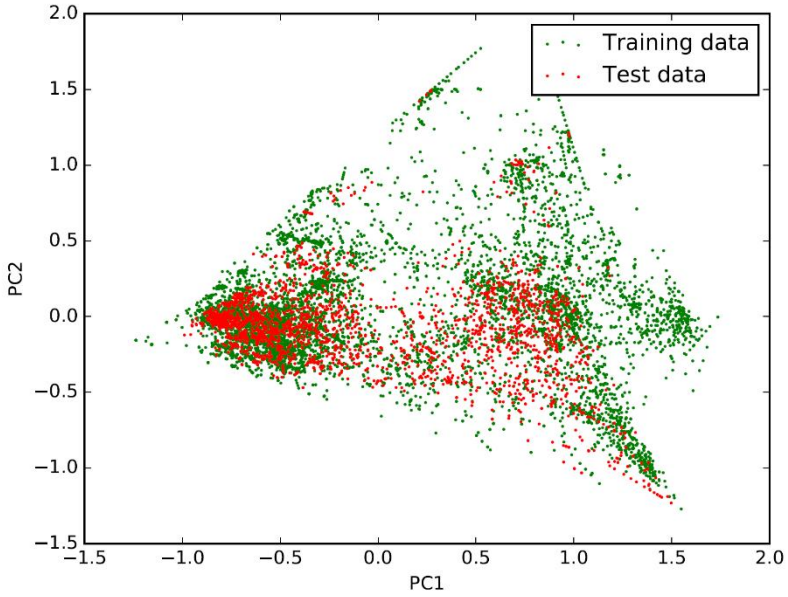
(a) Random 20%



(b) P_2O_5



(c) TiO_2



(d) Li_2O

Figure 4-5. Distribution of first two principal components of fingerprints. Test sets are divided by (a) randomly selected 20%, (b) P_2O_5 test case, (c) TiO_2 test case, and (d) Li_2O test case.

4.5.3 Similarity analysis of fingerprints

Considering the last speculation in the previous subchapter, we tried another analysis, which is based on the cosine similarity. Cosine similarity measures the similarity not of the lengths but of the directions of the two vectors via the inner product. It is widely used in high-dimensional data processing, such as text matching and data mining [85,86]. In our research, a fingerprint is regarded as a 140-dimensional vector. Because this vector constantly has identical length (specifically, the length measured by the L_1 metric) because the sum of the concentrations was constantly 100%, the cosine similarity method, which is specialized for angular information, was expected to be effective. The cosine value of two fingerprint vectors is defined as the cosine of the angle between the two vectors. Note that two fingerprints are more similar as the cosine value approaches 1.0. We also define the maximum cosine similarity—the maximum of the cosine value between the fingerprints of a material in the test set and that of a material in the training set.

In Figure 4-6, we plotted the reverse logarithm scale of the maximum cosine similarity versus prediction errors. The blue circles and orange triangles represent the Li_2O and SnO_2 test sets (test sets with the smallest prediction error), respectively. The dashed line represents the mean absolute error (MAE) within half the reverse logarithm magnitude (e.g., from 0.900 to 0.968 and from 0.968 to 0.990) of the maximum cosine similarity range. The MAE is 74 °C for a maximum cosine similarity of 0.9, whereas it is 27 °C for a similarity of 0.9999. An inverse correlation between the cosine similarity and prediction error was evidently observed—the closer the fingerprint of the test material is to one of the trained fingerprints, the more accurate the prediction.

We found that the tendency between the similarity and accuracy was the same for the cosine similarity analysis results using the molar concentration and elemental attributes as shown in Figure 4-7. This means that the prediction results from any of the descriptors tend to be more accurate when there are similar materials in the training data. However, the three descriptors used in this study showed different scales of cosine similarities. For example, the required values of similarity with a mean absolute error of less than 50 °C were about 0.984, 0.9994, and 0.9999996 for cation fingerprints, molar concentration, and elemental attributes, respectively. Therefore, the absolute criterion will depend on the combination of the target material property and descriptor.

From these results, we can say that properties of a material are relatively predictable when one of the ‘input features’ in the training set are similar to that of the test material.

Furthermore, it is possible to evaluate the prediction accuracy quantitatively using cosine similarity, while the required degree of similarity seems to be vary depending on the descriptors. However, even the similarity analysis could not reveal the reason why the cation fingerprint is more robust to the extreme case of material search than other descriptors. To clarify this remains as a future task.

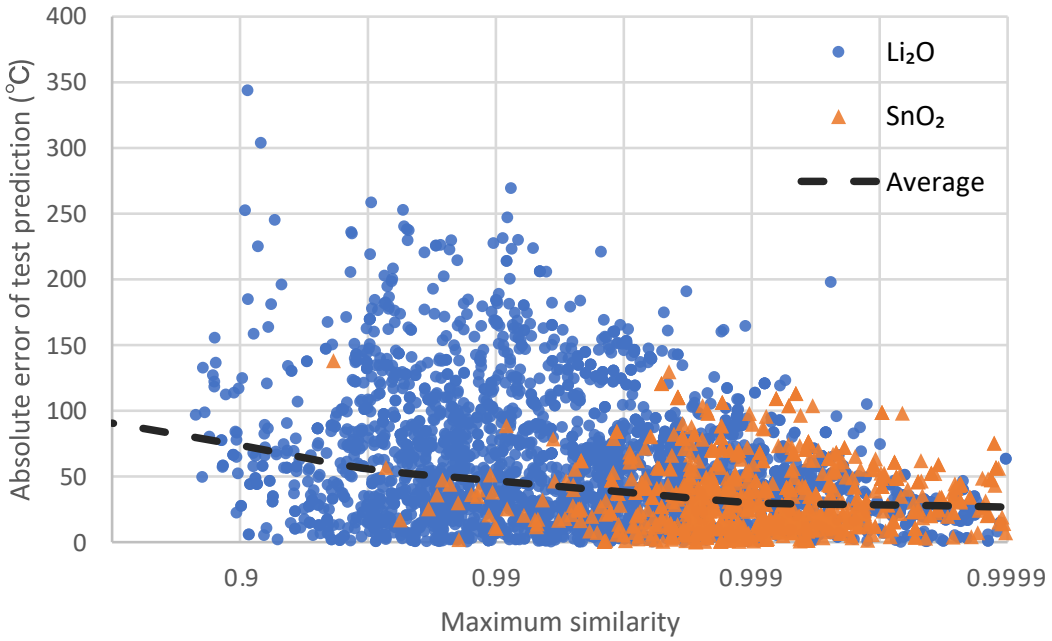
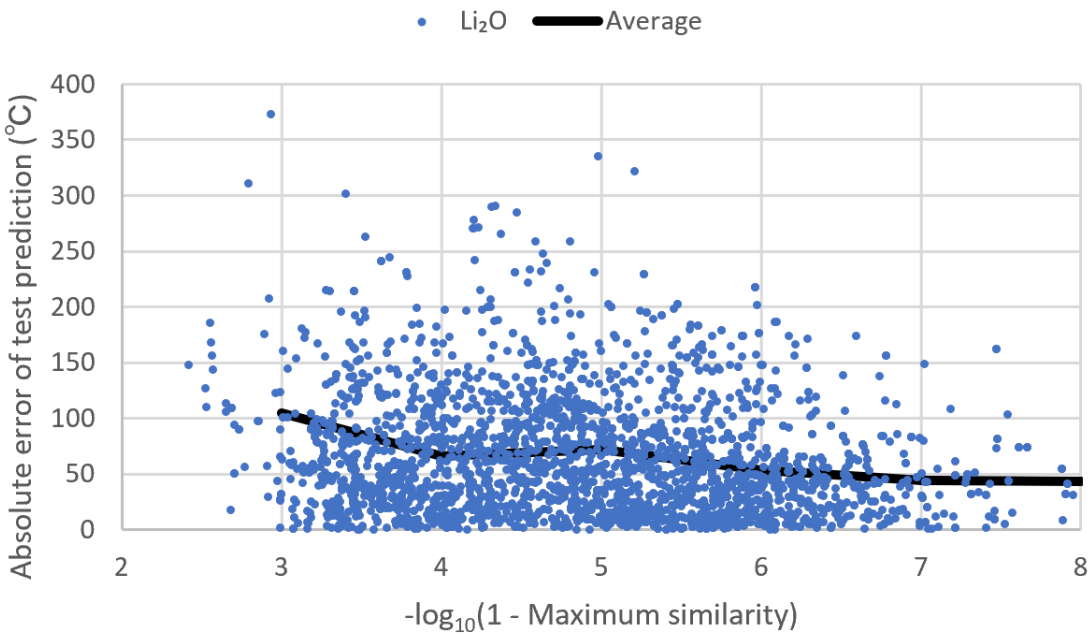


Figure 4-6. Maximum cosine similarity for training data (e.g. one dot of 0.99 means that this test data has no training data with cosine similarity larger than 0.99.) versus absolute error in Li_2O and SnO_2 test sets from Table 4-4. Dashed line represents the changes in mean absolute error over similarity range.

(a)



(b)

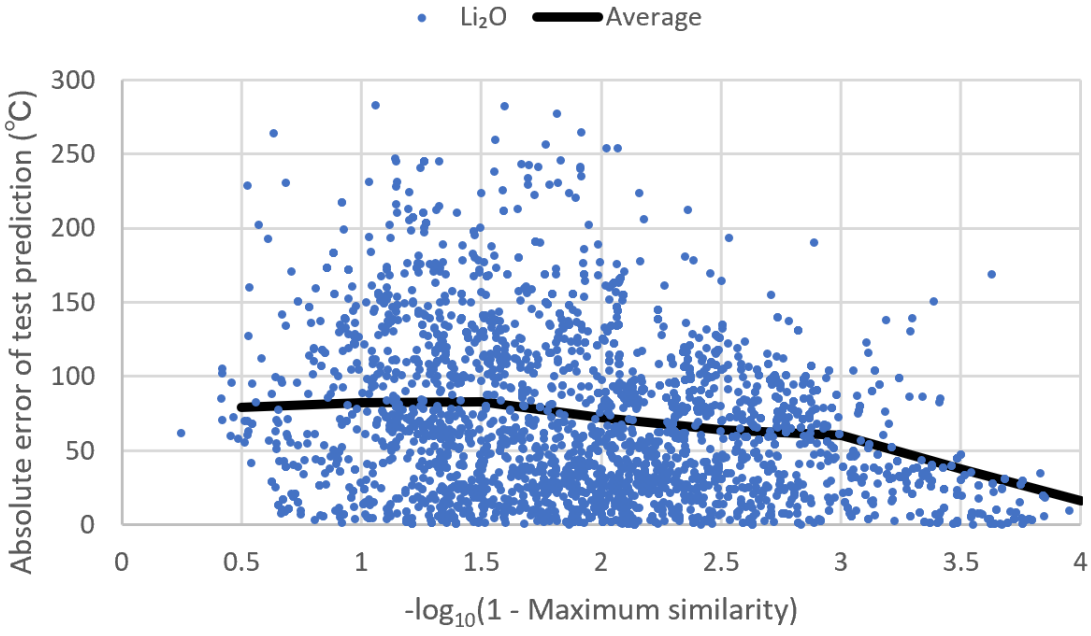


Figure 4-7. Maximum cosine similarity for training data versus absolute prediction error from (a) elemental attributes and (b) molar concentrations.

4.5.4 Demonstration for viscosity prediction of untrained novel materials

Finally, we show the experimental and predicted isokom temperatures at a viscosity level of $10^{6.6}$ Pa·s and raw compositions in Table 4-7. The first line shows the data of a test material with high Li concentration taken from the Li_2O test set, and the others show those of training materials that have the most similar fingerprints in the training set. In this case, the regularization scale was set to 0.3, as the maximum similarity is 0.9916. One interesting point seen in Table 4-7 is that all of the three training materials contain oxides not included in the test material, that is, CaO, La_2O_3, and TiO_2. This may be related to the fact that Li has an oxide density similar to Si and Ca, and a similar oxide melting point to those of Si and Ti, along with similar electronegativity to those of Ca and La. It is interesting to note that the isokom temperatures of the top-three materials with high similarity are quite different from that of the test material, even though the isokom temperatures of all of them are well predicted.

In the above, we demonstrate the successful performance of our method. However, we would like to note that this does not mean success in extrapolation; extrapolation is still challenging because we have high prediction error in the cases without materials that have high cosine similarity to the test material in the training ones. Instead, our results show that some of the extrapolation situations can be successfully transformed into interpolation situations. In our case, this is confirmed from the cosine similarity of descriptors. In addition, we show that it is possible to estimate the prediction accuracy by evaluating the descriptor similarity. However, further research is needed to determine how much the similarity penalized correlations can be generalized.

Fingerprint similarity		Raw oxides composition (mol %)								Isokom temperature at $10^{6.6}$ Pa·s	
		SiO_2	Al_2O_3	PbO	As_2O_3	CaO	La_2O_3	TiO_2	Li_2O	Experimental value	Predicted value
Test material		70.77	14.91	1.88	0.10				12.34	944	980.4
Training materials	0.9916	81.33	13.69	2.50		2.48				760	741.1
	0.9906	76.63	13.89	3.17		6.31				740	732.3
	0.9900	82.00	12.00				4.00	2.00		1110	1087.5

Table 4-7. One example of Li_2O test material and the top three training materials with similar fingerprints.

4.6 Conclusions

In this chapter, we have developed machine learning models to predict the viscosity of oxide glass materials using artificial neural networks to demonstrate the effectiveness of elemental fingerprints (called cation fingerprints in this chapter).

The prediction error for the isokom temperature, the temperature required to achieve a specific viscosity, is 33 °C in the case of cation fingerprints, which is lower than in the cases of elemental attributes (43 °C) and molar fraction (38 °C). In particular, for predictions for the dataset including novel raw oxides not used for training, the prediction error in the case of cation fingerprints is 53 °C, which is significantly lower than in the cases of elemental attributes (73 °C) and molar fraction (62 °C). Cation fingerprints have not only high predictive performance but are also able to identify important features that mainly determine a material property (such as the oxide density and melting point for viscosity) by varying primary features used in the construction of the fingerprints. In addition, we have suggested that the maximum cosine similarity of the descriptors between materials in the test and training sets can be used to estimate the prediction accuracy.

Chapter 5 Case study on first-principles calculation database

In this chapter, we try to enhance predictive performance by improving both descriptors and the ANN model. In addition, we attempt to explore candidate materials through Bayesian optimization for material discovery that is unavailable with the traditional prediction model. These studies were done using one of the first-principles calculation databases that enables us to compare prediction performance of fingerprint and ANN model with other reported results more easily than glass materials databases.

Subchapter 5.1 introduces the first-principles calculation database, called OQMD, used in this study. In Subchapter 5.2, standard formation enthalpy was used as the training objective. Standard formation enthalpy is one of the most important properties used in material development and screening to measure stability. Formation enthalpy is also often used as the first trial for incorporating machine learning into materials science. Therefore, there are many previous reports that can be easily compared with the training results in this study. We compared the prediction error with the case of using other descriptors. We also examined a method to improve the descriptor further. At the same time, as the descriptor, we tested improvements in the ANN model using adversarial training in Subchapter 2.4. We examined if two improvements could give better prediction performance on formation enthalpy.

Unlike other areas where deep learning is active such as image classification and language processing, data in materials science are often insufficient for learning. Due to this reason, an accurate prediction model using big data is often inadequate to use for material development. Therefore, in Subchapter 5.3, we try Bayesian optimization introduced in Subchapter 2.4, which efficiently explores various possibilities with as few observations as possible. The adversarial training adopted in Subchapter 5.2 not only improves the prediction accuracy but also enables Bayesian optimization from ANN models. However, Bayesian optimization using adversarial training has never been applied to material properties. In Subchapter 5.3, we set a hypothetical figure of merits using band gap energy and density to test whether Bayesian optimization is applicable for the material development process.

5.1 Open Quantum Materials Database (OQMD)

Since its theoretical basis was introduced in 1964, density functional theory (DFT) has been widely used in the field of materials science to investigate the electronic structure, simulate chemical reactions, etc. [35,36]. Open Quantum Materials Database (OQMD) is one of the largest first-principles calculation databases [87]. As of November 23, 2019, it consists of DFT calculations of about 640,000 inorganic crystal materials. One of the merits of OQMD is that the simulation conditions are strictly controlled. All the calculations were performed by the Vienna Ab-initio Simulation Package (VASP) [88] version 5.3.2 with the generalized gradient approximation (GGA) suggested by Perdew, Burke, and Ernzerhof (PBE) [89]. All other calculation settings, such as k-point density and cutoff energy are strictly controlled by calculation configurations.

Elemental fingerprints proposed in the present research have a one-to-one correspondence with the chemical formula. However, unless additional information about the crystal structure is added, elemental fingerprints cannot distinguish various possible crystal structures in the same chemical formula. Therefore, in order to secure the one-to-one correspondence between elemental fingerprints and DFT calculation results, we use the most stable crystal structure of each accessible chemical formula. We select the most stable material in each chemical formula with high accuracy calculation configurations (labeled as static or standard). Consequently, we got 253,153 inorganic materials and their calculation results.

From the database, we extract the data of total energy, electronic band gap energy, and density (calculated from volume per atom) to test our elemental fingerprints. Randomly chosen 90% of the total data was used for training the ANN model, and the rest of the 10% was used for testing the prediction accuracy of ANN model as we previously mentioned in Subchapter 2.3.1. For a fair comparison, the prediction error in this chapter refers to the prediction error of the test set unless otherwise noted.

5.2 Prediction of standard formation enthalpy

Except for some strongly correlated materials, total energy calculations using DFT have relatively high accuracy. OQMD reported that the mean absolute error (MAE) of DFT energy calculation is 81 meV/atom for 1,670 experimentally measured formation energies [87]. This high accuracy leads to high prediction capability. For example, in the Na-Cl system, the only known stable compound had been NaCI, but Oganov et al. predicted that $NaCl_3$ would stably exist under a high pressure above 60GPa from DFT calculations [90]. Their prediction was confirmed through experiments using the diamond anvil cell [90].

While cation fingerprints in Chapter 4 were generated using seven properties of raw oxides, elemental fingerprints in this chapter were generated using properties of pure elements. In addition, the elemental ratio was used in preparing histograms instead of the mole fraction of raw oxides. We used 22 properties suggested by Ward et al. that were originally prepared for the elemental attributes set [45]. Other details of methods are the same as those in Subchapter 3.1.

To prevent overfitting of the ANN model, in this chapter, we use the adversarial training with FGSM introduced in Subchapter 2.4. For the perturbation parameter (the degree of perturbation) ϵ of adversarial examples, we tested the recommended default value of 1% together with 0% (standard neural network), 0.1%, and 10%. Table 5-1 shows the optimal hyperparameter settings. We examine each hyperparameter and how it interacts with the neural network model and affects the prediction accuracy in the following subchapters.

Hyperparameters	Optimal settings
Activation functions	Exponential linear unit (ELU)
Adversarial perturbation (ϵ)	0.01 (1% of the input range)
Number of hidden layers	8
Number of bins	10
Number of independent networks	10

Table 5-1. Optimal hyperparameter settings

5.2.1 The choice of activation functions

Theoretically, a perfectly tuned ANN model can yield similar performances regardless of the activation function. However, in practice, the choice of activation function affects the speed of training process and the performance of the trained model.

At the beginning of neural network research, ANN models mainly used the sigmoid (logistic) function. However, the sigmoid function has the problem that the gradients change too slowly compared to the change of inputs. In other words, it could not distinguish well between the cases that require significant change of model parameters and those require slight change. As an alternative solution, hyperbolic tangent (Tanh), a rescaled version of the sigmoid function, has been used for a long time. In recent years, rectifier-based activation functions such as rectified linear unit (ReLU) [48] or exponential linear unit (ELU) [91], which can fix the vanishing gradient problem, draws attention as another alternative.

The vanishing gradient problem is a phenomenon in which the gradient converges to zero in the training of the ANN model, and thus training cannot progress. This is because the gradient decreases exponentially with the number of layers in the gradient calculation based on the chain rule since the gradient for each layer is between 0 and 1. Rectifier-based activation functions can suffer less this problem by fixing the gradient of one side (input is larger than zero) of the function to one because one is one even if it is multiplied by several times.

ELU is an activation function suggested by Clevert et al. after the advent of ReLU and its several derivatives [91]. While ELU has a similar shape to ReLU, it is a rectifier that uses the exponential function to make the activation function differentiable. While ReLU solves the vanishing gradient problem, it cannot guarantee the noise-robust deactivation state. Clevert et al. claimed that ELU could solve both issues. Possibly this characteristic of ELU allowed it to be a more suitable activation function for neural network ensembles by accurately evaluating the predictive uncertainty in this study.

In this subchapter, we compare the prediction performance of eight different activation functions. Table 5-2 shows the mean absolute error (MAE) of standard formation enthalpy predictions. We can see that the two activation functions ReLU and ELU had exceptionally high performances. Both functions are based on a rectifier instead of the traditional S-shaped function. These results support the above-mentioned recent research trend of choice of activation function

for the training of deep neural networks. When training a single neural network, ReLU was the best activation function, even though the difference was tiny.

However, in training with the neural network ensemble, an interesting difference is seen. The numbers in the parenthesis of Table 5-2 refer to prediction errors from the ensembles consisting of 5, 10, and 20 networks, respectively. To obtain these results, we trained 20 ANN models using ReLU or ELU, and then calculated the ensemble average $\mu_*(x)$ to evaluate prediction error as explained in Subchapter 2.4.2. As seen in Table 5-2, The prediction accuracy of neural network ensembles surpassed that of a single neural network. As a result, we adopted ELU as our activation function because ELU showed the best performance in the ensemble model. However, the performance difference with ReLU was small.

Type of activation function	Mean absolute error of formation enthalpy (meV/atom)
ReLU	**52.6** (**43.2**, 41.4, 40.4)
ELU	53.4 (43.5, **41.3**, **40.2**)
Softplus	55.6
Softsign	58.2
Tanh	61.6
SeLU	69.3
Hard sigmoid	73.9
Sigmoid	118.6

Table 5-2. Test of activation functions. The numbers in the parenthesis refer to prediction errors from the ensembles consisting of 5, 10, and 20 networks, respectively.

5.2.2 Degree of adversarial perturbation

Goodfellow et al., who suggested FGSM, recommended that an adversarial perturbation (ϵ) of 1% is appropriate in various neural networks [60]. However, it is necessary to confirm whether this value is also suitable for a numerical prediction problem in this study and how it can increase the accuracy of neural network ensembles. Here, we examine this point.

Figure 5-1 shows training curves without adversarial perturbation (simple neural network) and with 0.1%, 1%, 2%, and 10% perturbations. The lateral axis represents the training error, the vertical axis represents the validation error, and the dashed line indicates an ideal situation where these two errors are identical. As shown in Figure 5-1, validation errors are always larger than training errors, and their differences increase with the progress of training. In the case of a simple neural network, the lowest validation error of 63 meV/atom was achieved when the training error was approximately 25 meV/atom. With further training, the validation error increased again, while the training error continuously decreased. This is a common phenomenon of overfitting, which is explained in Subchapter 2.3. However, because the number of data in the dataset, 253,153, is huge, the overfitting effect seems minor.

Adding the adversarial training with perturbation from 0.1% to 2% not only prevented the overfitting but also lowered the overall validation errors. It is also worth noting that in these cases, validation error continuously decreased within the range of training in the present study.

On the other hand, 10% of perturbation led to an even larger validation error than that of the simple neural network. This may be because excessive perturbation interfered training of the model. Adversarial training forces adversarial example with perturbation and the original example have the same prediction results. We speculate that 10% of perturbation cannot train a model with satisfying this condition.

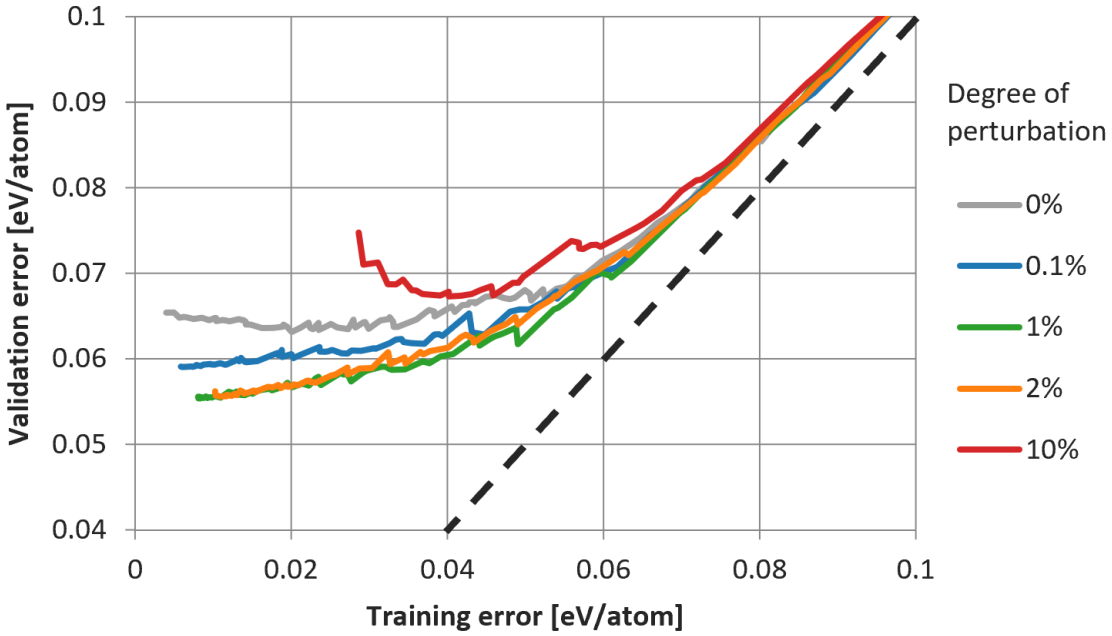


Figure 5-1. Trend of training-validation error depending on the perturbation parameter. The dashed line represents the ideal training condition.

5.2.3 Optimum number of hidden layers

The number of hidden layers is an important hyperparameter that dictates the number of all trainable parameters of the ANN model. Figure 5-2 shows the dependence of validation error on the number of hidden layers. This diagram shows validation errors of not only the single network but also the neural network ensembles that consist of five or ten networks. By varying the number of hidden layers from 4 to 16, we observed that the validation error decreased as the number of hidden layers increased. In the case of the single network, the best performance was achieved at 16 hidden layers.

In contrast, the ensemble of five networks showed the most accurate prediction results with a network of 12 hidden layers. This tendency was also observed in the ensemble of ten networks, where the best choice was the model with ten hidden layers. These results suggest that using more networks enables better prediction with a smaller number of hidden layers.

This is an interesting tendency and consistent with the claim by Lakshminarayanan et al. that neural network ensemble requires minimal hyperparameter tuning. In other words, over-tuned hyperparameters with a single network may decrease the performance of the neural network ensemble. Presumably, because the neural network ensemble is used to evaluate predictive uncertainty, over-tuned hyperparameters provide too small predictive uncertainty for unseen data, which leads to consistently wrong predictions. Because the validation error of an ensemble with ten networks with more than eight hidden layers had little improvement, we decided to use eight hidden layers.

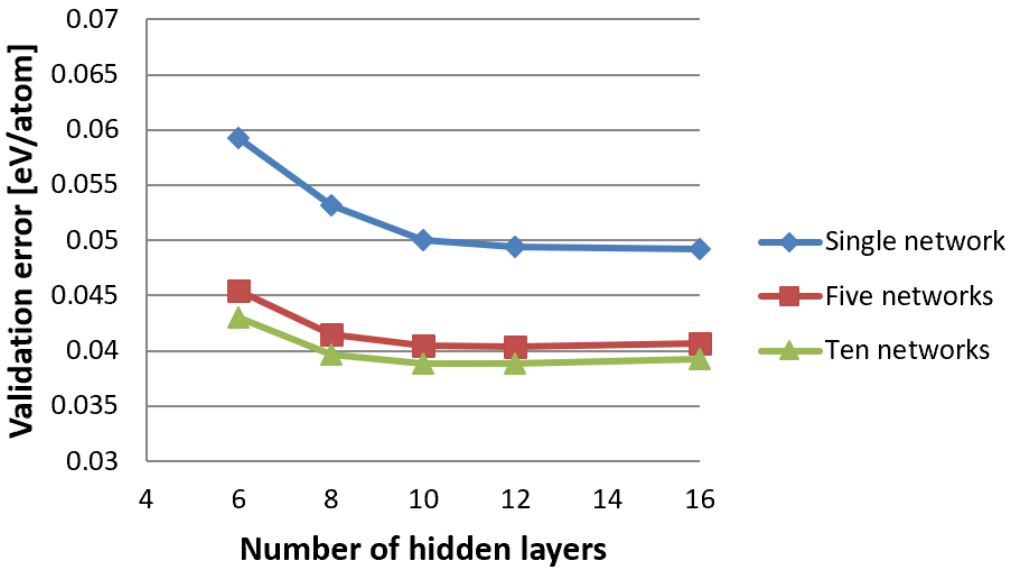


Figure 5-2. Effect of the number of hidden layers when using a single network to ten networks ensemble.

5.2.4 Number of bins and other descriptors

So far, all hyperparameter tuning was done by using elemental fingerprints with 10 bins as input features. However, the number of bins is also an important hyperparameter in a prediction model, as we investigated in Subchapter 4.4.1. In Subchapter 4.4.1, the optimal number of bins was 20,

and fingerprints showed better performance than elemental attributes and molar concentration. In this subchapter, we investigated the number of bins once again to see whether different results would occur if the target property were changed. We have also figured out a way to utilize these three descriptors more efficiently.

Figure 5-3 shows the mean absolute prediction errors (MAEs) of standard formation enthalpy as functions of the number of networks. Two dotted lines at the top of the graph are the prediction error of standard formation enthalpy from the decision tree regression model using elemental attributes reported by Ward et al. and the DFT calculation error of OQMD, respectively. As shown in this graph, the prediction error always decreases as the number of networks that constituting the ensemble increases. The two graphs of broken lines indicate the errors of using elemental molar concentration and elemental attributes as a descriptor, respectively. While elemental attributes perform slightly better than elemental molar concentration, both errors are much higher than the errors of elemental fingerprints.

In Figure 5-3, the prediction errors of elemental fingerprints are indicated as circular marks when using n bins with the range of n= 5 to 15. In short, a model with more bins always shows a lower prediction error. However, the difference of prediction error consistently decreases as the number of networks in the network ensemble increases. For example, when using 15 bins instead of 10 bins, the prediction error of a single neural network improved by more than 12%, while that of the ensemble of 10 networks improved by only about 5%. This tendency is similar to what we discussed in Subchapter 5.2.3. Since over-tuned hyperparameters are not helpful in ensembles, in this paper, we decided to use 10 bins when using elemental fingerprints.

Figure 5.3 also showed that the use of existing descriptors together is a better strategy beyond increasing the number of bins in elemental fingerprints. This strategy showed significant improvement in prediction accuracy. Diamond markers represent the case of energy prediction using elemental molar concentration, elemental attributes, and elemental fingerprints with five bins. This is a remarkable result considering that their information must overlap with each other because these three descriptors are all based on the same information. Presumably, this is because the way how that each descriptor expresses elemental information is different, and the information that the ANN model can read from each descriptor is different. Therefore, by using multiple descriptors together, it seems to be able to better express the parts that one descriptor cannot express. Finally, we achieved the prediction of the standard formation enthalpy within a mean

absolute error of 31 meV/atom. This prediction error is lower than the experimental thermochemical accuracy of the formation enthalpy, 1 kcal/mol (~43 meV/atom) [92], and also better than the DFT calculation error, 81 meV/atom [87].

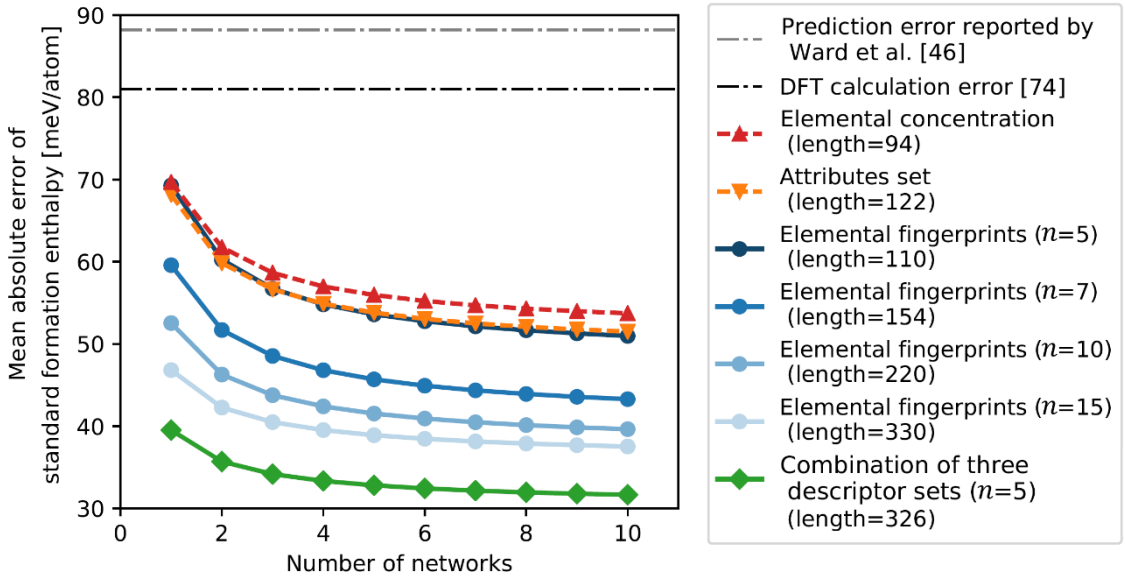


Figure 5-3. Standard formation enthalpy prediction error of test set from various descriptors. 'Length' in the parenthesis represents the length of input features, and n represents the number of bins.

5.3 Material discovery under virtual situation: band gap energy and density

In this subchapter, we examine the effectiveness of approximating the posterior distribution with an adversarial neural network ensemble in material discovery using the Bayesian optimization. As described in Subchapter 1.2.4, material discovery using the Bayesian optimization is one of the most promising application of machine learning to materials science. It is known that the

computational cost of the standard procedure of the Bayesian optimization, which use the Gaussian process to estimate posterior distribution, increases rapidly as the number of data increases. For such many data cases, posterior distribution with an adversarial neural network ensemble is expected to be effective.

In Bayesian optimization, better figure of merit (FOM) is searched based on the previously observed FOMs. Since the use of many data is crucial for our purpose mentioned above, we adopted OQMD for the present study. However, OQMD currently lacks material properties interesting enough to search for actual materials. Therefore, in this study, we set a virtual FOM described in Equation (5-1) for demonstration purpose.

$$\text{Figure of Merit (FOM)} = \text{Band gap energy(eV)} \times \text{Density}(^{\text{g}}/_{\text{cm}^3}) \quad (5\text{-}1)$$

Figure 5-4 shows the correlation between the density of inorganic crystals and the electronic band gap energy calculated by GGA-PBE exchange-correlation functional. As can be seen in this figure, the band gap energy of materials tends to be inversely proportional to the density. For example, materials with a high density such as Pu, Os, and Ir_3Re tend to be metallic. In contrast, materials with a low density such as He, B_2H_5, and SiH_8 tend to be insulators. We expect that the present problem setting would be a good prototype of materials discovery because of the above conflict of the two simple properties.

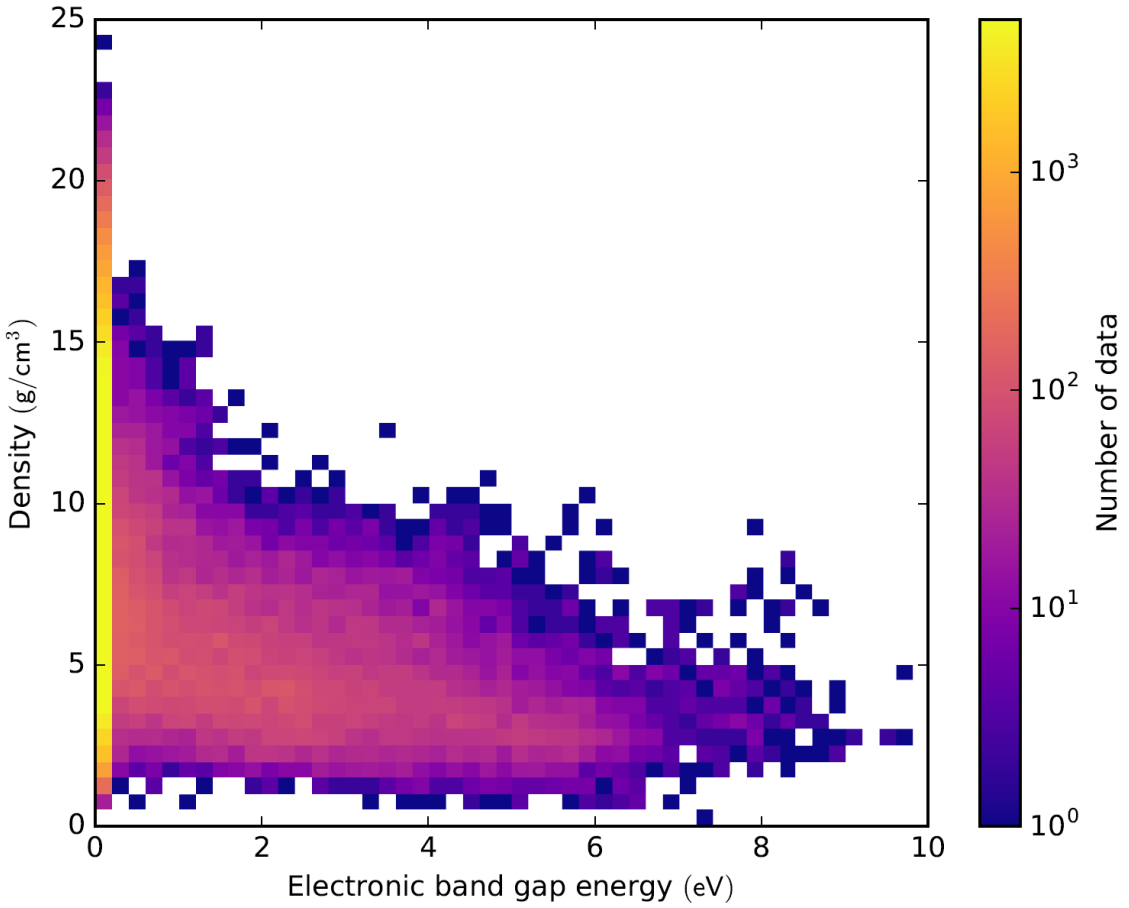


Figure 5-4. The correlation between the density of inorganic crystals and the electronic band gap energy calculated by GGA-PBE exchange-correlation functional on 253,153 data. Note that many materials in the bright yellow region have zero band gap energy.

As mentioned before, the computational cost of Bayesian optimization increases rapidly with the number of data, and in consequence no one has attempted to apply Bayesian optimization for a large number of inorganic materials as many as 253,153. To speed up the trial-and-error in this study, we experimented on material discovery with a limited number of data. That is, we used only the materials that contain one or more oxygen atoms in their unit cells. In this case, the number of candidate data was 16,444, and the material with the highest FOM was $F_5Lu_3O_2$ (ICSD-80365) with a band gap energy of 6.1 eV and a density of 9.3g/cm^3.

In the present study, we used the elemental fingerprints with 10 bins as the descriptor. We used ten networks to evaluate the predictive uncertainty. Note that this number, ten, is larger than that used in the study of Lakshminarayanan et al. for MNIST image classification problem, five. We set so to evaluate the predictive uncertainty more accurately. In addition, to prevent the neural network model from becoming too complicated, only two hidden layers were used for each network.

For comparison, we also tried materials discovery using the random search and the Bayesian optimization with the Gaussian process. The Gaussian process consisted of four kernels to represent various smoothness of objective function effectively. Two were the Matern kernel with $\nu = 1.5,\ \nu = 2.5$, another one was the RBF kernel (identical to Matern kernel with $\nu \rightarrow \infty$), and the last one was the white kernel to represent the noise of objective function. The scale of each kernel was automatically adjusted by the L-BFGS optimizer.

5.3.1 Comparison of material discovery performance

Figure 5-5 shows the results of applying the Gaussian process (GP), adversarial neural network ensemble (ANNE), and random search for the virtual material discovery. Each search was performed five times, beginning with a different random seed. The lateral axis of the graph represents the number of observations, while the vertical axis represents the highest FOM found within the corresponding number of observations.

Theoretically, a random search requires more than 8,000 observations before discovering the best candidate material. Only the top 1% of the candidates had a FOM larger than 40, and thus it was expected that the random search had to perform about 100 observations before finding the first material with a FOM larger than 40. Finding such a material was accomplished after about 25 observations by both GP and ANNE.

The average observation count required to discover $F_5Lu_3O_2$ was 59 for GP and 47 for ANNE. Despite the lack of theoretical background for accurately predicting posterior distribution, optimization results of ANNE are on par or even better than the results of GP. This indicates that ANNE using elemental fingerprints as input features may also be useful in material discovery.

We emphasize that the Bayesian optimization with GP or ANNE required 25 observations to achieve the result obtained by 100 random searches, while it required only less than 100 searches to obtain the champion material as opposed to 8,000 random searches. This result illustrates how efficiently Bayesian optimization suggests the next observation point.

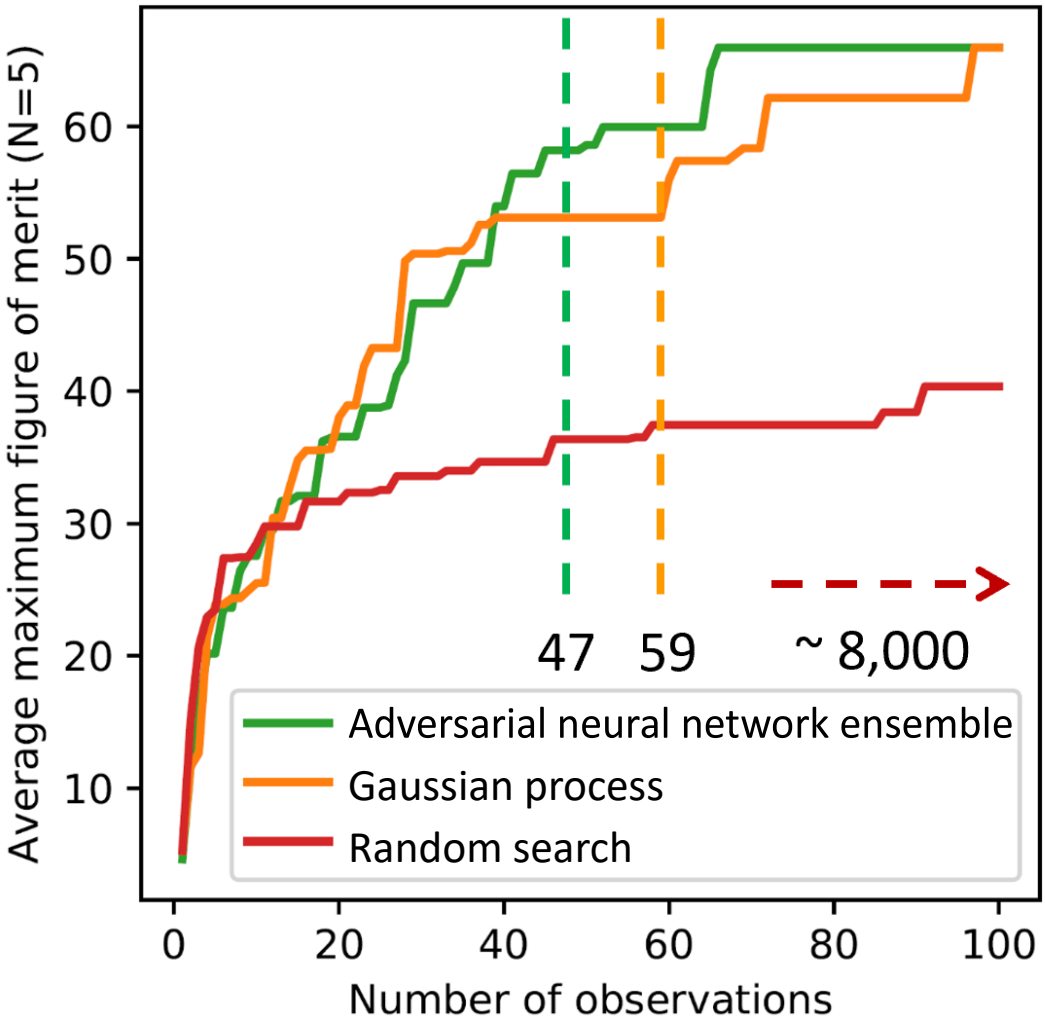


Figure 5-5. Virtual material discovery by Adversarial neural network ensemble, Gaussian process, and random search among 16,444 candidates. The figure of merit is $\text{Band gap energy(eV)} \times \text{Density(g/cm}^3\text{)}$. Next observation points (materials) were selected by maximum expected improvement. Dashed lines point to the required average observations in each method.

5.3.2 Importance of adversarial training for Bayesian optimization

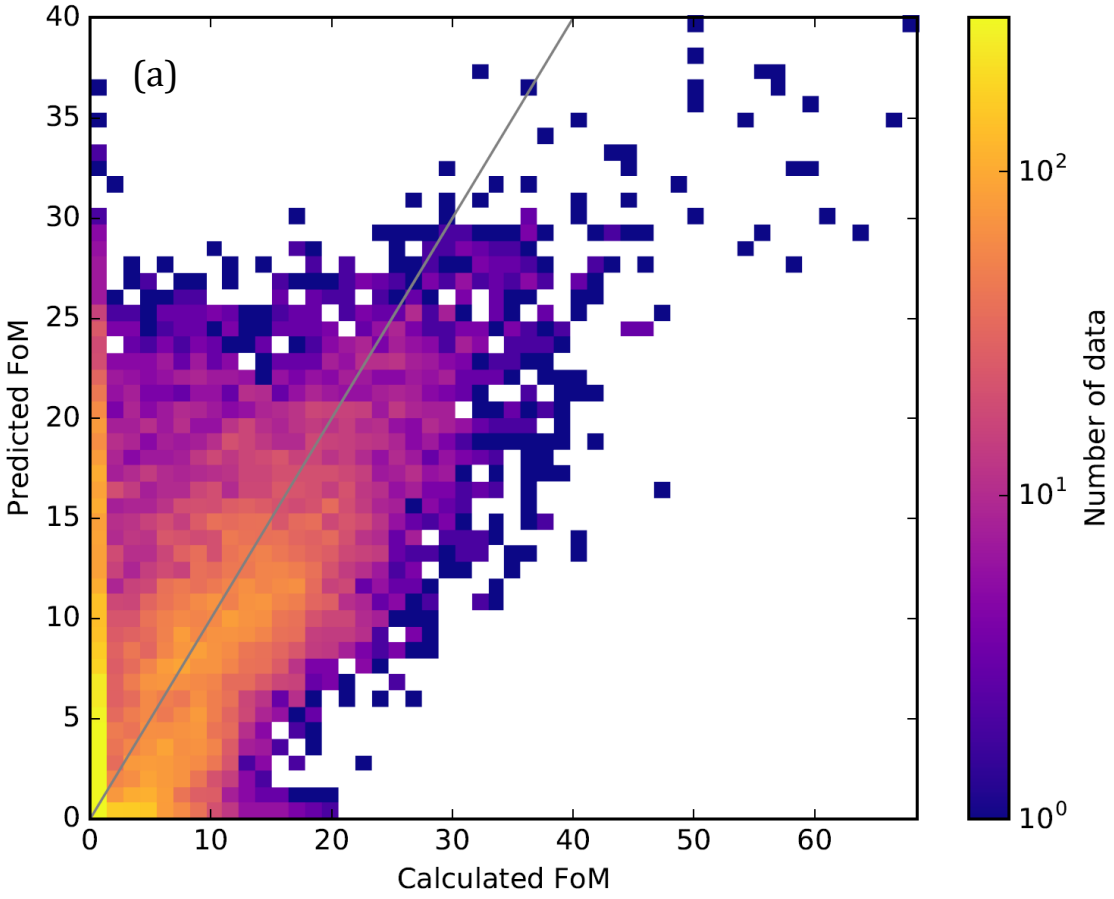
We also tested a simple neural network ensemble (SNNE) for material discovery to identify the effect of adversarial training on Bayesian optimization. In this test, we applied the Bayesian optimization to all of 26,784 materials listed on ICSD instead of the data used in the previous subchapter which were screened by the condition of containing oxygen atoms. The champion material in this case is ErF_3 (ICSD-81411), which has the band gap energy and density of 8.3 eV and 8.2g/cm^3, respectively.

When using the traditional GP for Bayesian optimization, the median of the required observation count to discover ErF_3 was 163. In contrast, the SNNE needed 186 observations, which means that its performance for material discovery was slightly lower than that of GP. Meanwhile, adversarial neural network ensembles with 1% and 10% perturbation required 119 and 115 observations, respectively. These results support the argument of Lakshminarayanan et al. that the random initial state of the neural network model alone may be sufficient in the evaluation of predictive uncertainty. Although Bayesian optimization was possible only with random initial state, adversarial training improved optimization performance surely. It is worth noting that although 10% perturbation performed worse than the 1% perturbation in Subchapter 5.2.2, its material discovery performance was slightly better than that of the 1% perturbation in the test of Bayesian optimization.

Figure 5-6 is a heatmap illustrating how accurately ANNE with 10% perturbation and 200 observations predicts the FOM of the total number of 26,784 materials. Note that these 200 observations were not randomly selected but were selected to find materials with high FOM. Because less than 1% of the entire dataset was used for training, the overall prediction accuracy is very low. However, the coefficient of determination (R^2) of the entire prediction is about 0.4. This value is surprising because this suggests that ANNE captures the trend of more than 99% of unseen data. When predicting the top 100 materials with high FOM, 45 were actually in the top 100. Due to the characteristic of Bayesian optimization that searches the maximum FOM, it tends to predict materials with high FOM more accurately.

As mentioned previously, the Gaussian process is used in Bayesian optimization due to its ability to calculate the variance of predicted values analytically. However, because n training points have a computational complexity of $\mathcal{O}(n^3)$, it is not easy to train more than thousands of

data with the Gaussian process [93]. In this subchapter, we demonstrated that in such a situation, adversarial neural network ensemble can not only predict based on the big data but also perform material discovery more efficiently by Bayesian optimization than Gaussian process. This also shows the new potential of neural network models.



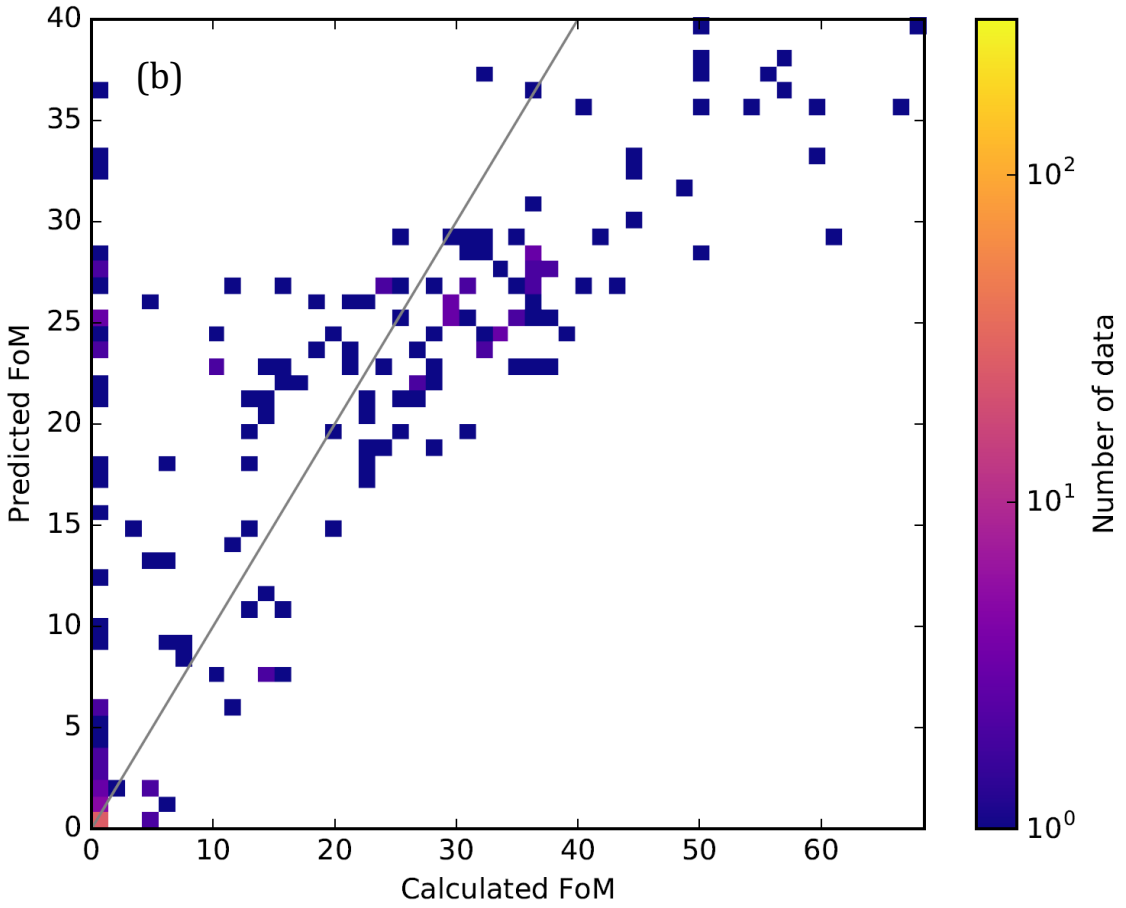


Figure 5-6. Prediction results of Figure of Merit (FoM, Band gap energy(eV) × Density($^{g}/_{cm^3}$)) with a solid line of ground truth. Heatmap represents the number of data normalized by decimal logarithm. The darkest blue area represents one data point in each region, and the brightest yellow area represents more than 300 data points in each region. When (a) both 200 training data (suggested by Bayesian optimization) and 26,584 test data are plotted, or (b) only 200 training data are plotted.

5.4 Conclusions

In this chapter, using the first-principles calculation database, we examined how to improve fingerprints and neural network models and how well these combinations can predict and explore material properties.

In Subchapter 5.2, a model to predict the standard formation enthalpy was developed. In this process, among the results obtained using elemental fingerprints, elemental attributes, and molar fraction as descriptors, fingerprints showed the best prediction performance. When three descriptors, including fingerprints, were used at the same time (descriptor set), an even lower prediction error was achieved. These results suggest that elemental fingerprints seem to contain more information or their information seem more understandable for neural network models, than the other two descriptors. The results also suggest that probably the information expressed by the other two descriptors is not completely included in the fingerprints. Considering the fact that when we have the necessary information to create elemental fingerprints, we can also create the other two descriptors, it will be effective to use all of the three descriptors together.

The ANN model was also improved. We confirmed that the neural network ensemble trained by adversarial training shows improved predictive performance than a traditional single neural network. In addition, the investigation of the effects of the degree of adversarial perturbation showed that the adversarial training can suppress overfitting.

An ensemble consisting of 10 neural networks with three descriptor sets predicted the standard formation enthalpy with an error of 31 meV/atom. This prediction performance is better than those of crystal graph (39 meV/atom) [43], ElemNet (50 meV/atom) [28], and elemental attributes with decision tree regression (71 meV/atom) [45]. It is worth noting that the prediction error of our model, 31 meV/atom, is lower than the experimental thermochemical accuracy of the formation enthalpy, 1 kcal/mol (~43.3 meV/mol) [92]. This accuracy of our best model means that it will be useful in practical material research such as screening of candidate materials using formation enthalpy.

In Subchapter 5.3, we tested whether we could efficiently search for candidate materials with as little data as possible using Bayesian optimization. A typical feedforward neural network

model cannot perform Bayesian optimization, but a neural network ensemble with adversarial training used in Subchapter 5.2 made Bayesian optimization possible.

We attempted a hypothetical material search for a total of over 16,000 materials with the product of the bandgap energy and density as a figure of merit (FoM). This hypothetical FoM may not be particularly interesting in practice, but was set up to examine our method in the searching process. As a result, neural network ensembles using fingerprints succeeded in finding the best material out of 16,000 candidates with an average of 47 observations. This means that the optimal setting of an experiment with up to 16,000 possibilities can be found in less than 50 trials.

In conclusion, the descriptor set including elemental fingerprints and the neural network ensemble model can not only accurately predict the material properties, but also accelerate material discovery or optimization with Bayesian optimization in cases of insufficient training.

Chapter 6　Case study on metallic materials

While elemental fingerprints are the compositional descriptor, properties of the material are not entirely determined by composition alone. The material to be made varies depending on the synthetic conditions, and the experimental conditions for testing the properties of the material also affect the test results. In practical research and development of materials, variables other than composition often play an important role. For example, in the case of metal, it is known that changes in microstructure due to post-processing, such as heat treatment, can significantly modify material properties such as higher tensile strength. In addition, the same strength test results also vary greatly depending on the temperature of the specimen.

In Chapter 6, we examine how effective the combination of the elemental fingerprints for ANN models in the cases where manufacturing processes and experimental conditions seem crucial. We decided to use a metallic materials database for this test. Metals are suitable targets for the above purpose of this chapter, because they have various combinations of composition and are also sensitive to variables other than composition. Furthermore, verification of machine learning is expected to be easily carried out because a large amount of experimental data have been accumulated for metals.

We attempted to train a model that predicts the results of tensile testing, which is one of the most basic and essential testing in quality verification or development of metal. Raw composition of metal, heat treatment conditions, and tensile testing temperature were used as independent variables for the prediction model, and four properties observed by tensile testing were used as dependent variables. Through this experiment, we investigate whether the ANN model using fingerprints can simultaneously evaluate the effects of the metal composition as well as the tensile testing temperature and heat treatment conditions.

6.1 Single prediction model throughout various alloys

Metal alloys, including copper, iron, and aluminum, are dominant structural materials in modern civilization. For this reason, with the development of science, discovering and understanding metals have developed as well. Although metals have been studied for a very long time, various research on the development of novel alloys are still ongoing. They are ranging from materials staying stable at very high temperatures, which are required in energy plants and engine turbines, to materials remaining ductile at very low temperatures, which are required in liquid gas storage and space programs. One of the examples is nickel-based superalloy. This superalloy can maintain good mechanical properties even at exceptionally high temperatures above 650 °C that steel cannot withstand, and many studies are still discussing how to improve it further [94-97]. As operating heat engines at higher temperatures give higher thermal efficiency, the evolution of these materials allows for more environment-friendly energy conversion.

As with the development of materials science, numerical prediction models to explain metallic properties have also been tried in various ways. However, similar to the prediction models for glass materials in Chapter 4, most models were made with limiting the range of the composition in the cases of metallic materials. For example, in order to predict the yield strength of nickel-based superalloys, Roth et al. have proposed a model that explains the effects of alloying elements on the solid solution hardening [98]. This prediction model, like the Fluegel equation discussed in Chapter 4, has parameters optimized for each element. Roth et al. have shown that the hardening strength was proportional to the square root of the total sum of the parameters multiplied by elemental compositions. However, this model was optimized by simulating the effects of each element on the materials independently.

Elemental fingerprints and ANN models could provide a breakthrough in this situation as we succeeded in describing not only the soda-lime glass but also the non-silicate glass by a single model in Chapter 4. In Chapter 4, we found that the melting point and density of oxides are important factors in determining the viscosity of glass materials as well as an accurate prediction. Elemental fingerprints and ANN model could enable a new understanding of metal properties through correlations that were not possible to analyze limited compositions. Therefore, we will

try to train a single model that covers a wide range of metals such as steel, aluminum alloys, and nickel alloys to explain metallic properties.

6.2 Tensile strength and creep-rupture lifetime

Tensile testing is an experiment where tension is continuously applied onto a specimen until the specimen breaks with observing the deformation (strain) and the engineering stress on the material at that strain. It is usually one of the measurements which are performed firstly in studies of metals. This is because the tensile testing can quantitatively provide many characteristics of materials such as yield strength, tensile strength, Young's modulus, etc. In modern experiments, specimen fabrication and experimental procedures have been standardized according to the American Society for testing and materials (ASTM) standard E8 [99] and the standard procedures have been widely used as a criterion for determining whether the desired metal has been properly made. Through these standardized tests, comparison of various metallic materials on the same basis has become possible.

Ultimate tensile strength (TS), which is one of the properties that can be measured by tensile testing, represents the maximum stress that a material can tolerate until it breaks. TS also changes depending on the test temperature: It usually decreases as the test temperature increases. At a high temperature, even before reaching TS, if the material experienced constant stress, a metal specimen will gradually get stretched, and the metal will eventually get destroyed after a certain time. This is called the creep rupture and needs to be carefully considered when designing engines and turbines which operate at high temperatures for a long time.

For example, SUS 403, one of the stainless steels, has TSs of 550 MPa at 450 °C and 730 MPa at room temperature [100]. However, when it was kept under the stress of 333 MPa at 450 °C, which is roughly about 60 percent of TS at 450 °C, the time to rupture was 746 hours (roughly one month). Under a lower stress of 294 MPa (roughly 53 percent of TS), the time to rupture was 11638 hours (roughly 1.3 years). These measured results mean that even if the force

is not enough to destroy the metal immediately, damages caused by its application at a high temperature over a long period of time can destroy metallic materials.

However, experiments for several months to more than ten years are necessary in order to acquire reliable results from these creep rupture tests. These long-term experiments are one of the reasons for the high costs and time consumption in the research and development of metals. Therefore, many studies with various approaches are actively underway to develop a model that could simulate or predict creep properties such as the time to rupture and minimum creep rate. Yoo et al. have proposed an ANN model for predicting the creep-rupture lifetime of Ni-superalloy [101]. Input features of their prediction model were only the weight percent of 11 elements used in metal, and two experimental conditions of temperature and stress. The output feature was creep-rupture lifetime (exactly speaking, the decimal logarithm of a lifetime was set as the training target because the lifetime was distributed in a wide range varying from 0.2 hours to 10,783 hours). Despite this simple configuration, the proposed ANN model showed quite successful predictions. By analyzing the results, it was seen that the $\log_{10}$(lifetime) of about 95% of the materials was estimated within ±0.5 of the scale. This means that if the lifetime of actual material is one year, the model would predict the lifetime to be between four months to three years.

The results of Yoo et al. are very interesting; however, they are applicable only to Ni-superalloys, and vulnerable to unseen data since the model uses raw elements as input features. In addition, in spite of the fact that the physical properties of the metal can drastically change depending on the heat treatment conditions, even for the same chemical compositions, their model does not take heat treatment conditions into consideration. As Li et al. [102] reported, the creep-rupture lifetime might differ by one order magnitude depending on the aging temperature of a certain Ni superalloy. Therefore, it is necessary to consider the heat treatment conditions to make an accurate prediction model.

6.3 Metallic materials database

A reliable database is essential for the training of material properties. Metallic Material Database (Kinzoku) [103] and Creep Data Sheet (CDS) [104] provided by the National Institute for Materials Science (NIMS) of Japan is one of the largest metal databases available free of charge.

Kinzoku is a database of experimental results of metals from various academic papers, but the update support has been suspended. In this database, even for the same materials, experimental conditions may have been omitted depending on the papers that reported the results. For example, in the case of materials that were standardized by ASTM or Japanese Industrial Standards (JIS), not only the chemical compositions but also the heat treatment conditions were provided in the standard, and it is known that these conditions may not be specified in the papers and consequently not included in Kinzoku. This means that the original data may be incomplete and cause errors when a prediction model is developed.

On the other hand, CDS includes many controllable experimental conditions relatively clearly. Furthermore, as CDS includes the tensile testing results of each sample, it is possible to simultaneously train both the tensile properties and creep-rupture lifetime based on the chemical compositions and heat treatment conditions. However, compared to Kinzoku, CDS has a smaller number of materials because the creep rupture tests require a long period of time.

Considering the above, we decided to use four types of tensile testing results in Kinzoku to train ANN models using elemental fingerprints.

6.4 Prediction of tensile properties without considering heat treatment

First, we train the yield strength, tensile strength, elongation, and reduction of area measured by tensile testing using Kinzoku. The independent input variables of this prediction model are compositions represented by elemental fingerprints and tensile testing temperature. From a total

of 3,500 data points, 20% of randomly selected data was used as a test set to verify the training model. Hyperparameter settings of the ANN model are shown in Table 6-1.

Because the number of data was not large, we used the ANN model with two hidden layers to predict tensile properties. As in Chapter 5, 22 elemental properties proposed by Ward et al. [45] were used as elemental properties for creating elemental fingerprints. Training results of the four tensile properties based on these models are shown in Table 6-2.

In both the training and test of four properties, the coefficient of determination was higher than 0.9. Therefore, we can say that the ANN model can explain the properties of the metals well. These prediction results are remarkable, considering that the model was trained without any restriction of the chemical compositions of metals. Among the tensile properties, elongation and reduction of area had relatively low prediction accuracy, though the difference was not large. This difference is ascribed to the fact that both the properties are measured after fracture of the metal specimens. In other words, since crack propagation just before the fracture occurs randomly even on the same material, there may be some uncertainty in the experimental results.

The prediction error of tensile strength was 51 MPa, which is comparable with an experimental error. For example, let us examine the case of SUS 403 mentioned above. In CDS, a total of nine specimens of SUS 403 were tested. The average tensile strength at room temperature was 708 MPa with standard deviation of 61 MPa [100], while the predicted tensile strength was 720 MPa. Thus, it can be said that the strength prediction by the ANN model has an accuracy of a similar level to that of the experimental error.

Figure 6-1 showed the comparison of experimental data and prediction for elongation and tensile strength. In the case of elongation, the data points are more widely distributed from the ground truth, which represents the lower prediction accuracy. In the case of tensile strength, the overall dispersion is small. However, there seems a tendency that the higher the tensile strength is, the larger the prediction error is. This may be due to the lack of heat treatment conditions in the input features on the current prediction model. The high prediction error may originate from the hardening effects by the heat treatments.

Number of hidden layers	2
Number of elemental properties	22
Number of bins for fingerprints	10
Number of neurons in each hidden layer	220
Activation function	Rectified linear unit
Optimization algorithm	Adaptive moment estimation

Table 6-1. Hyperparameters of the neural network model

	Training results		Test results	
	Root mean square error	Coefficient of determination	Root mean square error	Coefficient of determination
Yield strength	40 MPa	0.959	49 MPa	0.940
Tensile strength	33 MPa	0.977	51 MPa	0.951
Elongation	4.0 %	0.930	4.5 %	0.926
Reduction of area	5.4 %	0.932	6.2 %	0.911

Table 6-2. Training results of four tensile properties

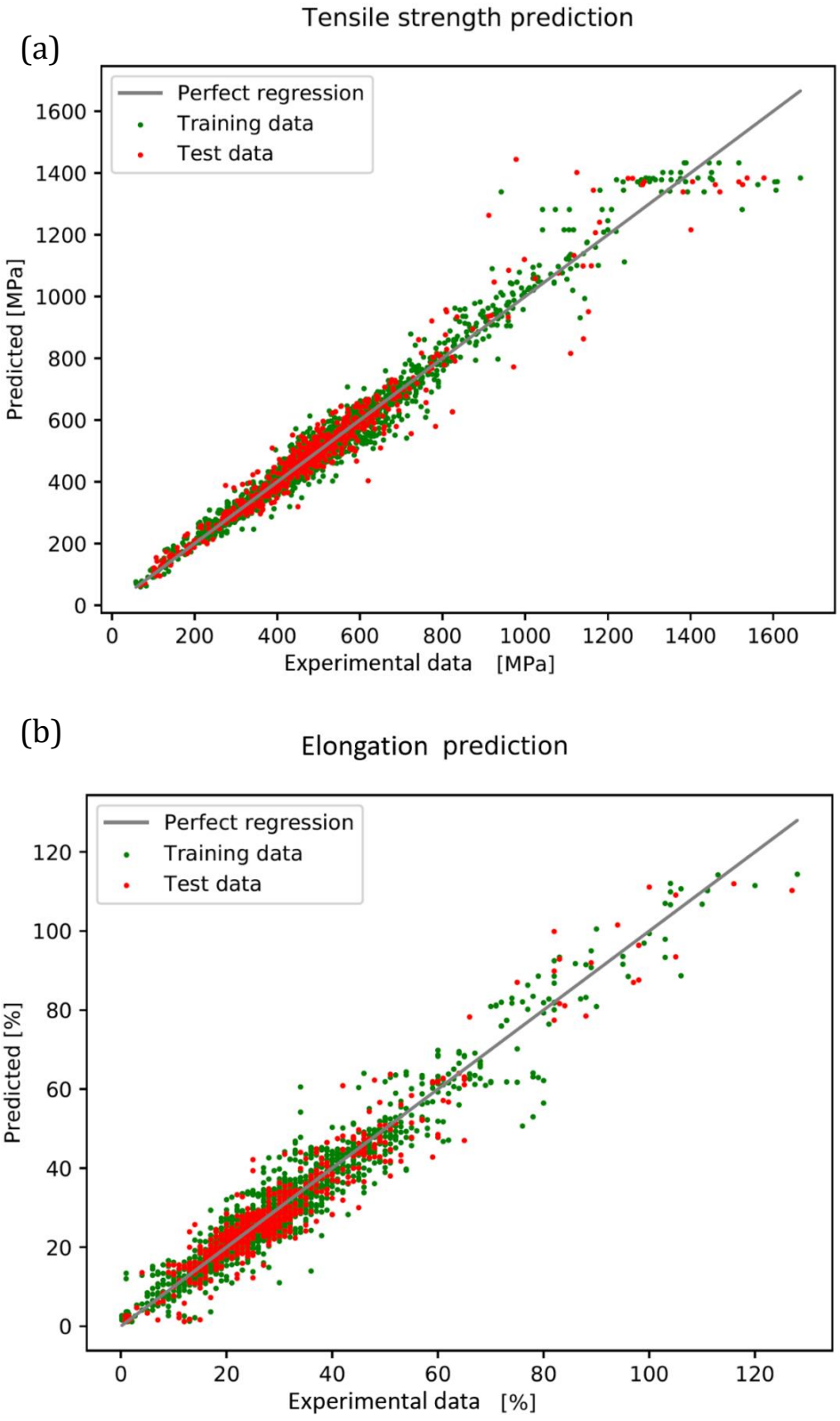


Figure 6-1. Comparison of predicted values with experimental values on (a) tensile strength, and (b) elongation.

6.5 Prediction of tensile properties for heat-treated metals

The prediction model in the previous subchapter does not use heat treatment conditions as input features, as well as the models proposed by Roth et al. [98] and Yoo et al. [101]. Since it is well known that heat treatment conditions can drastically change the properties of metals, such as tensile strength, we examine how well our prediction model could explain the change of structural properties due to heat treatment in this subchapter. For this purpose, the data was prepared as follows. In contrast to the test set in the previous subchapter, prepared by a random sample of 20% of the entire data set, this time, the materials without heat treatment conditions were extracted from Kinzoku and were set as a training set, and the materials with heat treatment conditions were set as a test set. Although there could be some materials with a missing indication of heat treatment condition in the training set, materials in the test set would experience various heat processes. Note that the test temperature is still included in the input features, same as in the previous subchapter.

Figure 6-2 shows training data (without heat treatment) as gray dots and test data (with heat treatment) as black dots for the prediction of yield strength. The results were quite impressive with training data showing similar distribution as Figure 6-1 (a), whereas test data is distributed to the right side of the graph. This means that for many test data, the actual experimental yield strength is greater than the predicted yield strength. This discrepancy was not simply a large error (deviation) but had a consistency (bias). This biased discrepancy was consistent with the property changes due to the hardening effect from some heat treatments.

For example, a test data point in the upper right-side corner shows a material with the predicted yield strength of 870 MPa for the experimental yield strength of 1,760 MPa. In this case, the prediction error is greater than twice the true value, so this not a good result for the prediction model. However, since this prediction model trained the tensile testing for untreated metals, it predicts the yield strength of the untreated metals. Therefore, these large prediction errors can be considered reasonable: The lack of information on the heat treatment conditions is considered to be the main cause of the large prediction errors. However, from this tendency of prediction error, if the predicted tensile strength is significantly lower than the experimental value, we can assume that some heat treatment has been performed.

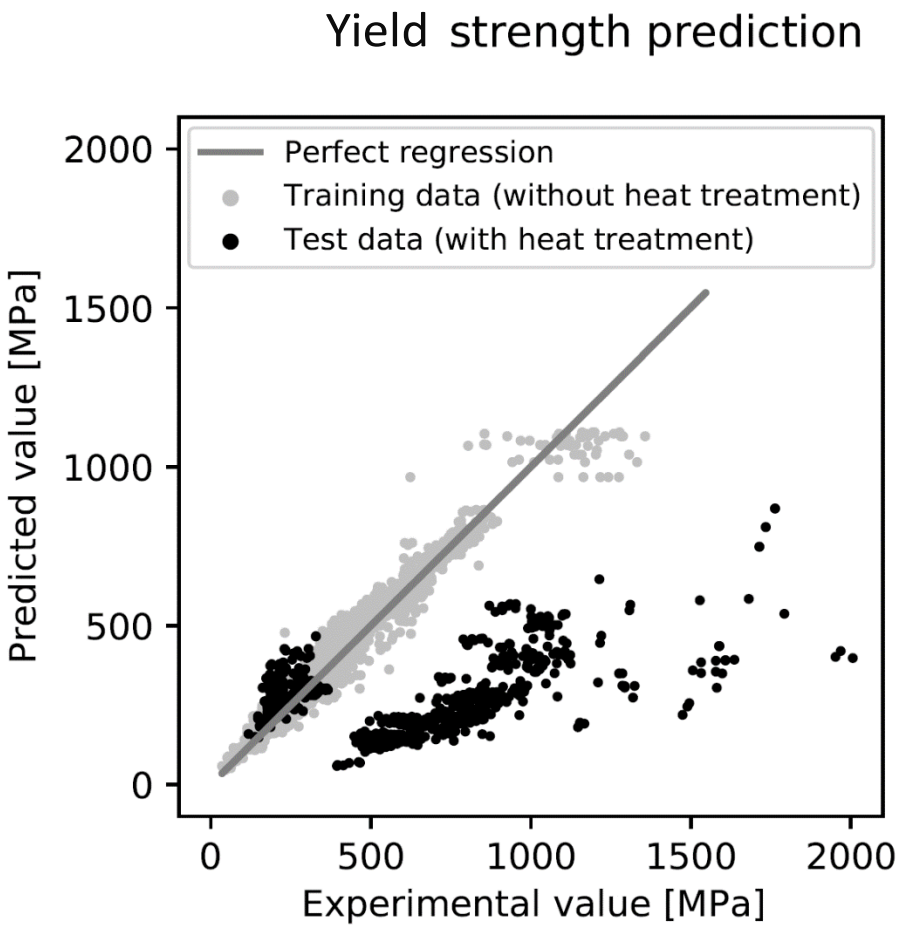


Figure 6-2. Prediction of yield strength from both training data (without heat treatment) and test data (with heat treatment) with perfect agreement line.

It is worth noting that even though the heat treatment conditions were not included as an input feature, Figure 6-1 (a) showed more accurate predictions than Figure 6-2. This may be because most of the metallic materials listed in the database already have optimal heat treatment conditions for the compositions through previous studies. In other words, if the chemical composition correlates with the heat treatment conditions and the training data is enough, the prediction model, even using only the chemical compositions as input features, may be able to describe the heat-treated metals.

It should be mentioned that the method using the heat process as input features should be studied as well. However, for this purpose, it is required to develop descriptors for explaining various heat process techniques such as annealing, tempering, normalizing, and quenching. One possibility could be to use a recurrent neural network that uses temperature, cooling medium, and time as input features. This remains a future task.

6.6 Conclusions

In this chapter, we trained several prediction models using elemental fingerprints with an ANN model that take only compositional information as inputs for tensile tests of metals. Although our ultimate goal on this subject is the development of prediction model that incorporate manufacturing processes and experimental conditions with the elemental fingerprints and an ANN model, no descriptor to represent the heat treatment conditions of metals has been reported yet. Considering this situation, as a preparatory step, we investigated how well the experimental results could be explained by only compositions and test temperature, excluding heat treatment conditions.

In Subchapter 6.4, both untreated metals and heat-treated metals are trained by a single model that did not use the heat treatment conditions as the input features. As a result, tensile strength was successfully predicted within the range of experimental error. In this model, it was observed that the higher the tensile strength, the larger the prediction error. This phenomenon appears because most of the high strength metals are heat-treated, and the heat-treated materials are predicted relatively inaccurately.

Therefore, Subchapter 6.5 was designed to find out what difference exits between the untreated metals and the heat-treated metals, and whether the prediction model could distinguish the two groups. For this purpose, data were divided into two groups according to the presence or absence of heat treatment of metal, and the prediction model was trained only by untreated metals. We observed that there are significant differences between prediction results of heat-treated metals from the prediction model and experimental results. As a result, it was found that there is

a significant tensile property difference between the heat-treated metals and the non-treated metals, and it is distinguishable by our prediction model.

However, these distinguishable differences contradict the results in Subchapter 6.4 that accurate predictions were possible. This seems to be because the optimal heat treatment conditions are correlated with the composition of the metal, and thus the effect of heat treatments can be evaluated indirectly from hidden correlations.

We remark that if the tensile testing results are strongly correlated with the effects of heat treatment, a prediction model with the tensile test results as input would be effective. Such a model could be used to predict the creep-rupture lifetime that requires enormous experimental cost and time, using tensile testing results that can be measured relatively easily. Trying to construct such a model would be an interesting future task.

Chapter 7 Conclusions

Through this dissertation, we have attempted to combine machine learning techniques with materials science for the prediction of material properties and the discovery of the best candidate material. In particular, we focused on the situation where the independent variables available for machine learning are just chemical formulas or composition of raw material and experimental conditions. In this study, we have proposed the elemental fingerprints as a new descriptor that could be created using the chemical composition of materials or molar concentration of raw materials.

In Chapter 1, general research trends in machine learning and attempts to incorporate machine learning into materials science were introduced. Since we propose a novel descriptor in this dissertation, we investigated the characteristics of various descriptors for material in detail to clarify the objective of this study.

In Chapter 2, various techniques for understanding and utilizing artificial neural network models used in the present study were introduced.

In Chapter 3, first, we pointed out the limitations of existing descriptors. Then the basic concepts of elemental fingerprints were explained. In addition to explaining the key principles underlying elemental fingerprints, the way to generate elemental fingerprints was also demonstrated using a chemical formula as an example. We also discussed several expected merits of elemental fingerprints.

In Chapter 4, we demonstrated the effectiveness of elemental fingerprints taking the prediction of the viscosity of oxide glass material as the training target. In addition, we have found that some interesting features of elemental fingerprints can be tested. First, we have found that the density and melting point of raw oxides are the most important physical properties to explain the viscosity of glass. Although the black box nature of the neural network model usually hinders the interpretation of the prediction results, elemental fingerprints make it possible to interpret the prediction results of the neural network model. Second, the possibility of prediction of unseen data has been shown. Elemental fingerprints outperformed other descriptors in the prediction of untrained materials.

In Chapter 5, the combination of the neural network model and elemental fingerprints have been applied to the training of standard formation enthalpy and other DFT calculation results. As there was a large amount of training data, the combination of the neural network model and elemental fingerprints was able to provide energy prediction much more accurately than the existing models. The prediction error of elemental attributes suggested by Ward et al. was 88 meV/atom [45], while the prediction error of neural network ensemble, which uses a descriptor set including elemental fingerprints and other two descriptors as an input feature, was 31 meV/atom. In addition, assuming a situation of material discovery, observation of the best inorganic material out of 16,444 candidates was successfully done in just 47 observations on average. This is equivalent to finding the optimal condition from 16,444 possibilities by conducting just 47 experiments. From this result, we expect that the present method can be used to optimize the experiments of actual materials in the future.

In Chapter 6, a prediction model has been trained by using the database of tensile testing of metals. In particular, it has been shown that a single model can describe all the materials regardless of base elements such as Fe, Ni, and Co. However, because heat treatment conditions were not used as an input feature, the prediction accuracy should be further improved. Even so, the test results of the heat-treated metal showed the possibility that the prediction model could explain the effects of heat treatments.

Through this study, we figured out the performance and possibilities of our novel descriptor. Elemental fingerprints successfully explain and predict the experimental or simulation results of materials and can discover new materials using the information of raw materials or pure elements. Since elemental fingerprints can be applied to various situations, it is expected to accelerate materials research and development using machine learning.

Bibliography

[1] Jain, Anubhav, et al. "Commentary: The Materials Project: A materials genome approach to accelerating materials innovation." *Apl Materials* 1.1 (2013): 011002.

[2] Rajan, Krishna. "Materials informatics." *Materials Today* 8.10 (2005): 38-45.

[3] Ramprasad, Rampi, et al. "Machine learning in materials informatics: recent applications and prospects." *npj Computational Materials* 3.1 (2017): 1-13.

[4] Pilania, Ghanshyam, et al. "Accelerating materials property predictions using machine learning." *Scientific reports* 3 (2013): 2810.

[5] Butler, Keith T., et al. "Machine learning for molecular and materials science." *Nature* 559.7715 (2018): 547-555.

[6] Schmidhuber, Jürgen. "Deep learning in neural networks: An overview." *Neural networks* 61 (2015): 85-117.

[7] LeCun, Yann. "The MNIST database of handwritten digits." *http://yann. lecun. com/exdb/mnist/* (1998).

[8] Lecun, Yann, et al. "Handwritten digit recognition with a back-propagation network." *Advances in Neural Information Processing Systems (NIPS 1989), Denver, CO*. Morgan Kaufmann, 1990.

[9] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.

[10] Ciregan, Dan, Ueli Meier, and Jürgen Schmidhuber. "Multi-column deep neural networks for image classification." *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012.

[11] Rawat, Waseem, and Zenghui Wang. "Deep convolutional neural networks for image classification: A comprehensive review." *Neural computation* 29.9 (2017): 2352-2449.

[12] Knuth, Donald E., and Ronald W. Moore. "An analysis of alpha-beta pruning." *Artificial intelligence* 6.4 (1975): 293-326.

[13] Strnad, Damjan, and Nikola Guid. "Parallel alpha-beta algorithm on the GPU." *Journal of computing and information technology* 19.4 (2011): 269-274.

[14] Bouzy, Bruno, and Tristan Cazenave. "Computer Go: an AI oriented survey." *Artificial Intelligence* 132.1 (2001): 39-103.

[15] Silver, David, et al. "Mastering the game of Go with deep neural networks and tree search." *nature* 529.7587 (2016): 484-489.

[16] Silver, David, et al. "A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play." *Science* 362.6419 (2018): 1140-1144

[17] Wu, Yonghui, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." *arXiv preprint arXiv:1609.08144* (2016).

[18] Johnson, Melvin, et al. "Google's multilingual neural machine translation system: Enabling zero-shot translation." *Transactions of the Association for Computational Linguistics* 5 (2017): 339-351.

[19] Sato, Shin. "On a new method of drawing the potential energy surface." *The Journal of chemical physics* 23.3 (1955): 592-593.

[20] Jones, John Edward. "On the determination of molecular fields.—II. From the equation of state of a gas." *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character* 106.738 (1924): 463-477

[21] Van Duin, Adri CT, et al. "ReaxFF: a reactive force field for hydrocarbons." *The Journal of Physical Chemistry A* 105.41 (2001): 9396-9409.

[22] Balachandran, Prasanna V., Scott R. Broderick, and Krishna Rajan. "Identifying the 'inorganic gene'for high-temperature piezoelectric perovskites through statistical learning." *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 467.2132 (2011): 2271-2290.

[23] Goldschmidt, Victor Moritz. "Die gesetze der krystallochemie." *Naturwissenschaften* 14.21 (1926): 477-485.

[24] Ghiringhelli, Luca M., et al. "Big data of materials science: critical role of the descriptor." *Physical review letters* 114.10 (2015): 105503.

[25] Gutzow, Ivan S., et al. *Glasses and the glass transition*. John Wiley & Sons, 2011.

[26] Fluegel, Alexander. "Glass viscosity calculation based on a global statistical modelling approach." *Glass Technology-European Journal of Glass Science and Technology Part A* 48.1 (2007): 13-30.

[27] Vatani, Ali, Mehdi Mehrpooya, and Farhad Gharagheizi. "Prediction of standard enthalpy of formation by a QSPR model." *International Journal of Molecular Sciences* 8.5 (2007): 407-432.

[28] Jha, Dipendra, et al. "Elemnet: deep learning the chemistry of materials from only elemental composition." *Scientific reports* 8.1 (2018): 17593.

[29] Stuart, Andrew M. "Inverse problems: a Bayesian perspective." *Acta numerica* 19 (2010): 451-559.

[30] Tanaka, Masataka, and George S. Dulikravich, eds. *Inverse problems in engineering mechanics*. Elsevier, 1998.

[31] Brochu, Eric, Vlad M. Cora, and Nando De Freitas. "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning." *arXiv preprint arXiv:1012.2599* (2010).

[32] Mockus, Jonas. "Application of Bayesian approach to numerical methods of global and stochastic optimization." *Journal of Global Optimization* 4.4 (1994): 347-365.

[33] Seko, Atsuto, et al. "Prediction of low-thermal-conductivity compounds with first-principles anharmonic lattice-dynamics calculations and Bayesian optimization." *Physical review letters* 115.20 (2015): 205901.

[34] Behler, Jörg. "Atom-centered symmetry functions for constructing high-dimensional neural network potentials." *The Journal of chemical physics* 134.7 (2011): 074106.

[35] Hohenberg, Pierre, and Walter Kohn. "Inhomogeneous electron gas." *Physical review* 136.3B (1964): B864.

[36] Kohn, Walter, and Lu Jeu Sham. "Self-consistent equations including exchange and correlation effects." *Physical review* 140.4A (1965): A1133.

[37] Alder, Berni Julian, and Thomas Everett Wainwright. "Phase transition for a hard sphere system." *The Journal of chemical physics* 27.5 (1957): 1208-1209.

[38] Kresse, Georg, and Jürgen Hafner. "Ab initio molecular dynamics for liquid metals." *Physical Review B* 47.1 (1993): 558.

[39] Kadau, Kai, Timothy C. Germann, and Peter S. Lomdahl. "Molecular dynamics comes of age: 320 billion atom simulation on BlueGene/L." *International Journal of Modern Physics C* 17.12 (2006): 1755-1761.

[40] Boes, Jacob R., et al. "Neural network and ReaxFF comparison for Au properties." *International Journal of Quantum Chemistry* 116.13 (2016): 979-987.

[41] Bartók, Albert P., Risi Kondor, and Gábor Csányi. "On representing chemical environments." *Physical Review B* 87.18 (2013): 184115.

[42] Bartók, Albert P., et al. "Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons." *Physical review letters* 104.13 (2010): 136403.

[43] Xie, Tian, and Jeffrey C. Grossman. "Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties." *Physical review letters* 120.14 (2018): 145301.

[44] Ye, Weike, et al. "Deep neural networks for accurate predictions of crystal stability." *Nature communications* 9.1 (2018): 3800.

[45] Ward, Logan, et al. "A general-purpose machine learning framework for predicting properties of inorganic materials." *npj Computational Materials* 2 (2016): 16028.

[46] Isayev, Olexandr, et al. "Materials cartography: representing and mining materials space using structural and electronic fingerprints." *Chemistry of Materials* 27.3 (2015): 735-743.

[47] Rosenblatt, Frank. "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review* 65.6 (1958): 386.

[48] Glorot, Xavier, Antoine Bordes, and Yoshua Bengio. "Deep sparse rectifier neural networks." *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. 2011.

[49] Minsky, Marvin, and Seymour A. Papert. *Perceptrons: An introduction to computational geometry*. MIT press, 1969.

[50] Cybenko, George. "Approximation by superpositions of a sigmoidal function." *Mathematics of control, signals and systems* 2.4 (1989): 303-314.

[51] Hornik, Kurt. "Approximation capabilities of multilayer feedforward networks." *Neural networks* 4.2 (1991): 251-257.

[52] Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors." *nature* 323.6088 (1986): 533-536.

[53] Robbins, Herbert, and Sutton Monro. "A stochastic approximation method." *The annals of mathematical statistics* (1951): 400-407.

[54] Spall, James C. "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation." *IEEE transactions on automatic control* 37.3 (1992): 332-341.

[55] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).

[56] Hawkins, Douglas M. "The problem of overfitting." *Journal of chemical information and computer sciences* 44.1 (2004): 1-12.

[57] Picard, Richard R., and R. Dennis Cook. "Cross-validation of regression models." *Journal of the American Statistical Association* 79.387 (1984): 575-583.

[58] Hoerl, Arthur E., and Robert W. Kennard. "Ridge regression: Biased estimation for nonorthogonal problems." *Technometrics* 12.1 (1970): 55-67.

[59] Ng, Andrew Y. "Feature selection, L 1 vs. L 2 regularization, and rotational invariance." *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004.

[60] Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *arXiv preprint arXiv:1412.6572* (2014).

[61] Eykholt, Kevin, et al. "Robust physical-world attacks on deep learning models." *arXiv preprint arXiv:1707.08945* (2017).

[62] Lakshminarayanan, Balaji, Alexander Pritzel, and Charles Blundell. "Simple and scalable predictive uncertainty estimation using deep ensembles." *Advances in neural information processing systems*. 2017.

[63] Gal, Yarin, and Zoubin Ghahramani. "Dropout as a bayesian approximation: Representing model uncertainty in deep learning." *33rd International Conference on Machine Learning, ICML 2016*. Vol. 3. 2016.

[64] Williams, Christopher KI. "Computing with infinite networks." *Advances in neural information processing systems*. 1997.

[65] Ghahramani, Zoubin. "Probabilistic machine learning and artificial intelligence." *Nature* 521.7553 (2015): 452-459.

[66] Debenedetti, Pablo G., and Frank H. Stillinger. "Supercooled liquids and the glass transition." *Nature* 410.6825 (2001): 259.

[67] Lubchenko, Vassiliy, and Peter G. Wolynes. "Theory of structural glasses and supercooled liquids." *Annu. Rev. Phys. Chem.* 58 (2007): 235-266.

[68] Vannoni, M., A. Sordini, and G. Molesini. "Relaxation time and viscosity of fused silica glass at room temperature." *The European Physical Journal E* 34.9 (2011): 92.

[69] Angell, C. Austen. "Formation of glasses from liquids and biopolymers." *Science* 267.5206 (1995): 1924-1935.

[70] Blodgett, M. E., et al. "Proposal for universality in the viscosity of metallic liquids." *Scientific reports* 5 (2015): 13837.

[71] Dyre, Jeppe C. "Colloquium: The glass transition and elastic models of glass-forming liquids." *Reviews of modern physics* 78.3 (2006): 953.

[72] Garca-Coln, L. S., L. F. Del Castillo, and Patricia Goldstein. "Theoretical basis for the Vogel-Fulcher-Tammann equation." *Physical Review B* 40.10 (1989): 7040.

[73] Fulcher, Gordon S. "Analysis of recent measurements of the viscosity of glasses." *Journal of the American Ceramic Society* 8.6 (1925): 339-355.

[74] Angell, C. Austen. "Entropy and fragility in supercooling liquids." *Journal of research of the National Institute of Standards and Technology* 102.2 (1997): 171.

[75] Cohen, Morrel H., and G. S. Grest. "Liquid-glass transition, a free-volume approach." *Physical Review B* 20.3 (1979): 1077.

[76] Moynihan, C. T., et al. "Crystallization and viscosity of heavy metal fluoride glasses." *Infrared Optical Materials and Fibers IV*. Vol. 618. International Society for Optics and Photonics, 1986.

[77] Frischat, Günther H. "Book Review: Mathematical Approach to Glass. By MB Volf." *Angewandte Chemie International Edition in English* 28.6 (1989): 820-820.

[78] SciGlass. 6 ed. San Leandro, CA: MDL Information Systems; 2003.

[79] INTERGLAD International Glass Database. 7 ed. Tokyo, Japan: New Glass Forum; 2016.

[80] Priven, A. I., and O. V. Mazurin. "Glass Property Databases: Their History, Present State, and Prospects for Further Development." *Advanced Materials Research*. Vol. 39. Trans Tech Publications, 2008.

[81] Weast, Robert C., Melvin J. Astle, and William H. Beyer. *CRC handbook of chemistry and physics*. Vol. 69. Boca Raton, FL: CRC press, 1988.

[82] Pedregosa, Fabian, et al. "Scikit-learn: Machine learning in Python." *the Journal of machine Learning research* 12 (2011): 2825-2830.

[83] Pearson, Karl. "LIII. On lines and planes of closest fit to systems of points in space." *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901): 559-572.

[84] Jolliffe, Ian T., and Jorge Cadima. "Principal component analysis: a review and recent developments." *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 374.2065 (2016): 20150202.

[85] Bayardo, Roberto J., Yiming Ma, and Ramakrishnan Srikant. "Scaling up all pairs similarity search." *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007.

[86] Huang, Anna. "Similarity measures for text document clustering." *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*. Vol. 4. 2008.

[87] Kirklin, Scott, et al. "The Open Quantum Materials Database (OQMD): assessing the accuracy of DFT formation energies." *npj Computational Materials* 1 (2015): 15010.

[88] Kresse, Georg, and Jürgen Furthmüller. "Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set." *Physical review B* 54.16 (1996): 11169.

[89] Perdew, John P., Kieron Burke, and Matthias Ernzerhof. "Generalized gradient approximation made simple." *Physical review letters* 77.18 (1996): 3865.

[90] Zhang, Weiwei, et al. "Unexpected stable stoichiometries of sodium chlorides." *Science* 342.6165 (2013): 1502-1505.

[91] Clevert, Djork-Arné, Thomas Unterthiner, and Sepp Hochreiter. "Fast and accurate deep network learning by exponential linear units (elus)." *arXiv preprint arXiv:1511.07289* (2015).

[92] Pople, John A. "Nobel lecture: Quantum chemical models." *Reviews of Modern Physics* 71.5 (1999): 1267.

[93] Williams, Christopher KI, and Carl Edward Rasmussen. *Gaussian processes for machine learning*. Vol. 2. No. 3. Cambridge, MA: MIT press, 2006.

[94] Chandra, S., et al. "Creep rupture data assessment of Alloy 617." *Creep & Fracture in High Temperature Components: Design & Life Assessment Issues* (2005).

[95] Ul-Hamid, A., et al. "Evolution of oxide scale on a Ni–Mo–Cr alloy at 900 C." *Materials characterization* 58.1 (2007): 13-23.

[96] Pettit, Fred. "Hot corrosion of metals and alloys." *Oxidation of Metals* 76.1-2 (2011): 1-21.

[97] Creep Data Sheet No. 59, " Creep data sheet no. 59 NCF 718-B (Ni based 19Cr-18Fe-3Mo-5Nb-Ti-Al)", NIMS, 2011, https://smds.nims.go.jp/MSDS/pdf/sheet/C59J.pdf

[98] Roth, H. A., C. L. Davis, and R. C. Thomson. "Modeling solid solution strengthening in nickel alloys." *Metallurgical and Materials Transactions A* 28.6 (1997): 1329-1335.

[99] ASTM, E8–99. "Standard test methods for tension testing of metallic materials." *Annual book of ASTM standards. ASTM* (2001).

[100] Creep Data Sheet No. 13B, " Creep data sheet no. 13B SUS 403-B (12Cr)", NIMS, 1994, https://smds.nims.go.jp/MSDS/pdf/sheet/C13BJ.pdf

[101] Yoo, Y. S., C. Y. Jo, and C. N. Jones. "Compositional prediction of creep rupture life of single crystal Ni base superalloy by Bayesian neural network." *Materials Science and Engineering: A* 336.1-2 (2002): 22-29.

[102] Li, Xiang-Wei, et al. "Effect of Aging Heat Treatment on the Microstructure and Creep Properties of the Cast Ni-Based Superalloy at Low Temperature." *Acta Metallurgica Sinica (English Letters)* 32.5 (2019): 651-658.

[103] Metallic Material Database (Kinzoku), "Kinzoku.", NIMS, 2011, https://metallicmaterials.nims.go.jp/index.html.

[104] Creep Data Sheet (CDS), " Creep datasheet", NIMS, 1972, https://smds.nims.go.jp/creep/index.html.

List of Publications

Journals

[1] Hwang, Jaekyun, Yasunobu Ando, and Satoshi Watanabe. "Materials Search of Perovskite Cathode in SOFC by Statistical Analysis." *ECS Transactions* 68.1 (2015): 549.

[2] Hwang, Jaekyun, et al. "Prediction of viscosity behavior in oxide glass materials using cation fingerprints with artificial neural networks." *Science and technology of advanced materials* 21.1 (2020): 492-504.

Domestic Conferences

[1] J. Hwang, 安藤康伸, 渡邉聡、「SOFC のペロブスカイトカソードにおける酸素交換レートとバルクイオン伝導率・電子伝導率との相関に関する統計学的解析」、『第 40 回固体イオニクス討論会』、2014 年 11 月 18 日、口頭。

[2] J. Hwang, 安藤康伸, 渡邉聡、「ペロブスカイト型カソードの伝導率の第一原理計算による評価」、『第 41 回固体イオニクス討論会』、2015 年 11 月 25 日、口頭。

International Conferences

[1] J. Hwang, Y. Ando, and S. Watanabe, “Materials Search of Perovskite Cathode in SOFC by Statistical Analysis”, ECS Conference on Electrochemical Energy Conversion & Storage with SOFC-XIV, Glasgow, UK, Jul. 25, 2015, Oral.

[2] J. Hwang, Y. Ando, and S. Watanabe, “Ionic and Electronic Conductivity Calculations of 100 Perovskite Oxides for SOFC Cathode Materials” The 19th Asian Workshop on First-Principles Electronic Structure Calculations, Hsinchu, Taiwan, Nov. 1, 2016, Poster.

[3] J. Hwang, Y. Tanaka, S. Ishino, and S. Watanabe, “Artificial Neural Network Modeling to Predict Viscosity of Oxide Glasses” Asian Consortium on Computational Materials Science – Virtual Organization 11, Sendai, Japan, Dec. 19, 2016, Oral.

[4] J. Hwang, Y. Tanaka, S. Ishino, and S. Watanabe, “Neural Network Modeling to Predict Viscosity of Oxide Glasses” Tsinghua University – the University of Tokyo Joint Symposium on Multidiscipline, Beijing, China, Apr. 15, 2017, Poster.

[5] J. Hwang, and S. Watanabe, “Prediction of solid crystal properties from its chemical composition”, ACCMS-Theme Meeting on Multiscale Modelling of Materials for Sustainable Development 2018, Hanoi, Vietnam, Sept. 7, 2018, Oral.

[6] J. Hwang, and S. Watanabe, “Predicting Material Properties Using a Novel Descriptor “Elemental Fingerprints” with Neural Networks”, 2019 MRS Spring meeting & exhibit, Phoenix, USA, Apr. 23, 2019, Poster.

[7] J. Hwang, and S. Watanabe, “Prediction of Material Properties Using A Novel Descriptor “Elemental Fingerprints” Made from Chemical Formula”, ICMAT 2019, Marina Bay Sands, Singapore, Jun. 24, 2019, Oral.

[8] J. Hwang, and S. Watanabe, “Predicting Chemical and Electronic Properties Using a Novel Descriptor “Elemental Fingerprints set” with Neural Networks” NIMS WEEK 2019, Tokyo, Japan, Oct. 30, 2019, Poster.

[9] J. Hwang, and S. Watanabe, “Prediction of Material Properties from Its Chemical Formula Using Novel Descriptor “Elemental Fingerprints” with Neural Networks”, MRM 2019, Yokohama, Japan, Dec. 11, 2019, Oral.

Acknowledgements

I am grateful to the many people who supported me. I would first like to thank my supervisor, Professor Satoshi Watanabe. I am pleased that he offers me to freedom of pioneering a new direction. He encouraged and guided me to think about the inventive and original idea. He also consistently allowed this dissertation to be my own work but steered me in the right direction whenever he thought I needed it. I would also like to thank all the other laboratory members who were spending time together. I have been blessed with a wonderful academic environment in the Watanabe laboratory.

I am also thankful to five advisors, Prof. Emi Minamitani, Prof. Koji Tsuda, Prof. Mizoguchi Teruyasu, Prof. Shoichi Nambu and Prof. Junya Inoue. They provided me invaluable advice that has greatly improved the quality of my dissertation. I also thank Dr. Yasunobu Ando for giving me a lot of advice especially in my early days of graduate school.

I also appreciate the MERIT program（東京大学 統合物質科学リーダー養成プログラム）to support me financially and academically. Special thanks to Yuta Tanaka and Seiichiro Ishino, who the MERIT members at the same period as well as the co-authors of a paper published in the journal STAM. They gave me great help in realizing my idea.

Finally, I would like to express my very profound gratitude to my family, to my grandmother, to my parents, and to my younger brother for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this dissertation. This accomplishment would not have been possible without them. Thank you.

February 2021

黄　滋鈞