Molecular structure prediction based on graph convolutional networks

Xiaohui Lin¹, Yongquan Jiang¹², Yan Yang¹²
1 School of Computing and Artificial Intelligence, Southwest Jiaotong University
2 Institute of Artificial Intelligence, Southwest Jiaotong University

Abstract: Due to the important application of molecular structure in many fields, calculation by experimental means or traditional density functional theory is often time consuming. In view of this, a new Model Structure based on Graph Convolutional Neural network (MSGCN) is proposed, which can determine the molecular structure by predicting the distance between two atoms. In order to verify the effect of MSGCN model, the model is compared with the method of calculating molecular three-dimensional conformation in RDKit, and the result is better than it. In addition, the distance predicted by the MSGCN model and the distance calculated by the QM9 dataset were used to predict the molecular properties, thus proving the effectiveness of the distance predicted by the MSGCN model.

Key words: molecular structure; Graph convolutional neural network; Atomic distance.

1 Introduction

In recent years, with the rapid development of machine learning, more and more machine learning method has been applied to materials and chemical, and other fields, mainly including the material of crystal system prediction, prediction of high entropy alloys phase and so on. These works have achieved certain success, and greatly reflects the advantages of machine learning method in quantum chemistry calculation. As a result, machine learning is highly expected to be a tool to accelerate the discovery of molecules or materials with specific properties, and machine learning is also considered as an innovative mode of material development [1]. As a result, a lot of work is devoted to using machine learning methods to predict the properties of molecules or materials. For example, Song[2] et al. proposed a machine learning model to predict non-centrosymmetric crystalline materials by combining different features and different machine learning algorithms.[3] the neural network with regularization is used to predict the phase of the high entropy alloy; Schutt et al. proposed a SchNet model [4], which is a neural network that uses continuous filtering convolutional layer to learn the interaction of atoms at any position in a molecule, and can well predict the properties of molecules. But, in fact, the structure of the material to a large extent affected the nature of it, and in many chemical simulation theory experiment, such as the structure of the materials are playing a very important role, the specific structure of predict the new material does not yet exist, to better study the nature of it is very useful and carries on the synthesis and help speed up the discovery of new materials. At present, the molecular geometric structures of the traditional methods are usually obtained by the experimental method for determining or by calculation, experimental methods including microwave spectrum, X-ray diffraction, electron
diffraction and neutron diffraction method, which used in gas electron diffraction of electron diffraction (gas phase) is to put the gas molecules onto the diffraction in the cavity, accelerated after the electronic diffraction and measured data. Interatomic distance matrix is obtained by measuring the diffraction data, also is equivalent to the molecular structure, but the experimental methods are often complex operation, some still need high cost, the traditional calculation method of calculation and optimization of the main use of density functional theory, and the calculation method of the high precision of calculation costs are high, although some quick calculation method on the basis of experience faster, But the calculation results are rather crude. At present, some work has been done to predict the crystal structure, and some progress has been made. For example, the artificial bee colony algorithm is used to predict the complex crystal structure in [5]. However, the research work on molecular structure prediction is still less.

The characterization of molecules has a direct impact on prediction work. The form of graphs is a natural representation of molecules. In previous work, because graphs are difficult to process, molecular fingerprints are often used to characterize molecules as the input of neural network. Since 2018, Battaglia proposed graph network [6], people gradually discovered the powerful representation ability of graph, graph neural network also showed good performance in many works, such as natural Language processing, traffic flow prediction and other fields, and now graph neural networks have also been successfully applied to the field of molecules and materials, and more and more work uses graph neural networks to process molecules and crystals. Xie et al. proposed a CGCNN model [7]. They mainly represented the crystal structure by constructing a crystal graph, and used a general graph convolutional neural network model to automatically extract the most suitable feature representation for the prediction target. Chen et al. developed a graph network model called MEGNet [8], which accurately predicts the properties of molecules and crystals. Louis et al. proposed a new GATGNN [9] to solve the problem that many GNN models could not effectively distinguish the contributions made by different atoms when making property prediction. This model learns the local relationship between adjacent atoms and the contribution of atoms to the whole material property through the graph attention layer and the global attention layer, and achieves a very good prediction performance. However, most of these works use graph neural networks to predict the properties of molecules or crystals, and few studies on the use of graph neural networks to predict molecular structures. In view of this, this paper proposes a new graph convolutional neural network model (MSGCN) for molecular structure determination. This model takes the Simplified Molecular Input Line Entry Specification (Smiles) of the molecule as the input to construct the molecular graph, and the atomic distance matrix of the molecule is predicted by the model, so that the specific structure of the molecule is obtained.

The main contributions of this paper are as follows : (1) A new graph convolutional network model for molecular structure prediction is proposed; (2) A comparative experiment was conducted to explore and analyze the influence of different molecular differences on the model; (3) Using the predicted structure to predict the molecular properties, to prove the effectiveness of the method.

### 2 Method
2.1 Encoding

In this paper, the molecule is defined as a graph $G = (V, E)$, where the graph node $V$ represents the atom and the edge $E$ represents the bond. The eigenvector of the atom is defined as $v, v \in V$, and the eigenvector of the bond is defined as $e, e \in E$, where $v$ is encoded as a 19-dimensional eigenvector and $e$ as a 4-dimensional eigenvector. The specific vector composition is shown in Table 1.

| Vector name | Encoding description |
|-------------|-----------------------|
| **atom**    |                       |
| Molecular composition | 5 - dimensional vectors, the number of atoms of each element |
| atomic type | 5 - dimensional vector, one-hot coding of CNOFH |
| atomic number |                       |
| covalent radius |                       |
| electron arrangement | 2 - dimensional vector |
| electronegativity |                       |
| First ionization energy | The Log transformation |
| atomic radius |                       |
| atomic orbital | 2 - dimensional vector |
| **bond**   |                       |
| bond type | 4 - dimensional vector, one-hot coding of single bond, double bond, triple bond and aromatic bond |

2.2 Model

The MSGCN model proposed in this paper takes the graph constructed according to the input molecule SMILES as the input of the model, and the overall frame diagram of the model is shown in Figure 1. Due to the need to predict the molecular distance matrix, so when building the input graph, connecting the atoms which don’t have chemistry bond between them is necessary, namely to construct a complete graph. But complete graph destroyed the original structure of molecular graph information, therefore, the model is set up two branches to process the complete graph and the molecular graph, respectively. The branch where the molecular graph is located needs to accomplish two things: one is to predict the distance between bonded atoms, and the other is to send the molecular graph into the update layer with graph convolution operation [11] for node updating. After the update, the eigenvectors of all nodes are assigned to the nodes in the complete graph, which uses the updated nodes to complete the update of the edges between atoms without bonds. After going through L1 of these update layers, the corresponding edges after each updating are concatenated to predict the distance of atoms without bonds.

A multi-layer perceptron containing L2 hidden layers is used to predict the distance of atoms [12], which is defined as follows:

$$ e_i = \varphi(W_i \ast e_{i-1} + b_i) $$ (1)
Where, $e_i$ represents the output vector at the $i$-th layer, $W$ is the trainable weight parameter, $b$ is the bias, and $\phi$ is the activation function, which makes the function non-linear for better approximating the real value. ReLU is used as the activation function here.

Figure. 1 Model Structure

2.3 Update layer

In the update layer, the main purpose is to update nodes and edges. The main operation is graph convolution operation. For each node in the molecular graph, its neighbor nodes are aggregated to update itself. The aggregation operation is shown in Formula 2:

$$h_{N(i)}^{(l+1)} = \text{aggregate}\{h_j^l, \forall j \in N(i)\}$$

(2)

Where, $N(i)$ represents all the neighbor nodes of node $i$, $l$ represents the number of graph convolution layers, and aggregate represents the aggregation method. In this paper, the average value is used as the aggregation method. After the aggregation information of neighbors is obtained, it is concatenated together with node $i$, sent into the full connection layer, and the updated node features of node $i$ are output. The node update formula is shown in Formula 3:

$$h_i^{(l+1)} = \phi( f( h_i^l \| h_{N(i)}^{(l+1)} ))$$

(3)

Where $\|$ represents the operation of concatenation, $\phi$ means activation function ReLU, $f$ means full connection layer. When all nodes in the molecular graph are updated, new node features are assigned to nodes in the complete graph, and then edges between atoms without bonds in the complete graph are updated. The operation is shown in Formula 4:

$$e_{ij} = \phi( f( v_i^r \| v_j^s ))$$

(4)

Where, $e_{ij}$ represents the eigenvectors of the $i$-th edge, and $v_i^r$ and $v_j^s$ represent the eigenvectors of the two nodes forming the edge. When all edges are updated, the updating layer ends. After passing through L1 such updating layers, the edge feature vectors obtained in each layer and the node features forming the edge in the last layer are concatenated together and sent into the

Update Layer

Input

Preprocessing

MLP1

MLP2

Output

MLP2

Concatenation

Edge

Preprocessing

Input

MLP1

Output

Figure. 1 Model Structure
multi-layer perceptron to predict the length of the edge. The operation is shown in Formula 5:

\[ e = \varphi( f( e^0 || e^1 || \ldots || e^L_R || e^L_S )) \]  

(5)

Where \( e \) represents the predicted distance between all atoms without bonds, \( e^i \) means the edge features after \( i \)-th update layer, and \( e^L_R \) and \( e^L_S \) represent the feature vector sets of the two nodes that constitute each edge in the last update layer.

It is worth mentioning that after the convolution operation of each update layer, the batch normalization layer (BN layer) is used to prevent overfitting and gradient disappearance.

### 2.4 Model Constructing and Training

The model was built based on PyTorch framework. In order to achieve rapid convergence of the model, multiple molecular maps were formed into a batch for training. The batch size was 2048, and a total of 200 epochs were trained. Adam optimizer was used to optimize gradient descent in the model, and the learning rate was 0.01. In the later stage, the value of learning rate would be dynamically adjusted in order to achieve better convergence. The total loss function of the whole model is defined as Formula 6:

\[ \text{Loss} = \alpha \text{loss1} + \beta \text{loss2} \]  

(6)

Loss1 represents the loss of MLP1 and Loss2 represents the loss of MLP2. Through the experiment, \( \alpha \) and \( \beta \) were set as 0.4 and 0.6 respectively, and the mean absolute error was used for both of these losses, which are defined as:

\[ \text{loss} = \frac{1}{n} \sum_{i=1}^{n} | y_{i}^{\text{label}} - y_{i}^{\text{predict}} | \]  

(7)

where \( n \) is the number of edges in each batch, \( y_{i}^{\text{label}} \) represents the label, and \( y_{i}^{\text{predict}} \) represents the predicted value calculated by the model.

The model was trained on Intel(R)Core(TM) i7-4710MQCPU. Figure 2 shows the decline curve of the loss function when using a data set less than 11 atoms for training. The MSGCN model can converge within 100 epochs, while with the increase of data volume, the model can converge within fewer epochs.

![Figure 2 Loss decline curve](image)
3 Result

3.1 Dataset

In this paper, we use the QM9 dataset to train and test our proposed model. The molecules in the dataset consist of elements such as C, N, O, F, H and contain up to 9 heavy atoms. The atomic coordinates in the data set are first generated by the initial Cartesian coordinates of Corina's parse molecule Smiles. Then MOPAC is used to perform geometric relaxation at the semi-empirical theoretical level of PM7, and the results are used as the input to the geometric relaxation of Gauss B3LYP [10] to obtain the optimized atomic coordinates. There are 133885 molecules in the dataset, and the number distribution of atoms in the constituent molecules is shown in Figure 3. After removing the molecules containing incorrect data, the remaining 131,808 molecules are used as the data set of this paper.

![Figure 3 Atomic distribution](image)

In order to prove the effect of the model, the following experimental schemes were implemented, in which both the training set and the test set were 9:1.

3.2 Compare to the ETKDG methods in RDKit

In order to study the influence of atomic size difference in the training set on the model, the subset composed of molecular data with atomic number less than 11, 15 and 20 was screened out from the QM9 data set and all the data were taken as the data set to train the model. In addition, in order to illustrate the advanced nature of the method, we compared our model with the method used in the RDKit chemical calculation package to calculate the geometric structure of atoms [13]. The results are shown in Table 2, where the average time of the method used in RDKit to calculate a molecule is 0.0069 seconds, and the average time of the MSGCN model to calculate a molecule is 0.0160 seconds. RDkit methods are mainly based on the distance of the method and the method based on knowledge (ETKDG) The distance-based method is to generate the optimized molecular boundary matrix, randomly generate the distance matrix according to the boundary matrix, and then map the distance matrix to the three-dimensional space to generate atomic coordinates. Finally, the force field is used to roughly optimize the atomic coordinates [14]. The molecular structure generated by this method is relatively rough, and it needs to be optimized by force field again, while the knowledge-based method is based on some rules summarized by Riniker et al.
from the small molecular structure of crystal structure database to modify the results obtained by the distance-based method [15].

Table.2 Comparison of training results between RDKit and MSGCN in training sets with different atomic size differences (Unit: Å)

| Maximum number of atoms | Size of dataset | Method   | MAE   | RMSE  |
|-------------------------|-----------------|----------|-------|-------|
| 11                      | 2075            | ETKDG    | 0.1466| 0.2433|
|                         |                 | Ours     | 0.0962| 0.2270|
| 15                      | 25706           | ETKDG    | 0.2340| 0.3960|
|                         |                 | Ours     | 0.1390| 0.3470|
| 20                      | 103844          | ETKDG    | 0.3194| 0.5239|
|                         |                 | Ours     | 0.1636| 0.4271|
| all                     | 131808          | ETKDG    | 0.3480| 0.5630|
|                         |                 | Ours     | 0.1749| 0.4525|

As you can see from Table 2, with the increasing difference of the number of atoms in the dataset, whether it is ETKDG in RDKit or our model, the errors generated are gradually increasing. This is because as the number of atoms increases, the distance between atoms without bonds also increases, making the model more difficult to approximate. In addition, in molecules with a large number of atoms, the distance between atoms is already larger, which also leads to a larger prediction error. However, it can be seen that our error is generally smaller than ETKDG method, and with the increasing difference in the number of atoms, the error growth range of ETKDG is larger than that of our model, which also indicates that our model has better robustness compared with ETKDG.

To show the effect of the model more visually, the predicted results of interatomic distance with the error around the mean error were selected from the test set of less than 11 atoms and compared with the ground truth. As shown in the Figure 4, the distance between atoms with bonds is relatively small and very close to the ground truth, while the distance error between atoms without bonds is relatively large, but the trend is basically consistent with the ground truth.

![Figure.4 Prediction of interatomic distance; (a) the distance between atoms with bonds; (b) the distance between atoms without bonds](image)

In addition, in order to facilitate the observation of the distance prediction error distribution between atoms with bonds, we randomly selected two thousand molecules from the test set...
composed of molecules with atomic number less than 20, and calculated the absolute error of the
distance between atoms with bonds. The statistical results are shown in Figure 5. Most errors are
concentrated between 0 and 0.01 Å, with an average error of 0.021 Å. It can be seen that the
model is more accurate in predicting the distance between bonded atoms.

![Figure 5](image)

**Figure.5 Distance prediction error distribution between bonded atoms**

### 3.3 Ablation experiments

During training, the model used molecular components as node features to expect the model to
learn some global information and improve the prediction results. In addition, the model added the
BN layer after the convolution operation of the update layer to prevent over-fitting and gradient
disappearance. To explore the role of these two components, we performed ablation experiments
on the dataset in which the number of atoms in a molecule is less than 15. We respectively trained
the model with a BN layer and the molecular components vector was added to the node feature
vector (model with both), the model without a BN layer and the molecular components vector was
added to the node feature vector (model with MC), the model with a BN layer and the molecular
components vector wasn’t added to the node feature vector (model with BN) and the model
without these two part (Model without BN and MC). The results are shown in Table 3.

| Model                  | MAE   | RMSE  |
|------------------------|-------|-------|
| Model with both        | 0.1390| 0.3470|
| Model with MC          | 0.1497| 0.3655|
| Model with BN          | 0.1528| 0.3680|
| Model without MC and BN| 0.1656| 0.3859|

As can be seen from Table 3, the model with the BN and MC two parts added achieved the
best effect, followed by the model with only one part of them added, and the model with no two
parts added had the worst effect, indicating that these two parts played a certain role in improving
the accuracy of the model.
3.4 Comparison of property prediction

In order to prove the effectiveness of the proposed method, we predicted the properties of the molecule respectively according to the distance between two atoms calculated from the QM9 and the distance predicted from our method. We selected a total of 10404 data of all molecules whose atomic number is equal to 15 in the dataset to form a small dataset QM9-15, and divided the training set and the test set according to the ratio of 8:2. The training set was used to train the MSGCN model. After the training was completed, the test set was put into the model to get the predicted two-atom distance between molecules and save it. The one-hot code of the atomic composition and atomic distance of the molecule were concatenated together as eigenvectors, and the machine learning method was used to perform regression to predict the properties of the molecule, such as the highest occupied molecular orbital energy (\( \epsilon_{\text{HOMO}} \)), the lowest non-occupied molecular orbital energy (\( \epsilon_{\text{LUMO}} \)) and energy gap (\( \epsilon_{\text{gap}} \)).

Here, the Gradient Boosting Regressor in the auto ML package was selected as the regressor to predict the properties of the molecules. The results are shown in the Table 4. Due to the limitation of the data, the prediction accuracy of the properties is not high, but it can be seen that the predicted results of atomic distances predicted by our model (Predicted) are very close to the predicted results of atomic distances calculated by the data in QM9(QM9-15). Among them, the difference between HOMO’s mean absolute errors is close to the error difference between MEGNet-Full and MEGNet-Simple in MEGNet. This indicates that the intermolecular atomic distance predicted by MSGCN can be applied to reality and the error is within the acceptable range.

Table 4 Error comparison of property prediction using QM9-15 and predicted atomic distance

(Unit: Ha)

| Properties | QM9-15 | Predicted |
|------------|--------|-----------|
| \( \epsilon_{\text{HOMO}} \) | 0.0094 | 0.0096 |
| \( \epsilon_{\text{LUMO}} \) | 0.0138 | 0.0145 |
| \( \epsilon_{\text{gap}} \) | 0.0184 | 0.0168 |

4 conclusion

From what has been discussed above, this paper proposes a two-branch MSGCN model based on graph convolutional network to solve the problems of high experimental cost and high cost of traditional calculation methods in determining molecular structures. The model takes the molecular SMILES expression as the input and converts it into graph representation through data preprocessing, which is used to predict the distance between two atoms in the molecule to achieve the purpose of determining the molecular structure. In order to prove the advance of the method, we compared the prediction results of this model with the ETKDG method in the RDKit chemical calculation package, and obtained better results than the ETKDG method on the subsets composed of molecules with less than 11, 15 and 20 atoms selected from the QM9 and on the whole dataset. In addition, as the atomic number difference between molecules in the dataset increases, the error
of our model has a smaller increase than that of ETKDG, showing better robustness. Subsequently, in order to verify the effectiveness of our proposed model, experiments on molecular property prediction were also conducted. The results showed that the error of molecular property prediction based on the distance predicted by our model was close to that based on the calculated distance of QM9, and the error difference between the two was within an acceptable range. Although the prediction results of MSGCN are generally better than those of ETKDG, there are many improvements in the model. In the next step, we will consider how to let the model learn more overall features of molecules, so as to reduce the prediction error of the distance between atoms without bonds, and get a more accurate molecular structure.

References

[1] Yan Zhang, et al. "Phase prediction in high entropy alloys with a rational selection of materials descriptors and machine learning models." Acta Materialia 185. (2020): doi:10.1016/j.actamat.2019.11.067.

[2] Yuqi Song, et al."Machine Learning based prediction of noncentrosymmetric crystal materials." Computational Materials Science 183.(2020):. doi:10.1016/j.commatsci.2020.109792.

[3] M. Attarian Shandiz, ,and R. Gauvin."Application of machine learning methods for the prediction of crystal system of cathode materials in lithium-ion batteries." Computational Materials Science 117.(2016):. doi:10.1016/j.commatsci.2016.02.021.

[4] Schütt, Kristof, et al. “SchNet: A Continuous-Filter Convolutional Neural Network for Modeling Quantum Interactions.” 31st Annual Conference on Neural Information Processing Systems, NIPS 2017, vol. 30, 2017, pp. 992–1002.

[5] Xuecheng Shao. “Simulation and Prediction Methods for Complex Crystal Structures.” Jilin University, PhD dissertation. 2019. (in Chinese)

[6] Battaglia, Peter W., et al. "Relational Inductive Biases, Deep Learning, and Graph Networks." ArXiv Preprint ArXiv:1806.01261, 2018.

[7] Xie, Tian, and Jeffrey C. Grossman. “Crystal Graph Convolutional Neural Networks for an Accurate and Interpretable Prediction of Material Properties.” Physical Review Letters, vol. 120, no. 14, 2018, p. 145301.

[8] Chen, Chi, et al. “Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals.” Chemistry of Materials, vol. 31, no. 9, 2019, pp. 3564–3572.

[9] Louis, Steph-Yves, et al. “Graph Convolutional Neural Networks with Global Attention for Improved Materials Property Prediction.” Physical Chemistry Chemical Physics, vol. 22, no. 32, 2020, pp. 18141–18148.

[10] Ramakrishnan, Raghunathan, et al. “Quantum Chemistry Structures and Properties of 134 Kilo Molecules.” Scientific Data, vol. 1, no. 1, 2014, pp. 140022–140022.

[11] Hamilton, William L., et al. “Inductive Representation Learning on Large Graphs.” Advances in Neural Information Processing Systems, vol. 30, 2017, pp. 1024–1034.

[12] Hornik, K., et al. “Multilayer Feedforward Networks Are Universal Approximators.” Neural Networks, vol. 2, no. 5, 1989, pp. 359–366.

[13] The RDKit: Open-Source Cheminformatics Software. http://www. rdkit.org

[14] Ebejer, Jean-Paul, et al. “Freely Available Conformer Generation Methods: How Good Are
They?” Journal of Chemical Information and Modeling, vol. 52, no. 5, 2012, pp. 1146–1158.

[15] Riniker, Sereina, and Gregory A. Landrum. “Better Informed Distance Geometry: Using What We Know To Improve Conformation Generation.” Journal of Chemical Information and Modeling, vol. 55, no. 12, 2015, pp. 2562–2574.

[16] Bruna, Joan, et al. “Spectral Networks and Locally Connected Networks on Graphs.” ICLR 2014: International Conference on Learning Representations (ICLR) 2014, 2014.

[17] Carneiro, Marcelo V., et al. “Prediction of Mechanical Properties of Steel Tubes Using a Machine Learning Approach.” Journal of Materials Engineering and Performance, vol. 30, no. 1, 2021, pp. 434–443.

[18] Ryu, Seongok, et al. “A Bayesian Graph Convolutional Network for Reliable Prediction of Molecular Properties with Uncertainty Quantification.” Chemical Science, vol. 10, no. 36, 2019, pp. 8438–8446.

[19] Coley, Connor W., et al. “A Graph-Convolutional Neural Network Model for the Prediction of Chemical Reactivity.” Chemical Science, vol. 10, no. 2, 2019, pp. 370–377.

[20] Imrie, Fergus, et al. “Deep Generative Models for 3D Linker Design.” Journal of Chemical Information and Modeling, vol. 60, no. 4, 2020, pp. 1983–1995.

[21] Li, Yanran, et al. “Densely Connected GCN Model for Motion Prediction.” Computer Animation and Virtual Worlds, vol. 31, 2020.

[22] Senior, Andrew W., et al. “Improved Protein Structure Prediction Using Potentials from Deep Learning.” Nature, vol. 577, no. 7792, 2020, pp. 706–710.

[23] Butler, Keith T., et al. “Machine Learning for Molecular and Materials Science.” Nature, vol. 559, no. 7715, 2018, pp. 547–555.

[24] Kearnes, Steven M., et al. “Molecular Graph Convolutions: Moving beyond Fingerprints.” Journal of Computer-Aided Molecular Design, vol. 30, no. 8, 2016, pp. 595–608.