

Identifying Conformation States of Polymer through Unsupervised Machine Learning

Sun Li-Wang, Li Hong, Zhang Xiao-Qin, Gao He-Bei, Luo Meng-Bo

Cite this article as:

Sun Li-Wang, Li Hong, Zhang Xiao-Qin, Gao He-Bei, Luo Meng-Bo. Identifying Conformation States of Polymer through Unsupervised Machine Learning[J]. *Chinese J. Polym. Sci.*, In press. doi: 10.1007/s10118-020-2442-6

View online: <https://doi.org/10.1007/s10118-020-2442-6>

Articles you may be interested in

Structures and Surface States of Polymer Brushes in Good Solvents: Effects of Surface Interactions

Chinese J. Polym. Sci. 2018, 36(9): 1047 <https://doi.org/10.1007/s10118-018-2100-4>

CAPABILITY OF DNA-FUELED MOLECULAR MACHINE IN TUNING ASSOCIATION RATE OF DNA-FUNCTIONALIZED GOLD NANOPARTICLES

Chinese J. Polym. Sci. 2013, 31(9): 1183 <https://doi.org/10.1007/s10118-013-1319-3>

CROWDING EFFECT INDUCED PHASE TRANSITION OF AMPHIPHILIC DIBLOCK COPOLYMER IN SOLUTION

Chinese J. Polym. Sci. 2013, 31(11): 1491 <https://doi.org/10.1007/s10118-013-1346-0>

不断向自然与金属冶金学习的高分子加工

Polymer Processing via Learning from Nature and Metal Metallurgy

高分子学报. 2016(7): 843 <https://doi.org/10.11777/j.issn1000-3304.2016.16048>

高分子研究中的一个知识问题——柔性链“从线团到小球”的构象变化

One Intellectual Problem in Polymer Research—the Coil-to-Globule Conformation Transition of Flexible Chains

高分子学报. 2017(9): 1389 <https://doi.org/10.11777/j.issn1000-3304.2017.16362>

流场中单链高分子过孔输运的动力学行为

Dynamics of Single Polymer Translocation Through a Fluidic Nanopore

高分子学报. 2015(10): 1196 <https://doi.org/10.11777/j.issn1000-3304.2015.15069>

Identifying Conformation States of Polymer through Unsupervised Machine Learning

Li-Wang Sun^a, Hong Li^{a*}, Xiao-Qin Zhang^{a*}, He-Bei Gao^{b*}, and Meng-Bo Luo^c

^a College of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou 325035, China

^b Department of Information, Wenzhou Polytechnic, Wenzhou 325035, China

^c Department of Physics, Zhejiang University, Hangzhou 310027, China

Abstract The study of the critical behavior is important for classifying different configuration states. Recently, machine learning is capable of discriminating polymer states in the presence of human supervision. Here, we introduce an unsupervised approach based on the self-organizing map (SOM) and the autoencoder network to locate critical phase transitions from raw configuration without the necessity for manual feature engineering. High-dimensional configuration data can be encoded to low-dimensional codes by employing neural network of multilayer restrictive Boltzmann machines and the intermediate code can also be reconstructed to high-dimensional input vector. And then the codes are used to cluster different configuration states for polymers adsorbed on the homogeneous and the stripe-patterned surface by the SOM network and K-Means method. This work presents an unusual tool to identify polymer configuration.

Keywords Phase transition; Machine learning; Clustering; Polymer; Configuration

Citation: Sun, L. W.; Li, H.; Zhang, X. Q.; Gao, H. B.; Luo, M. B. Identifying conformation states of polymer through unsupervised machine learning. *Chinese J. Polym. Sci.* <https://doi.org/10.1007/s10118-020-2442-6>

INTRODUCTION

Deep learning (DL)^[1,2] is an active research branch of machine learning (ML), which is exquisitely tailored to derive essential characteristics to build superior features to efficiently represent objects in complex data.^[3] The latest advances in the field of image processing^[4–6] and speech recognition have driven the employment of DL to numerous problems in condensed-matter physics^[7] and polymer.^[8] Applying DL to classify phases of matter and to recognize phase transitions has become the focus of attention.^[7–20]

Classifying phases of matter, either topological^[9] or symmetry-breaking,^[15] generally utilizes a DL approach based on supervised methods. Such methods need the previous labeling of the states in well-known regimes, for example, whether a state belongs to a coil or globule phase.^[8] However, the label of supervised learning restricts the possibility that the method will be able to detect undiscovered states. In contrast to the supervised case, unsupervised methods obtain features from the raw data without pre-labeling. In consequence, the unsupervised learning is especially valuable when the classification is not known to us. The principal component analysis (PCA) is an effective dimensionality

reduction technique used for polymer configuration clustering^[21] and computer vision.^[22–24] Compared with the PCA that obtains the highest possible variance whose components are orthogonal, the autoencoder could reserve the original state as much as possible, which can be seen as mapping from high scale to low scale for image identification.

In this paper, we adopt an unsupervised ML strategy based on the autoencoder^[25,26] and the self-organizing map (SOM),^[27,28] which can efficiently classify conformation states of polymers without the prior knowledge and can recognize critical phase transitions from raw configuration data. Results show that the unsupervised ML can efficiently identify the two-state phase transitions^[29–34] and the three-state phase transitions^[35] of polymers adsorbed on the homogeneous and stripe-patterned surfaces, respectively.

FEATURE ENGINEERING AND CLUSTERING METHOD

Polymer was simulated by the method of self-avoiding walk.^[29–31] In data preprocessing,^[36] all conformations are normalized according to the three dimensions of Cartesian coordinates, respectively, and stretched into one dimension as the input data for machine learning.

The standard neural network for supervised learning usually requires a large amount of labelled data. Here, what we need is the intermediate coding of the autoencoder neural network to describe polymers, which covers almost all the physical information of configurations. The autoencoder in-

* Corresponding authors, E-mail: lihong@wzu.edu.cn (H.L.)
E-mail: xqzhang@wzu.edu.cn (X.Q.Z.)
E-mail: bogolyx@126.com (H.B.G.)

Received March 9, 2020; Accepted April 10, 2020; Published online June 30, 2020

cludes an encoder network to convert the high-dimensional conformation into a low-dimensional code and a decoder network to recover the configuration from the code, which is treated as the feature to represent the structure of polymers. Autoencoder employs the mean square error between the original input and the decode output as the loss function for fine-tuning the network. By minimizing the loss between the reconstructed data and the original data, the code can represent the polymer conformation as much as possible. Autoencoder is depicted in Fig. 1.

In Fig. 1, we train the network from Encoder to code layer by two modes. First, a restrictive Boltzmann machine (RBM) is used to initially train network parameters, as shown in Fig. 2(a). Then, the trained parameters are copied to Encoder network and Decoder network. Finally, we further train the intermediate code of neural network through BP algorithm to make the decoding data close to the input data, as shown in Fig. 2(b). Next, we will discuss these two network modes in

detail. In order to be close to the global optimum, RBM is used for the weight $\mathbf{w}$ initialization of the characteristic network for every two neighbor layers from the input layer to the code layer, as shown in Fig. 2(a). The input vectors correspond to visible units while the feature detectors correspond to hidden units. The RBM via minimize system energy to boost the probability that the code can clearly represent each conformation. The energy is given by

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{b}^T \mathbf{v} - \mathbf{c}^T \mathbf{h} - \mathbf{v}^T \mathbf{w} \mathbf{h} \quad (1)$$

where $\mathbf{v}$ and $\mathbf{h}$ are the scaled configuration vector and the feature vector, respectively, $\mathbf{b}$ and $\mathbf{c}$ are their biases, and $\mathbf{w}$ is the weight matrix between $\mathbf{v}$ and $\mathbf{h}$. Every binary value h_j is set to be 1 by the metropolis algorithm

$$P(h_j = 1 | \mathbf{v}) = \sigma(c_j + \mathbf{v}^T \mathbf{w}_{:,j}) \quad (2)$$

where $\sigma(z) = 1 / [1 + \exp(-z)]$ is the sigmoid function and c_j is the bias of the hidden layer. Next, one-step contrastive divergence (CD-1) is used to minimize the system energy. All

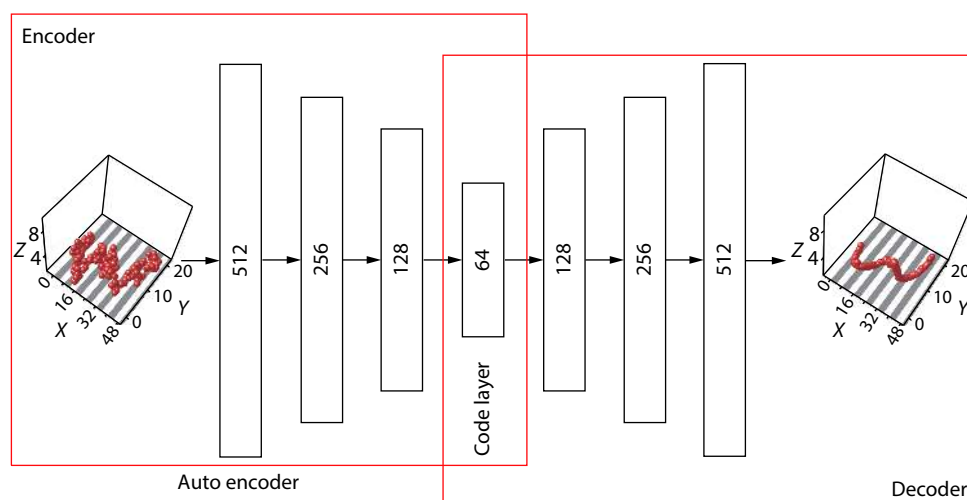


Fig. 1 Schematic diagram of autoencoder network with 7 layers. The coding layer outputs a 64-dimensional feature vector. The conformation generated by Decoder is similar to the origin one with small error.

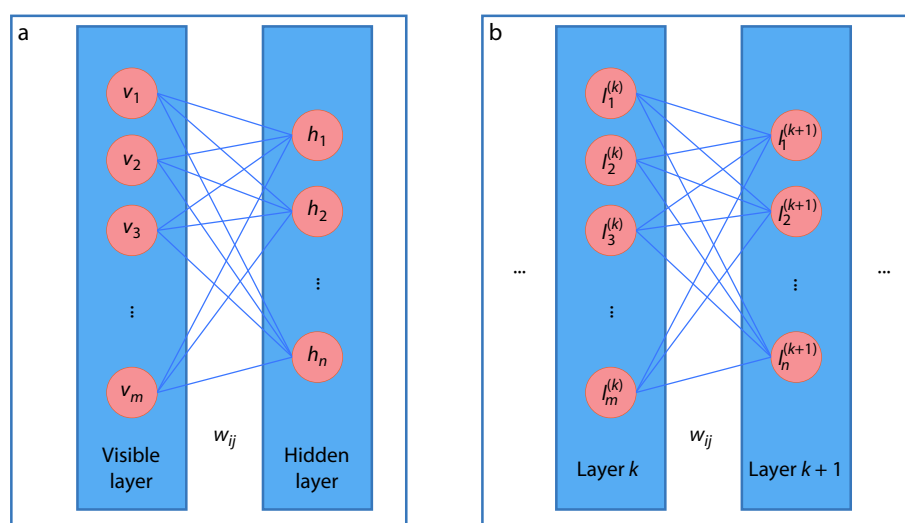


Fig. 2 The sketch of two neighbor layers in a neural network. (a) A restricted Boltzmann machine, including a visible layer $\mathbf{v}$ and a hidden layer $\mathbf{h}$. (b) The autoencoder, representing forward and feedback process.

parameters are updated by

$$\begin{aligned} \mathbf{w}, \mathbf{b}, \mathbf{c} &= \Phi(\mathbf{w}, \mathbf{v}, \mathbf{h}), \Phi(\mathbf{b}, \mathbf{v}, \mathbf{l}), \Phi(\mathbf{c}, \mathbf{h}, \mathbf{l}) \\ \Phi(\mathbf{x}, \mathbf{y}, \mathbf{z}) &= \mathbf{x} + \eta(\mathbf{y}_1 \mathbf{y}_2 - \mathbf{z}_1 \mathbf{z}_2) \end{aligned} \quad (3)$$

Entire CD-1 process consists of two parts of calculation. Eq. (2) is the first iteration, with $\mathbf{y}_1$ and $\mathbf{z}_1$ being the result of the iteration. Under reversing the role of the visible layer and hidden layer, Eq. (2) is calculated again, which is the second part of calculation, with $\mathbf{y}_2$ and $\mathbf{z}_2$ being the corresponding result. η is the learning rate.

After the RBMs network training is completed, their parameters are reconstructed into the autoencoder network. Then, the autoencoder network is fine-tuned using the back propagation (BP) algorithm. The typical two neighbor layers are shown in Fig. 2(b).

Different from above RBMs, l_j is not binary state, which is generated from

$$l_j^{(k+1)} = \varphi \left(l^{(k)\top} \mathbf{w}_{:j}^{(k)} + b_j^{(k)} \right) \quad (4)$$

here $l^{(k)}$ and $l^{(k+1)}$ are the vector of two neighbor layers, $\mathbf{w}_{:j}^{(k)}$ and $b_j^{(k)}$ are the weight and the bias of unit j of $l^{(k)}$, respectively, and $\varphi(x)$ is the rectified linear unit (ReLU) as activation function. The update of parameters by the BP algorithm is as follow:

$$w_{ij}^{(k)}, b_j^{(k)} = \Gamma \left(w_{ij}^{(k-1)} \right), \Gamma \left(b_j^{(k-1)} \right) \quad (5)$$

here $\Gamma(x) = x - \varepsilon \nabla_x \ell$, where $\nabla_x \ell$ is originated from chain rule and ε is the learning rate under updating.

After completing the feature extraction, we can get 64-vector code for each polymer. Then we start to cluster samples of polymers by using the hierarchical clustering algorithm, self-organizing map (SOM), which can efficiently visualize clustering results. The SOM consists of 20×20 two-dimensional grid of map units. Each unit is represented by a prototype vector $\mathbf{x}$ with the same dimension as input vector $\mathbf{y}$. The Euclidean distance of vector $\mathbf{x}$ from the origin is always scaled to 1 in SOM network. During training, units of SOM are adjusted iteratively by the input data. At each training step, distances between the input data $\mathbf{y}$ and all the prototype vectors $\mathbf{x}$ are computed. When a node $\mathbf{x}_i$ wins the competition with the minimum distance between $\mathbf{x}_i$ and $\mathbf{y}$ in a training process, all nodes $\mathbf{x}_j$ are adjusted by

$$\mathbf{x}_j = \mathbf{x}_j + \eta \hat{k}(\mathbf{x}_i, \mathbf{x}_j) (\mathbf{x}_i - \mathbf{x}_j) \quad (6)$$

where $\hat{k}(\mathbf{x}_i, \mathbf{x}_j) = \exp \left[-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / (2\sigma^2) \right]$. $\eta = 0.01$ is the initial learning rate and σ is the standard deviation of the radial basis function (RBF), which decreases over iterator. After enough training, similar nodes come together.

RESULTS AND DISCUSSION

To examine classifiers capable of the neural network of the autoencoder and SOM, different kinds of substances were chosen, for example, polymers simulated by the method of self-avoiding walk. The simulation uses a simple cubic lattice with the bond length of 1, $\sqrt{2}$, or $\sqrt{3}$. Monomer interacts with the attractive surface with $\varepsilon = -1$. Two typical simulations are used here for polymers on different surfaces^[31,35] with the chain length $N = 160$. One is the homogeneous attractive surface,^[31] while the other is the stripe-patterned surface with attractive strip and repellant strip, respectively.^[35] The simulation of

polymer adopts the Metropolis algorithm^[37] and the annealing algorithm^[38] under temperature from $T = 8.0$ to $T = 0.05$. The unit of temperature is $1/k_B$ with k_B the Boltzmann constant. Each simulation produces 9.6×10^3 independent conformations at every temperature for clustering and calculating the critical phase transition temperature.

Homogeneous Surface

The critical adsorption temperature (T_c) of polymer on the homogeneous surface can be estimated by the adsorption rate. As the polymer chain is adsorbed on the surface, its configuration changes with the adsorption degree. For example, the dependence of mean square radius of gyration $\langle R_g^2 \rangle$ on temperature T is presented in Fig. 3. We observe that the critical temperature T_c is roughly in the range between $T = 1.0$ and $T = 2.0$. To demonstrate the capability of unsupervised ML in recognizing the adsorption state and the desorption state, we performed two numerical experiments. One is to calculate T_c on different number of samples, n , at each temperature, and the other is to show results of clustering configuration state.

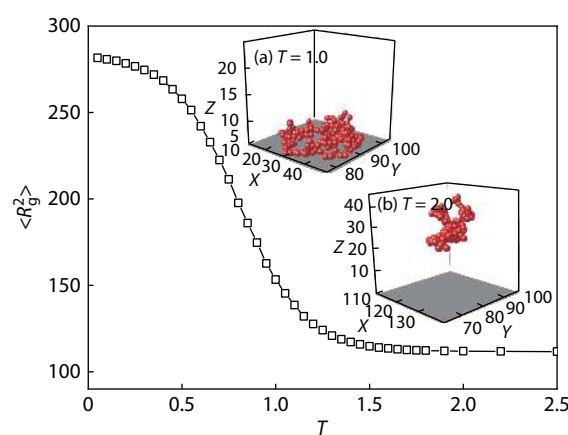


Fig. 3 The mean square radius of gyration of polymer on the homogeneous surface for the temperature from $T = 0.05$ to $T = 2.5$. The inset (a) is a typical conformation at $T = 1.0$; the inset (b) is a typical conformation at $T = 2.0$.

The original conformation data are normalized and used as the input of the autoencoder network to extract features coding vector. And then coding vectors are used to cluster by the SOM network and K-means method. T_c is calculated from the clustering results. Fig. 4 shows the relationship between T_c and the number of samples per temperature, n . When n is larger than 300, the value of T_c is convergent at $T_c = 1.35$. And when n is less than 300, T_c changes with n because the sample volume is too small to mine the desorbed state. Therefore, we take advantage of $n = 400$ in subsequent studies to estimate the stabilizing value of T_c .

After enough sample training, the distance to the nearest neighbor d_{nn} in the SOM grid is shown in the Fig. 5(a). The distance d_{nn} is obtained by mean Euclidean distance between the unit and its six surrounding units. The smaller the value d_{nn} is, the more similar the neighbor units are. All samples are mostly mapped in smaller d_{nn} regions. Then, K-means method is used to divide the grid area through the input data with the coding 64-vector. It repeats 10 times for

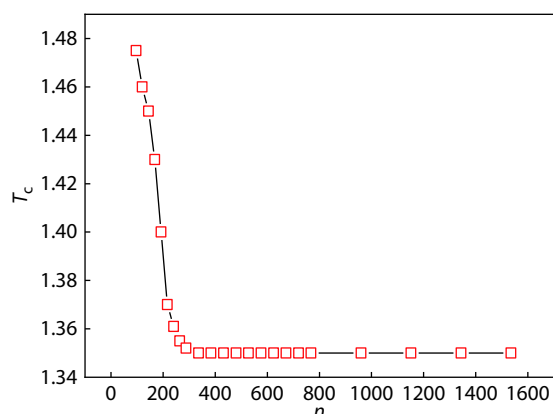


Fig. 4 Dependence of T_c on the number of samples n per temperature.

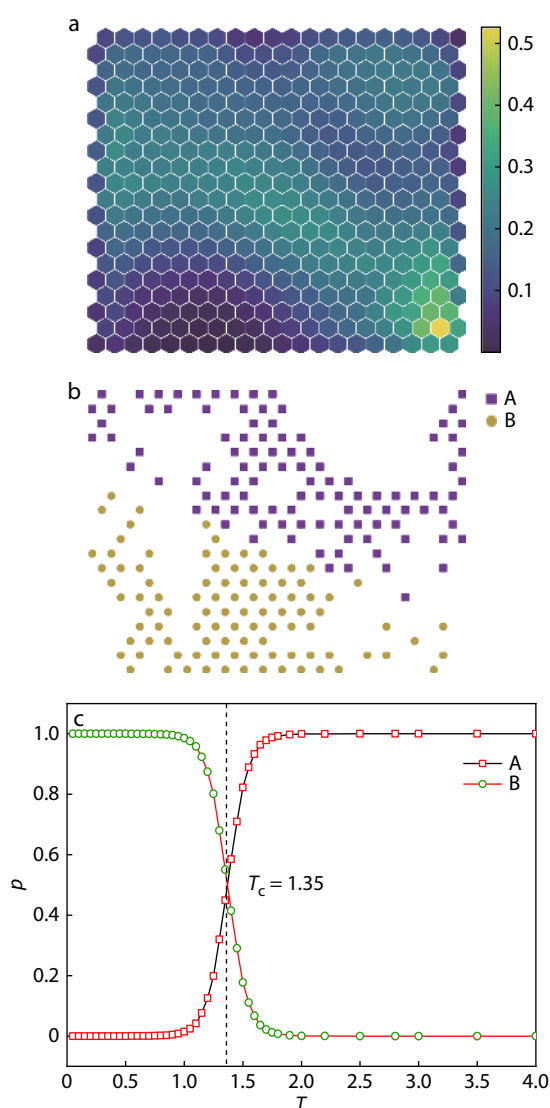


Fig. 5 (a) The distance to the nearest neighbor d_{nn} in SOM. (b) The distribution of samples maps on the grid. (c) The relationship between the class probability p and the temperature T . Label A represents desorption state and label B represents adsorption state.

searching the smallest SSE. And the distribution of all the samples mapped on the grids is shown in Fig. 5(b), where labels A and B represent different conformation classes of polymer. Finally, we count the proportion of samples in different categories at each temperature, as shown in Fig. 5(c). Here p means the probability of a sample judged as a particular class, which is obtained from the percentage of samples under each temperature.

Fig. 5(c) shows that the unsupervised learning can recognize the two-state phase transition of polymer adsorbed on the homogeneous surface. T_c is equal to 1.35, which has an identical conclusion as the traditional compute method.^[29]

Stripe-patterned Surface

In our previous research, we know that polymers adsorbed on the stripe-patterned surface exhibit three different states such as the desorption state, the multi-stripe adsorption state, and the single-stripe adsorption state. The dependence of mean square radius of gyration $\langle R_g^2 \rangle$ on temperature T is shown in Fig. 6. We observe that polymer is adsorbed on the single stripe for the single-stripe-adsorption state in inset (a), adsorbed on multiple stripes for the multi-stripe-adsorption state in inset (b), and away from the surface for the desorption state in inset (c). And we manage to obtain the location of phase transition directly from the dependence of the mean square radius of gyration on T . It is very difficult to recognize conformations of polymers adsorbed on stripes by machine learning. Firstly, it is difficult to cluster high dimensional data directly. Dimensionality reduction is very important, but in the process of dimensionality reduction, the loss of main features must be minimized. Secondly, near the critical adsorption point (T_c) or the critical single stripe adsorption point (T_s), conformations of polymers are very similar. No obvious features can be distinguished directly from coordinate analysis. And it is hard to abstract their obvious features.

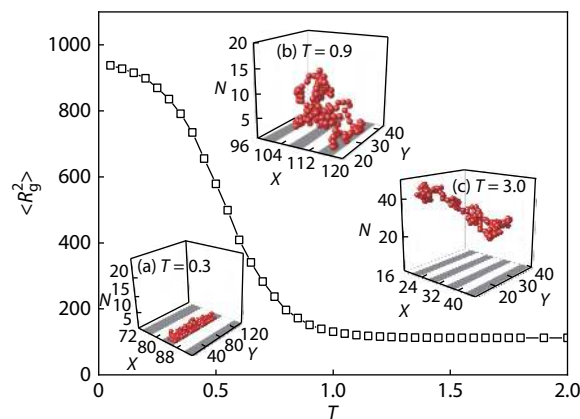


Fig. 6 The mean square radius of gyration of polymer on the stripe surface for the temperature from $T = 0.05$ to $T = 2.0$. Typical conformations of the polymer in (a) single-stripe adsorption state at $T = 0.3$, (b) multi-stripe adsorption state at $T = 0.9$, and (c) the desorption state at $T = 3.0$. The dark part of the surface is adsorbable for polymers.

The normalized conformation data are used as the input of the autoencoder network to extract features coding vector. And then coding vectors are used to cluster by the SOM net-

work and K-means method. After sufficiently training, the distance of unit to nearest neighbor d_{nn} in the SOM grid is shown in Fig. 7(a). And then the distribution of all the samples mapped on the grids is shown in Fig. 7(b), where A, B, and C represent the desorption state, the multi-stripe adsorption state, and the single-stripe adsorption state, respectively. Finally, we present the dependence of probability p on the temperature T , as shown in the Fig. 7(c). The methods successfully identify the three-state phase transition of polymers on the stripe surface. As the temperature T decreases, there are two critical temperatures $T_1 = 0.57$ and $T_2 = 1.02$. The probability distribution in this experiment is similar to results obtained by the labeling method.^[39] Results show that configuration state of polymers on the stripe surface can be identified by the neural network of autoencoder and SOM after sufficient training.

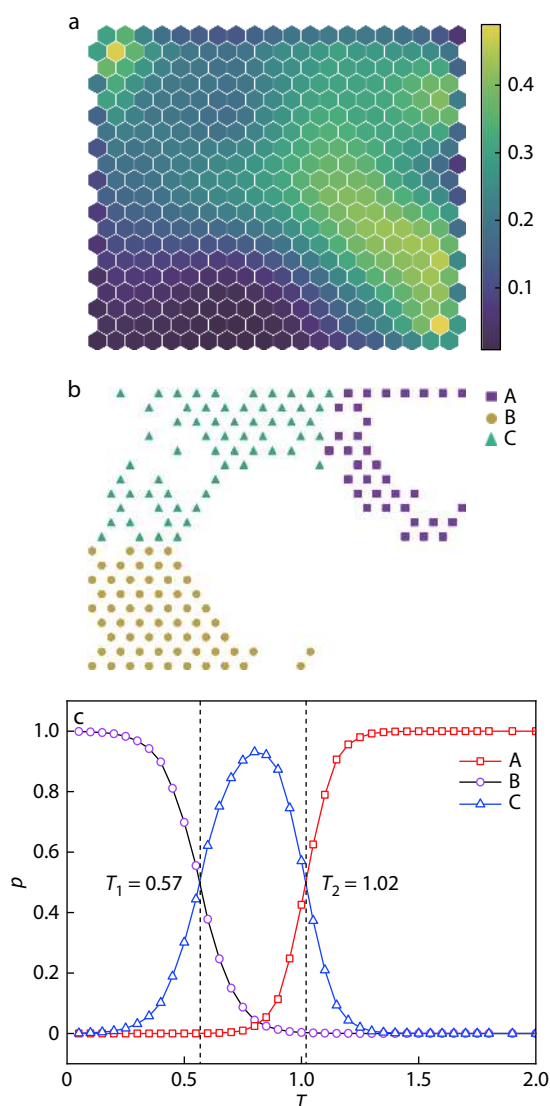


Fig. 7 (a) The distance to the nearest neighbor d_{nn} in the SOM. (b) The distribution of samples mapped on the grid. (c) The relationship between class probability p and temperature T . Labels A, B, and C represent desorption state, single-stripe adsorption state, and multi-stripe adsorption state, respectively.

Finally, we compare the results (T_c) of the traditional computing method,^[35] the labeled supervised learning,^[39] and the unsupervised learning in Table 1. The prediction result of unsupervised learning is closer to the simulation result than that of label learning. The labeled supervised learning shows lower accuracy because it is difficult to label the correct state of the polymer conformation near CAP. However, the auto-encoder network is employed to encode a high-dimensional configuration data to a lower dimensional code which can be reconstructed to the high-dimensional vector similar to the input vector. Then we use the self-organizing map (SOM) network and the K-mean method to cluster the intermediate code. We find the prediction accuracy is close to the traditional computing method because the intermediate code keeps the main information of polymer conformation. Without any prior knowledge, the unsupervised method achieves right results with regard to others methods.

Table 1 Critical temperature of three methods with the same data.

Method	Homo T_c	Stripe T_r	Stripe T_c
Traditional	1.35	0.58	1.05
Labeled	1.5	0.55	1.1
Unsupervised	1.35	0.57	1.02

CONCLUSIONS

We present an unsupervised machine learning method for clustering features of polymer chains which are produced by Monte Carlo simulation. Conformations of polymers absorbed on the homogeneous surface or stripe-patterned surface can be recognized correctly by training the autoencoder network and the self-organizing map (SOM) network. When the number of samples per temperature is above 400, the values of the critical temperature gradually stabilize. The neural network of autoencoder and SOM can identify not only two states between the desorbed state and the adsorbed state for polymers adsorbed on the homogeneous surface, but also three different states including the desorbed state, multi-stripe adsorbed state, and single-stripe adsorbed state for polymers on the stripe-patterned surface. The neural network is a novel method for identifying molecular configuration and investigating phase transition.

ACKNOWLEDGMENTS

This work was financially supported by the National Natural Science Foundation of China (No. 11775161).

REFERENCES

- 1 LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, 521, 436–444.
- 2 Zhang, X. Q.; Liu, Q. Q.; Wang, D.; Zhao, L.; Gu, N.; Maybank, S. Self-taught semisupervised dictionary learning with nonnegative constraint. *IEEE T. Ind. Inform* **2020**, 16, 532–543.
- 3 Zhang, X. Q.; Wang, T.; Wang, J. X.; Tang, G. Y.; Zhao, L. Pyramid channel-based feature attention network for image dehazing. *Computer Comput. Vis. Image Underst.* **2020**, 197–198, 103003.
- 4 Zhang, X. Q.; Wang, D.; Zhou, Z. Y.; Ma, Y. Robust low-rank tensor recovery with rectification and alignment. *IEEE T. Pattern Anal.*

- 2020, DOI: 10.1109/TPAMI.2019.2929043.
- 5 Zhang, X. Q.; Zheng, J. J.; Wang, D.; Zhao, L. Exemplar-based denoising: a unified low-rank recovery framework. *IEEE T. Circ. Syst. Vid.* **2020**, DOI: 10.1109/TCSVT.2019.2927603.
 - 6 Zhang, X. Q.; Hu, W. M.; Xie, N. H.; Bao, H. J.; Maybank, S. A robust tracking system for low frame rate video. *Int. J. Computer Vision.* **2015**, *115*, 279–304.
 - 7 Carrasquilla, J.; Melko, R. G. Machine learning phases of matter. *Nat. Phys.* **2017**, *13*, 431–434.
 - 8 Wei, Q.; Melko, R. G.; Chen, J. Z. Y. Identifying polymer states by machine learning. *Phys. Rev. E* **2017**, *95*, 032504.
 - 9 Rodriguez-Nieva, J. F.; Scheurer, M. S. Identifying topological order through unsupervised machine learning. *Nat. Phys.* **2019**, *15*, 790–795.
 - 10 Ch'ng, K.; Carrasquilla, J.; Melko, R. G. Machine learning phases of strongly correlated fermions. *Phys. Rev. X* **2017**, *7*, 031038.
 - 11 Zhang, Y.; Kim, E. A. Quantum loop topography for machine learning. *Phys. Rev. Lett.* **2017**, *118*, 216401.
 - 12 Zhang, Y.; Melko, R. G.; Kim, E. A. Machine learning Z_2 quantum spin liquids with quasiparticle statistics. *Phys. Rev. B* **2017**, *96*, 245119.
 - 13 van Nieuwenburg, E. P. L.; Liu, Y. H.; Huber, S. D. Learning phase transitions by confusion. *Nat. Phys.* **2017**, *13*, 435–439.
 - 14 Ohtsuki, T.; Ohtsuki, T. Deep learning the quantum phase transitions in random electron systems: applications to three dimensions. *J. Phys. Soc. Jpn.* **2017**, *86*, 044708.
 - 15 Yoshioka, N.; Akagi, Y.; Katsura, H. Learning disordered topological phases by statistical recovery of symmetry. *Phys. Rev. B* **2018**, *97*, 205110.
 - 16 Zhang, P.; Shen, H.; Zhai, H. Machine learning topological invariants with neural networks. *Phys. Rev. Lett.* **2018**, *120*, 066401.
 - 17 Carvalho, D.; García-Martínez, N. A.; Lado, J. L. Real-space mapping of topological invariants using artificial neural networks. *Phys. Rev. B* **2018**, *97*, 115453.
 - 18 Beach, M. J.; Golubeva, A.; Melko, R. G. Machine learning vortices at the Kosterlitz-Thouless transition. *Phys. Rev. B* **2018**, *97*, 045207.
 - 19 Wang, C.; Zhai, H. Machine learning of frustrated classical spin models (II): kernel principal component analysis. *Front. Phys.* **2018**, *13*, 130507.
 - 20 Hu, W.; Singh, R. R.; Scalettar, R. T. Discovering phases, phase transitions, and crossovers through unsupervised machine learning: a critical examination. *Phys. Rev. E* **2017**, *95*, 062122.
 - 21 Xu, X.; Wei, Q.; Li, H. Recognition of polymer configurations by unsupervised learning. *Phys. Rev. E* **2019**, *99*, 043307.
 - 22 Zhang, X. Q.; Li, W.; Ye, X. Z.; Maybank, S. Robust hand tracking via novel multi-cue integration. *Neurocomputing* **2015**, *157*, 296–305.
 - 23 Zhang, X. Q.; Hu, W. M.; Chen, S. Y.; Maybank, S. Graph-embedding-based learning for robust object tracking. *IEEE T. Ind. Electron.* **2014**, *61*, 1072–1084.
 - 24 Zhang, X. Q.; Li, C. C.; Tong, X. F.; Hu, W. M.; Maybank, S.; Zhang, Y. Human pose estimation and tracking via parsing a tree structure based human model. *IEEE T. Syst. Man Cy-S.* **2014**, *44*, 580–592.
 - 25 Hinton, G. E.; Salakhutdinov, R. R. Reducing the dimensionality of data with neural networks. *Science* **2006**, *313*, 504–507.
 - 26 Liou, C. Y.; Cheng, W. C.; Liou, J. W. Autoencoder for words. *Neurocomputing* **2014**, *139*, 84–96.
 - 27 Kohonen, T. The self-organizing map. *Proc. IEEE.* **1990**, *78*, 1464–1480.
 - 28 Vesanto, J.; Alhoniemi, E. Clustering of the self-organizing map. *IEEE T. Neural Networks* **2000**, *11*, 586–600.
 - 29 Li, H.; Qian, C. J.; Luo, M. B. Simulation of a flexible polymer tethered to a flat adsorbing surface. *J. Appl. Polym. Sci.* **2012**, *124*, 282–287.
 - 30 Li, H.; Qian, C. J.; Sun, L. Z. Conformational properties of a polymer tethered to an interacting flat surface. *Polym. J.* **2010**, *42*, 383–385.
 - 31 Li, H.; Qian, C. J.; Wang, C. Critical adsorption of a flexible polymer confined between two parallel interacting surfaces. *Phys. Rev. E* **2013**, *87*, 012602.
 - 32 Li, H.; Qian, C. J.; Luo, M. B. Critical adsorption of copolymer tethered on selective surfaces. *J. Chem. Phys.* **2016**, *144*, 164901.
 - 33 Luo, M. B. The critical adsorption point of self-avoiding walks: a finite-size scaling approach. *J. Chem. Phys.* **2008**, *128*, 044912.
 - 34 Luo, M. B.; Huang, J. H. Monte Carlo simulation of polymer chain with ferromagnetic Ising interaction. *J. Chem. Phys.* **2003**, *119*, 2439–2443.
 - 35 Li, H.; Gong, B.; Qian, C. J.; Luo, M. B. Critical adsorption of a flexible polymer on a stripe-patterned surface. *Soft Matter* **2015**, *11*, 3222–3231.
 - 36 Zhang, X. Q.; Li, W.; Hu, W. M.; Ling, H. B.; Maybank, S. Block covariance based l_1 tracker with a subtle template dictionary. *Pattern Recogn.* **2013**, *46*, 1750–1761.
 - 37 Haario, H.; Saksman, E.; Tamminen, J. An adaptive Metropolis algorithm. *Bernoulli*. **2001**, *7*, 223–242.
 - 38 Szu, H.; Hartley, R. Fast simulated annealing. *Phys. Rev. A* **1987**, *122*, 157–162.
 - 39 Sun, L. W.; Li, H.; Wang, P. J. Recognition of adsorption phase transition of polymer on surface by neural network. *Acta Phys. Sin.* **2019**, *68*, 60–66.