The prediction of molecule atomization energy using neural network and extreme gradient boosting

M Sumanto\textsuperscript{1}, M A Martoprawiro\textsuperscript{2}, A L Ivansyah\textsuperscript{3}

\textsuperscript{1} Master Program in Computational Science, Faculty of Mathematics and Natural Sciences, Bandung Institute of Technology, Bandung 40132, Indonesia
\textsuperscript{2} Department of Chemistry, Faculty of Mathematics and Natural Sciences, Bandung Institute of Technology, Bandung 40132, Indonesia
\textsuperscript{3} Department of Computational Science, Faculty of Mathematics and Natural Sciences, Bandung Institute of Technology, Bandung 40132, Indonesia

E-mail: majusumanto@students.itb.ac.id

Abstract. Machine Learning is an artificial intelligence system, where the system has the ability to learn automatically from experience without being explicitly programmed. The learning process from Machine Learning starts from observing the data and then looking at the pattern of the data. The main purpose of this process is to make computers learn automatically. In this study, we will use Machine Learning to predict molecular atomization energy. From various methods in Machine Learning, we use two methods namely Neural Network and Extreme Gradient Boosting. Both methods have several parameters that must be adjusted so that the predicted value of the atomization energy of the molecule has the lowest possible error. We are trying to find the right parameter values for both methods. For the neural network method, it is quite difficult to find the right parameter value because it takes a long time to train the model of the neural network to find out whether the model is good or bad, while for the Extreme Gradient Boosting method the time needed to train the model is shorter, so it is quite easy to find the right parameter values for the model. This study also looked at the effects of the modification on the dataset with the output transformation of normalization and standardization then removing molecules containing Br atoms and changing the entry in the Coulomb matrix to 0 if the distance between atoms in the molecule exceeds 2 angstrom.

Keywords: Atomization Energy, Extreme Gradient Boosting, Machine Learning, Molecule, Neural Network.

1. Introduction

Atomization energy of molecules plays an important role in the world today especially on compound design in chemical and pharmaceutical industries. Nowadays to find out the amount of molecule atomizing energy requires a long computation time, maybe a few days, several weeks, or even several months. So we have the idea to use the machine learning model to predict the atomization energy value of a molecule so that it can reduce computational time and also save costs needed for the computational process.

Throughout this paper we focus on how a machine learning model works so that it can predict the energy value of molecular atomization and finding the right parameter values for a Machine Learning model with purpose to minimize the prediction error of the molecule atomization energy value.
In this work we show how significant the improvement of the accuracy results after some of feature engineering and also we review some of machine learning techniques. Our best results reduce the error of prediction from root mean square error (RMSE) around 700 to around 0.0375.

2. Method
This dataset is extracted from the PubChem Substance and Compound database and has a Substance Identifier Number or SID from 1 to 250000 and provided by previous study [1]. This dataset has information about the atomic coordinates in the Cartesian plane of various molecules, the nature of the molecules indicated by shapeM, and the atomisation energy of the molecules [1]. This dataset has information about the molecule that consists of atoms H, C, N, O, F, Si, P, Cl, Br, and I. The maximum number atoms in molecule is 50 atoms and the minimum number atoms in molecule is 2 atoms [1]. The number of molecules at this time is 144032. In the figure 1 we can see the energy distribution of the dataset we have. Mean and standard deviation of the atomizing energy data that we have are 42.6012 and 27.7913, respectively.

![Figure 1: A histogram of atomization energy.](image)

After the data extraction process from the PubChem Substance and Compound database is complete, We know that our data can be converted using previous study [2, 3] with Coulomb matrix. So the data is processed into a Coulomb matrix. Coulomb Matrix is a matrix that has information regarding the charge of the atom ($Z_i$) and the coordinates of the atom ($R_i$).

Following is the equation for determining the entry value in the Coulomb matrix $C$:

$$C_{ij} = \begin{cases} 
0.5Z_i^2 & \forall i = j, \\
\frac{Z_iZ_j}{|R_i-R_j|} & \forall i \neq j. 
\end{cases}$$ (1)

The diagonal element of the Coulomb matrix shows the potential energy of the free atom and the non-diagonal element of the Coulomb matrix shows the repulsion energy of 2 different atoms. Two problems with using the Coulomb matrix as a molecular representation are:

1. The dimensions of the Coulomb matrix depend on the number of atoms in a molecule.
2. The order of the atoms in the Coulomb matrix is undefined so that many numbers of Coulomb matrices can represent a molecule simply by changing the row or column of the Coulomb matrix.
The first problem can be solved by using "invisible atoms" in a molecule [2, 3]. "The invisible atom" has a charge of 0 so it does not interact with another atoms. This results in "invisible atoms" not affecting other atoms in a molecule. So that we can make the number of atoms in all molecules in the atom to be constant, which is equals to 50. This results in all Coulomb matrices that we have have the same dimensions which is $50 \times 50$ [5, 6]. The second problem is more difficult to solve and also there is no solution that makes sense physically so in this study we use 2 representations namely:

1. Representation according to the atom order in the PubChem Database. This representation simply converts the coordinate data of atoms into entries in the Coulomb matrix according to the order of the atoms in the database or we can call it Non-Sorted Coulomb Matrix
2. If we look further into Non-Sorted Coulomb Matrix. This is a symmetry matrix when $a_{ij}$ is Non-Sorted Coulomb Matrix then applies $a_{ij} = a_{ji}$. So that in the Coulomb matrix we have $i \neq j$ then the input value $a_{ij}$ will appear twice. Then the idea arose to input only a portion of entries from the a matrix by inputting $a_{ij}$ with $j \geq i$. So that we can reduce the amount of data features in the data that have initially 2500 inputs to 1275 inputs.

Then some conditions are added to the extraction dataset, namely:

1. Removes all molecules containing Br atoms, because Br atoms is coming from different group than other elements.
2. For atoms a and b, where $a \neq b$ if the distance between atoms a and b exceeds 2.0 angstroms, the entry for a, b in the Coulomb matrix becomes 0. This is due to no bonding between atoms a and b.

For each machine learning model, we validate using 5-fold cross validation method. The Multilayer Neural Network model used to predict the value of atomic energy has 2500 inputs or 1275 inputs. In this study, the number of nodes used in the hidden layer varies from 1500 nodes to 1000 nodes for the first hidden layer and 750 nodes to 200 nodes for the second hidden layer. Then for the initial weight used in the Machine Learning model is $W_0 \sim \mathcal{N}(0, \frac{1}{\sqrt{m}})$ where $m$ is the number of nodes in the hidden layer.

In this study the learning rate used is quite varied starting from the constant learning rate, decreasing learning rate, increasing learning rate but after all we choose [4]. The activation function chosen is the sigmoid function where $f(x) = \frac{1}{1+e^{-x}}$ and the tanh function where $f(x) = \tanh x$. Then after several experiment, we deciese to choose 1000 nodes on the first hidden layer and 300 nodes on the second hidden layer. After that we choose tanh as activation function. Extreme Gradient Boosting model which is used to predict the value of atomization energy has several parameters that must be set in order to minimize the error value between the atomization energy data and the prediction of atomization energy. These parameters are $\text{max_depth}$, $\text{n_tree}$, $\text{learning_rate}$, $\text{colsample_size}$, and $l_1$ or $l_2$.

The method used to find the right parameters in this study is the coordinate descent method, which is a method by making 1 hyperparameter changeable and other hyperparameters constant. Then when it has found a hyperparameter that produces an optimal Root Mean Square Error (RMSE) then do the same method for other hyperparameters. So initially we chose to optimize $\text{learning_rate}$ with initial hyperparameter $\text{max_depth} = 5$, $\text{colsample_size} = 0.7$, $l_2 = 1$, and $\text{learning_rate} = 0.08, 0.04, 0.02, 0.01, 0.005$ so that the best $\text{learning_rate}$ value obtained is $0.01$, $\text{max_depth} = 7$, $\text{colsample_size} = 0.7$, and $l_2 = 1$.

We're using standardization and normalization on the target variable because the magnitude of the target value is greater than predictor and also the target variable has a big value for mean and standard deviation. So the formula for standardization is:

$$y_{old} = \frac{y_{new} - \mu}{\sigma},$$  \hspace{1cm} (2)
and the formula for normalization is:

$$y_{new} = \frac{y_{old} - y_{minimum}}{y_{maximum} - y_{minimum}}$$  \hspace{1cm} (3)

These two techniques are usually used in regression and classification when the data we have has different units.

3. Results and Discussions

For each result on the neural network model and the decision tree, the cross-validation method was tested. Following are the results of the neural network model with dataset A containing Coulomb matrix data without modification, dataset B containing Coulomb matrix data without molecules containing Br atoms, and dataset C containing Coulomb matrix data without molecules containing Br atoms and if the distance between atoms is more from 2.0 angstroms, the entry value in the Coulomb matrix becomes 0:

| Algorithm | Matrix Type | Dataset | RMSE     | Comp. Time |
|-----------|-------------|---------|----------|------------|
| N. Network | Original Coulomb | A    | 766.256  | 20475 s    |
| N. Network | Half Coulomb | A    | 770.698  | 11177 s    |
| N. Network | Original Coulomb | B    | 754.631  | 18252 s    |
| N. Network | Half Coulomb | B    | 772.211  | 7504 s     |
| N. Network | Original Coulomb | C    | 772.211  | 14604 s    |
| N. Network | Half Coulomb | C    | 772.211  | 7143 s     |

Table 1: The approximation results using machine learning without transformation.

Table 2: The approximation results using machine learning with standardization output. The RMSE below already multiplied by $\sigma = 27.7852$.

| Algorithm | Matrix Type | Dataset | RMSE     | Comp. Time |
|-----------|-------------|---------|----------|------------|
| N. Network | Original Coulomb | A    | 7.7909   | 18428 s    |
| N. Network | Half Coulomb | A    | 8.5133   | 10060 s    |
| N. Network | Original Coulomb | B    | 7.1269   | 16427 s    |
| N. Network | Half Coulomb | B    | 7.3491   | 6754 s     |
| N. Network | Original Coulomb | C    | 8.5772   | 13144 s    |
| N. Network | Half Coulomb | C    | 8.7717   | 6429 s     |

Table 3: Result of Neural Network Approximation with Output Normalization. The RMSE below already multiplied by $max_{target}$ and $min_{target}$ equals to 381.6295.

| Algorithm | Matrix Type | Dataset | RMSE     | Comp. Time |
|-----------|-------------|---------|----------|------------|
| N. Network | Original Coulomb | A    | 0.8777   | 18340 s    |
| N. Network | Half Coulomb | A    | 0.9922   | 10064 s    |
| N. Network | Original Coulomb | B    | 0.0022   | 16425 s    |
| N. Network | Half Coulomb | B    | 0.8395   | 6496 s     |
| N. Network | Original Coulomb | C    | 1.2593   | 13077 s    |
| N. Network | Half Coulomb | C    | 1.2593   | 6443 s     |
As we know that in the training process, the error value in the neural network model is very volatile. This is caused by the minibatch method used during the training process to avoid the gradient descent method in our model being trapped at the local minimum value. During the learning process we have divided our test data into smaller partitions. Of course the one partition and the other partitions have different complexity so that it also affects the value of the error which is very volatile.

Then by comparing the results at the table 1 and table 2, we can conclude the neural network model and the standardization method on the target variable has succeeded in increasing the accuracy of the model we have made. This is because the target variable has a high value of $\sigma$ and $\mu$ with $\sigma_{\text{target}} = 27.7852$, $\mu_{\text{target}} = 42.5980$, $\min_{\text{target}} = -99.0292$, and $\max_{\text{target}} = 282.6003$ while the neural network model that we made has a fairly small initial weight of $w \sim \mathcal{N}(0,1)$ and uses a tanh activation function with a range of values of $[-1,1]$.

Another thing that causes standardization is so influential is 95 percent of the predictor variables have a mean value close to 0 and a standard deviation close to 0 while the target variable has a high average value and standard deviation. But after standardization we get such a good RMSE value, we must multiply the RMSE value by the standard deviation value of the training data because the purpose of the model that we created is to predict the actual value of the atomization energy of the molecule. Table 2 and table 3 are tables of results from neural networks using standardized output after being multiplied by the standard deviation of $\sigma_{\text{target}} = 27.7852$ and the result table of normalized output after multiplied by the difference between $\max_{\text{output}}$ and $\min_{\text{output}}$ which is equals to 381.6285.

We can see the result at the figure 2 if we use a dataset without modification as training data, the accuracy we get is not good. If we look at figure 3, figure 5, and figure 7 then the model of training results with standardized output that has been multiplied by the standard deviation, the results obtained are quite satisfactory, but when compared with the results of training with normalized output that has been multiplied by the difference between the maximum output value and the minimum output value at figure 4, figure 6, and figure 8, the normalization results are far more satisfying. This proves that in addition to setting hyperparameter values, the process of preparing data prior to training is crucial in achieving better results. In this case the transformation of the output value is proven to make it easier for our model to study the data that we have prepared. The error value of the prediction of atomizing energy obtained by combining neural networks or Extreme Gradient Boosting with normalization is quite good. If we want to achieve better results, there are 2 solutions that allow this to be achieved, namely increasing the number of epochs, even though it results in longer computing time. But keep in mind also that we must be careful when increasing the number of epochs because it allows overfitting on the model we have. Overfitting is an event when the model we have has very good performance on training data but poor performance on test data.

If we compare each XGB model and neural network in terms of training time, the modification of the data set, ie. removing molecules containing Br atoms, reduces computational time by 5% − 10%. This is due to the reduced amount of training data. If we then make an entry from the Coulomb matrix to 0 provided that the distance between 2 atoms exceeds 2 angstroms then the computation time is reduced by 40% − 60%. This is caused by the complexity of the data being reduced because the input data that was not previously changed to 0.

4. Conclusion
In this study we can see neural network model and extreme gradient boosting are powerful tools to predict atomization energy and might contribute to compound design in chemical or pharmaceutical industries. All machine-learning algorithms used in this research achieve the first goal of making quantum-chemical computation a matter of milliseconds rather than hours or days when using ab initio calculations.
Figure 2: RMSE results for regression trees with the number of decision trees changing, for every 1 \( T \) it is equal to 100 decision trees with depth = 7.

Figure 3: RMSE results for regression trees with the number of trees that change, for every 1 \( T \) it is equal to 100 trees with a dataset without modification whose output values have been standardized.

Figure 4: RMSE results for regression trees with the number of trees changing, for every 1 \( T \) it is equal to 100 trees with a normalized dataset and without other modifications.
Figure 5: RMSE results for regression trees with the number of trees changing, for every 1 T unit it is equal to 100 trees with a dataset without molecules containing Br atoms and the output value has been standardized.

Figure 6: RMSE results for regression trees with the number of trees changing, for every 1 T it is equal to 100 trees with normalized dataset and without molecules containing Br atoms.

Figure 7: RMSE results for regression trees with the number of trees changing, for every 1 T it is equal to 100 trees with dataset without molecules containing Br atoms and the output value has been standardized as well as changing the Coulomb matrix entry value.
With respect to prediction accuracy, our results using Neural Network improve more than 600 RMSE than our results using Extreme Gradient Boosting or XGB. After we’re using standardization at the target variable, both models are improve so the RMSE decreasing. After the standardization the neural network model the RMSE decreasing from around 700 to around 7. For the extreme gradient boosting model the RMSE decreasing from range 9 to 12 to range 0.38 to 0.63. If we’re using normalization at the target variable, both models are improve better than before so for the Neural Network model the RMSE decreasing from 700 to around 0.8. Also for the extreme gradient boosting model the RMSE decreasing from range 9 to 12 to range 0.0375 to 0.075. We can conclude without any transformation at the target variable the Neural Network model perform better than the XGB model. After the transformation at the target variable, the Extreme Gradient Boosting model or XGB model perform better than the Neural Network model. The normalization at the target variable has a better effect than standardization at the target variable.

Afterwards the cause of normalization has a better effect that standardization because the distribution of target variable not follow the Gaussian distribution. From the results, we know that nothing changes after removing the Br atom. We can conclude removing the Br atom doesn’t change the complexity of this problem. In the future maybe we can try to remove another atom to simplify this problem to get a better accuracy. If we compare from the matrix type then we can conclude that half coulomb matrix perform better than full or original coulomb matrix from the perspective computational time. From the perspective accuracy the full coulomb matrix perform better than half coulomb matrix. So we can conclude, the half coulomb matrix decreasing the complexity of this problem but reduce some of information to the model that we train.

Finally the ML methods give us a good accuracy and fast to the task of atomization energy prediction, but feature engineering plays an important role in achieving good results. In the future we’ll focus on methods that are not only fast and good at accuracy but also make it easier for us to understand the behavior of atoms.

5. Acknowledgment
The authors would like to thank Carlos Ansotegui from Department of Informatics from Universidad de Lleida for his large contributions. The simulations were conducted in Universidad de Lleida at Informatics HPC Lab.
References
[1] Burak Himmetoglu 2016 Tree Based Machine Learning Framework for Predicting Ground State Energies of Molecules (Journal of Chemical Physics 145, 134101)
[2] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld. 2013 Correlations in the Motion of Atoms in Liquid Argon Machine Learning of Molecular Electronic Properties in Chemical Compound Space New Journal of Physics 15.9 p. 095003
[3] G. Montavon, Katja Hansen, Siamac Fazli, Matthias Rupp, Franziska Biegler, Andreas Ziehe, Alexandre Tkatchenko, O. Anatole von Lilienfeld. Klaus-Robert Muller. 2012 Learning Invariant Representations of Molecules for Atomization Energy Prediction. In Advances in Neural Information Processing Systems 440-448
[4] Daniel Graupe. 2013 Principles of Artificial Neural Network
[5] Katja Hansen, Grégoire Montavon, Franziska Biegler, Siamac Fazli, Matthias Rupp, Matthias Scheffler, O. Anatole von Lilienfeld, Alexandre Tkatchenko, and Klaus-Robert Müller 2013 Assessment and Validation of Machine Learning, Methods for Predicting Molecular Atomization Energies.
[6] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld. 2012 Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning Physical review letters 108.5 p. 058301