Unraveling intricate processes of glassy dynamics from static structure by machine learning relative motion

Hayato Shiba,1,* Masatoshi Hanai,1 Toyotaro Suzumura,2,1 and Takashi Shimokawabe1†

1Information Technology Center, University of Tokyo, Chiba 277-0882, Japan
2Graduate School of Information Science and Technology, University of Tokyo, Tokyo 133-8658, Japan
(Dated: July 19, 2022)

The recent developments in machine learning methodologies have enabled accurate predictions of glassy dynamics. The machine learning models developed thus far ensure that the hidden characteristic features of the local structures around each particle can be autonomously related to the hotspots where particles are more prone to move. Herein, we present a distinct approach for developing a novel graph neural network model that learns the relative motion between neighboring particle pairs as its target quantity, in addition to the self-displacement of each particle. This enables precise predictions of the distance of separation of each neighbor pair from its static structure. Furthermore, if we enable the model to simultaneously learn the self-displacement and relative motion, it can predict the particle propensity for self-motion within a short period, with an accuracy drastically higher than those of existing models. This is achieved by utilizing the physical information underlying these quantities that complement the intricate dynamics of both the short-time collective vibrations and structural changes. This model represents a new state-of-the-art prediction method for glassy dynamics, thus introducing a new approach for elucidating the crossover from short-time vibrational motion to long-time structural relaxation in glasses.

I. INTRODUCTION

When a liquid is rapidly cooled, it freezes into a glass which retains a random amorphous structure. This phenomenon, referred to as the glass transition, is ubiquitous, but its origin still remains unclarified. The elucidation of this origin has been an important research topic for decades [1]. Along with the drastic slowdown of atomic motion accompanying the glass transition, localized domains of particles that rearrange more preferentially than others grows up [2–4]. Because particles are randomly arranged, it is difficult to identify the structural features of a glass-forming liquid that determines where such domains appear. Local geometric [5] or non-geometric orders derived from static structures [6, 7] have accumulated evidences for the structure–dynamics correspondence. However, the underlying structural origin remains elusive.

The recent developments in machine learning methodologies and network analyses, along with the improvements in computational capability and the processing of large volumes of data, have enabled the accurate determination of the structure–dynamics correspondence in glassy systems [8–14]. In particular, Bapst et al. demonstrated the capability of graph neural networks (GNNs) in determining this correlation [13]. GNNs are a class of deep learning algorithms that operate on a graph and are highly capable of expressing complex features without intervening physical insights. GNNs can express molecular assemblies by embedding the characteristics of particles (atoms) and the relation between pairs of these particles as features of the nodes and edges in a graph, respectively. Therefore, they have recently been implemented in various applications in the field of materials science [15–18]. Furthermore, Bapst et al. utilized these features of GNNs to predict the particle propensity in a glass-forming liquid based on its static structure and reported an unprecedented accuracy in these predictions [13]. This high accuracy in the predictions of glassy dynamics has enabled the development of further applications, as evidenced by the application of inverse modeling [19]. This has also led to the development of a cost-effective machine learning model that can overcome the high computational cost of GNNs [14] by identifying structural descriptors internally designed by GNNs.

In most previous studies, the dynamics was probed via quantities defined for each particle, including particle displacement. However, the dynamics of structural arrangement in glasses concerns the interexchange of positional relations between particles rather than the movement of a single particle. Therefore, by setting a dynamical variable that characterizes relative motion between pairs of particles as the target quantity to be learnt, such rearranging motion can be directly probed by machine learning, which may help capture different physical process that underlies the intricate processes of dynamics. In the short-time scale, there exists a collective motion including quasi-localized vibrations and phonons [20–22], where the former is known to bear low energy barriers for long-time structural rearrangements [6, 7, 22–24]. As such transient dynamics are collective in their nature, the existing machine learning models targeting at particle-level dynamic descriptors are not capable of capturing such intricate dynamics, which leaves us a challenge of endowing GNNs with the capability of learning such fluctuations.

Herein, we show that this nontrivial task can be achieved using the proposed GNN model, which completely learns the relative motion between particles, along with the particle-level motion. This enables us to predict the short-time dynamics with significantly improved precision than the previous models. The proposed model, therefore, updates the state-of-the-art GNN model [13–14] in terms of prediction ability and forms a new ground basis for exploration of glassy slow relaxation by machine learning.
II. RESULTS

A. Predicting pair-distance change as edge feature

Here, we propose a new GNN model, “BOnd TAargeting Network (BOTAN)”, that can learn a target characteristic quantity assigned on edges corresponding to pairs of close-by particles. BOTAN is a straightforward extension of the GNN used in the previous study [13], which we term as NT-GNN (“node-targeting GNN”) because it was trained using only the particle propensity for motion at each node of the graph. BOTAN and NT-GNN are based on the interaction network [25][26]. Interaction network consists of an encoder–process–decoder architecture, where features of nodes and edges in a graph are mutually computed by exchanging messages between a pair of two-layer multi-layer perceptrons (MLPs) assigned for nodes and edges. This architecture is suitable in extracting intricate relationships from simulation data computed with high precision. Moreover, because the network treats node and edge features equivalently, regressing the network toward the characteristic quantities on edges as well as nodes is possible by decoding edge embeddings by an MLP (see Methods). BOTAN shares some common characteristics with a recent study exploiting interaction network [27].

As a new target quantity for learning interparticle relative motion, we introduce pair-distance change after time $t$,

$$E_{ij}(t) = r_{ij}(t) - r_{ij}(0),$$

where $r_{ij}(t)$ is the distance between a pair of particles $i$ and $j$ at time $t$. Here, only the pairs are considered within a threshold distance $(r_c = 2.0)$ at the initial time, $t = 0$. We track these pairs to determine whether they were separated subsequently, in order to detect the structurally rearranging regions. This concept is nearly similar to previous ideas of counting the replacement of neighboring pairs [6] [28][30], where it has also been referred to as “bond breakage” [8][31][33].

To reveal the predictive ability of BOTAN, we train it with a dataset to predict the pair-distance changes. The dataset for learning is trajectory data of a 3D binary Kob–Andersen Lennard–Jones (KALJ) mixture with 80:20 (types A and B) composition [34], wherein the pairwise force is smoothed at its cutoff [35][36]. The actual target quantity of machine learning is the ensemble average over the result of 32 independent simulation runs generated from each of initial configurations, so-called the isoconfigurational ensemble [6][37].

In the remainder of this article, we use the dimensionless units based on the Lennard-Jones potential and target quantities of learning that are always the average on isoconfigurational ensembles for which explicit notations are abbreviated. Machine learning hyperparameters are set similar to NT-GNN [13]. In particular, the standard $L_2$-norm loss function is used for learning and the input graph is constructed in a manner that pairs of nodes are connected by an edge when the corresponding pair distance is lower than $r_c$, resulting in approximately 166,000 edges. The model is trained with the dataset generated from 400 initial configurations, where for small $t$, a large number of epochs are required for the convergence of the loss function (see Supplemental Information, Fig. S1 for learning curve of this training). For training, different epoch numbers are set for the different time points and temperatures to avoid overlearning due to the limited volume of these data (see Supplemental Information, Table S1). Prediction of the model is examined with test dataset generated from other 100 initial configurations (see Methods for all details).

Fig. 1 depicts the prediction of the pair-distance change, $E_{ij}(t)$, for a particle configuration in the test dataset, using BOTAN trained at $t = (a) 130$ and (b) 4120 under the lowest temperature considered $T = 0.44$. These time points correspond to 0.03$\tau_a$ and $\tau_a$, with $\tau_a$ the $\alpha$-relaxation time. From left to right in each, (1) predicted values, (2) the “actual” distance change directly evaluated as the isoconfigurational ensemble average from the simulation, and (3) the error map, are shown in color maps projected onto segments representing pairs initially in the distance of $r_{ij}(t = 0) < 1.35$. They show that BOTAN reproduces spatial distribution of pair-distance changes well; particularly for short-time prediction (a) $t = 130$, we find that BOTAN discerns where specific pairs become dissociated. Over a long-time (b) $t = 4120$, which is equal to the $\alpha$-relaxation time $\tau_a$, the prediction becomes more diffuse but still captures the spatial contrast between mobile and immobile regions.

In Fig. 2, the prediction accuracy of BOTAN over $E_{ij}(t)$ is illustrated using the Pearson correlation coefficient, which quantifies the proximity between the predicted and ground truth data. This proximity should be quantified carefully to ensure that the comparison is performed over equivalent particle pairs. Therefore, among pairs of type A particles (pairs related to type B are excluded from consideration), pairs within the first neighboring distance $r_{ij}(t = 0) < 1.35$ are chosen in this comparison, because a neighbor pair in the first and second neighbor shells is expected to exhibit different distance changes on average. The prediction accuracy thus evaluated is high in the short time, wherein the prediction accuracy monotonically declines. This result is in clear contrast to that of the previous NT-GNN [13], where the prediction accuracy of particle propensity decreases in the early stages of structural relaxation.

Predicting particle propensity as node feature

BOTAN is modeled by adding a decoder applied to edge embeddings; however, this decoder can be possibly retained for node embeddings. This leads to an idea to train BOTAN with target quantities for both, that is, particle propensity for the self-displacement

$$S_i(t) = |r_i(t) - r_i(0)|$$

in addition to the pair-distance changes $E_{ij}(t)$. Here, $r_i(t)$ denotes the 3D position of particle $i$ at time $t$. The actual target quantities are again their isoconfigurational ensemble averages. The simultaneous learning of these quantities can be realized by setting the loss function as a weighted sum of $L_2$-norm losses for $S_i(t)$ and $E_{ij}(t)$. The model is then trained.
FIG. 1. Prediction of pair distance change $E_{ij}(t)$ and the ground truth data at a temperature $T = 0.44$ for two times $t = (a) 130 (0.03 \tau_\alpha)$ and (b) 4120 ($\tau_\alpha$), respectively. From left to right, (1) predictions, (2) ground truth data computed out of the simulation result, and (3) prediction errors computed as difference between them. A specific cross section ($11.1 < z < 11.9$, with the width of 0.8) is cut out of the 3D box system. The distributions are shown in color maps that are projected on each line. Lines represent pairs of particles $i$ and $j$ and only the pairs satisfying $r_{ij}(t=0) < 1.35$ are shown in the plot.

FIG. 2. Pearson correlation coefficients between predicted and actual values of pair-distance change $E_{ij}(t)$ as a function of time $t$, at temperatures $T = 0.44$, 0.50, 0.56, and 0.64. Error bars depict the median, best, and worst of five independently trained models. Dotted lines indicate the $\alpha$-relaxation time for respective temperature.

First, we compare the prediction ability of BOTAN with that of NT-GNN by Bapst et al. Figure 3 shows predictions for $t = (a) 130$ and (b) 4120 for $T = 0.44$ of NT-GNN and of BOTAN, in addition to the “actual” propensity directly evaluated from the trajectory data in the test dataset. As all the snapshots are shown in the same cross section as in Fig. 1 the spatial correspondence can be seen between the predictions of $S_i(t)$ and $E_{ij}(t)$. For short time prediction ($t = 0.03 \tau_\alpha$) in Fig. 3 (a), BOTAN is superior in predicting spatially heterogeneous patterns of particle propensity after a large number of epochs of training. Contrarily, prediction over a longer time $t = \tau_\alpha$ remains at the same level of accuracy as in the previous model, indicating that the addition of edge targets in BOTAN has not improved its ability over the long-time structural relaxation.

To determine the cause for the improvement in the short-time prediction and to get one step closer to the reason, two kinds of additional plots are created in Fig. 4. One is error maps of predictions as the difference between the predicted and ground truth propensity map, plotted for NT-GNN and BOTAN. The other is “locally-averaged” displacement field over nearest neighbors computed from the particle trajectory as the ground-truth data, which indicates particle motion that is not directly related to interparticle rearrangement. In this way, we intend to extract thermal collective motion including quasi-localized vibrations and phonons \cite{21,22}, existing behind the glassy structural relaxation.

In the error map of NT-GNN, the existence of large scale
heterogeneity represents its inability to capture the full heterogeneity in the map. However, this locally averaged displacement field, plotted with 3D vectors, demonstrates the existence of large-scale collective motion underlying the glassy structural relaxation, which considerably exceeds the length scale of the particle size. On comparing them, it is observed that the displacements are spatially correlated with the prediction errors, being parallel with the contour lines of the prediction error map. In the prediction result of BOTAN, on the other hand, these errors are not perfectly but clearly suppressed. Considering these results, the loss of accuracy in the prediction can be presumed to be related to collective thermal fluctuations, which the GNN cannot directly capture in nature, as discussed in the following.

B. Prediction accuracy improved by learning relative motion

GNNs are effective tools that can identify hidden features from relational data. Glassy systems remain in a metastable state at low temperatures for a long time, without any major changes in their structure; this can account for the high accuracy of NT-GNN in predicting particle propensity [13]. However, in the previous study conducted by [13], a decline was observed in the prediction accuracy of NT-GNN in terms of the short-time dynamics around the plateau of the self-intermediate function, despite the expected ease of prediction. In computing the particle propensity, particle motion is averaged over a long time to hide short-time fluctuations, which might take place in the form of quasi-localized vibrations or phonon fluctuations. Such spatially-extended motion makes its relationship with the structure nontrivial, which can be the reason for the decline in the prediction accuracy of NT-GNN [13].

Regarding the short-time dynamics, the pair-distance change $E_{ij}(t)$ has its own significance in spite of its extreme simplicity. The dynamic heterogeneity in glassy dynamics is often characterized through the self-motion of a tracer particle. The particle propensity, a previous target quantity of learning, falls into this category, whereas other quantities include self-intermediate scattering function, self-part of the van-Hove functions, and four-point functions (space-time density correlation) [2]. In the meanwhile, another aspect of the dynamics can be captured by quantifying structural rearrangement, which is more intimately linked to viscous relaxation process rather than the diffusion. An easy way to realize it is to count the replacement of neighboring pairs [6, 28, 29], which is also termed as “bond breakage” [3, 31, 32]. For the 3D KALJ liquid, an author has shown that the dynamic length scale characterized by the bond breakage and four-point density correlation is similar [32]. However, these two types of quantities are influenced by the details of the dynamics in a different manner; As an obvious example, when particles are directly driven by elastic sound waves, the resulting particle motion is collective and still results in a finite particle propensity, even without structural rearrangement.

From these viewpoints, we assess the prediction accuracy of BOTAN in comparison with the previous model and further
investigate the short-time collective fluctuations affecting the dynamics by computing the Pearson correlation coefficients. Because two types of particles (A and B) of a 3D KALJ liquid are expected to exhibit different propensities over a long period, the Pearson correlation coefficient is estimated using only the data for type A particles. Figure 5 shows the prediction accuracy of BOTAN and of NT-GNN with regard to the particle propensity $S_i(t)$. The trend of time dependence is in agreement with the previous results [13]: in the shorter time before reaching the plateau region ($t < 0.2$), the Pearson correlation coefficients assume high value, then it falls below 0.5, and afterwards gradually increases to reach its peak at around the $\alpha$-relaxation time. Conversely, BOTAN exhibits remarkable improvement in the prediction accuracy in the time region of $t < 100$ for all the temperatures under investigation. In fact, BOTAN first learns the short-time dynamics on edges and then acquires the ability to predict particle propensity at the nodes, as inferred from the dependence of the Pearson coefficients on the number of epochs during training (see Supplemental Information, Fig. S3). Overall, BOTAN internally fixes the prediction errors over $S_i(t)$ with high precision using information learned from pair-distance changes.

Finally, we determine the level of prediction accuracy that can be achieved by training NT-GNN by replacing target variables into a quantity in which the effect of collective thermal fluctuations is restricted to a certain extent. For this purpose, we introduce and briefly examine two quantities that characterize particle rearrangements being less affected by such collective motion. One is the relative displacement that they would have under an uniform strain.

\[
\mathcal{R}_c(t) = \frac{1}{N_{\text{n.n.}}} \sum_{j \in \text{n.n.}} \left| \left[ r_i^\alpha(t) - r_j^\alpha(t) \right] - \left[ r_i^\beta(0) - r_j^\beta(0) \right] \Lambda_{\alpha\beta} \right|
\]

wherein the vector and tensor components $\alpha, \beta \in \{x, y, z\}$ are explicitly denoted for clarity. Here, $\epsilon_{\text{sg}}$ represents the local strain tensor, which is evaluated from change in the local arrangement of neighboring particles in the distance of 1.6 (See “Methods”). When this local strain reduces to zero ($\epsilon_{\text{sg}} = 0$), which is valid in the limit of $t \to 0$, this quantity reduces to a more easily interpretable form $C_c(t) = |\Delta r_i(t) - N_{\text{n.n.}}^{-1} \sum_{j \in \text{n.n.}} \Delta r_j(t)|$, clearly manifesting itself as the relative displacement with respect to neighbor environments.

The other quantity is a further variant but widely used and defined as the mean-square of difference between the actual displacement and the uniform strain displacement of the neighbors.

\[
D_c(t) = \frac{1}{N_{\text{n.n.}}} \sum_{j \in \text{n.n.}} \sum_{\alpha} \left[ \left| r_i^\alpha(t) - r_j^\alpha(t) \right| - \left| r_i^\beta(0) - r_j^\beta(0) \right| \Lambda_{\alpha\beta} \right]^2.
\]

This quantity is close to $\mathcal{R}_c(t)$ in its form; the difference lies in the squared sum that is more distinctly affected by a specific $i$-$j$ pair if a specific $i$-$j$ pair becomes further away than other pairs. Therefore, $D_c(t)$ is more susceptible to hot spots where particle rearrangement preferably occurs.

In Fig. 5, Pearson correlation coefficients between predictions of NT-GNN and the actual values of neighbor-relative particle propensities $R_c(t)$ and $D_c(t)$ are also shown. The correlation coefficient is evaluated only for type-A particles, as a proximity between the predicted and ground truth values (see Supplemental Information Figs. S3 and S4 for visualization of the corresponding predicted and ground truth values of these quantities). The prediction accuracy of NT-GNN improves over these quantities, especially in the time regions shorter than 0.1$\tau$. This clearly implies that spatially extended static fluctuations cause decline in the prediction accuracy of $S_i(t)$ by NT-GNN in the short time, and that this decline in the prediction accuracy is recovered because, in these quantities, effect of these spatially-extended static fluctuations are removed to a certain extent.

III. DISCUSSIONS

In conclusion, our model, BOTAN, predicts intricate processes in glassy dynamics from the static structure, with a precision exceeding the original GNN model by Bapst et al. (NT-GNN). Effectively the same dataset and neural network is necessary for achieving this level of prediction accuracy. The difference is that BOTAN learns relative motion via pair-distance changes computed directly from the dataset. BOTAN learns...
FIG. 5. Pearson correlation coefficients between predicted and actual particle propensities $S_i(t)$ are plotted between BOTAN and the original NT-GNN, as functions of time $t$ for $T = (a) 0.44$, (b) 0.50, (c) 0.56, and (d) 0.64. Predictions of “neighbor-relative propensities” $R_i(t)$ and $D_i(t)$ are also compared to the actual values. Errorbars show the median, best, and worst of five independently trained models.

details of relative motion in the short time via pair-distance changes which surpasses particle propensity in its number by an order of magnitude. BOTAN leverages the highly precise information it learns on relative motion for making prediction of particle propensity far more accurately than in previous studies [13, 14]. Surprisingly, when the pair-distance change and particle propensity are learnt together, BOTAN can incorporate the effect of spatially-extended fluctuations on the particle propensity into its prediction. In other words, although particle propensity is affected both by spatially extended fluctuations and structural rearrangement processes, BOTAN acquires the ability to distinguish between these two effects by complementing detailed information on the structural rearrangement, learnt via the relative motion.

The high predictive performance of BOTAN demonstrates that the static structure is more related to the relative dynamics based on pairs than the particle. Essentially, the static structure determines neighbor pairs with a lower energy barrier for “bond breakage.” Thus far, this concept has not been extensively analyzed for van der Waals molecular liquids, including the KALJ liquid. The neighbor pairs with a lower energy barrier for “bond breakage” appear to be autonomously identified by the model. BOTAN leverages this feature and appears to “know” better than any previous models the forthcoming transient state that the system will assume through the neighbor exchange processes. Therefore, it may contribute toward the realization of an intimate link between short-time dynamics and long-time structural relaxation, thus addressing the underlying time-scale gap. It is also a next key issue to identify the additional structural descriptor used by BOTAN for determining the pairs with lower energy barrier for bond breakage, and make use of it for developing cost-effective and stable model in line with attempts devoted toward the previous NT-GNN model [14].

BOTAN’s machine learning principles can be used for new versatile applications, because BOTAN can acquire ability to predict spatially-extended dynamics. As immediate applications, first, BOTAN may be used for studying the dynamics wherein spatially-extended motion is involved, including quasi-localized and phonon vibrations, deformation of glasses under external strains (also studied by Bapst et al. [13]), or more diverse problems including e.g. dense active matter [41].

Moreover, as BOTAN exhibits high prediction ability over millions of molecular dynamics simulations steps, its prediction result may possibly be leveraged as new “reaction co-ordinates” along which the fluctuations should be enhanced for efficient sampling of molecular trajectory. Extensions to the recently discussed machine-learning-aided sampling technique [42, 43] would be an intriguing direction.

As BOTAN belongs to an elementary class of GNNs, incorporation of recent high-profile models in the field of data science may be another interesting direction. One approach is to incorporate the glassy dynamics into dynamic GNNs [44, 45]. We can construct GNNs that dynamically follow the structural changeover in glasses by accumulating highly-precise short-time predictions by internally reconfiguring the edges.
This may also be fruitful to leverage explainable graph networks [46, 47] to determine the underlying principles GNN internally learns for raising precision of their predictions.

ACKNOWLEDGMENTS

We thank Y. Asahi, J. J. Molina, and T. Kawasaki for helpful discussions. H.S. and T.S. were supported by JSPS KAKENHI Grant Number JP19H05662 and by “Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures” and “High Performance Computing Infrastructure” in Japan (Project ID: jh220052-NAH). M.H. and T.S. were supported by “Advanced Research Infrastructure for Materials and Nanotechnology” from MEXT, Japan. Training and testing are performed using NVIDIA A100 Tensor Core GPUs (40GB SXM) on Wisteria/BDEC-01 Aquarius subsystem at Information Technology Center, University of Tokyo. For generation of the dataset for training and testing, usage of the following computational resources are acknowledged: Oakforest-PACS at the Joint Center for Advanced High Performance Computing (JCAHPC), Oakbridge-CX at Information Technology Center, University of Tokyo, Supercomputer System B at Institute for Solid State Physics, University of Tokyo, and MASAMUNE-IMR at Institute for Materials Research, Tohoku University.

METHODS

A. Dataset details

We employ a Kob–Andersen-type binary Lennard–Jones mixture in three dimensions [13, 34] which is usually given by an interatomic pairwise potential

\[ u(r) = 4\epsilon_{\mu\nu} \left( \frac{\sigma_{\mu\nu}}{r} \right)^{12} - \left( \frac{\sigma_{\mu\nu}}{r} \right)^{6}. \]

\( \mu, \nu \in \{A, B\} \) denotes the particle types, where the species A consists 80% of the total number. The interaction energy and particle size are defined as \( \epsilon_{AA} = 1.0, \epsilon_{AB} = 1.5, \epsilon_{BB} = 0.5, \sigma_{AA} = 1.0, \sigma_{AB} = 0.8, \) and \( \sigma_{BB} = 0.88 \). In this study, distances, time, and temperature are denoted in units of \( \sigma_{AA}, \tau = \sqrt{m\sigma_{AA}/\epsilon_{AA}}, \) and \( \epsilon_{AA}/k_B, \) with \( k_B \) the Boltzmann constant.

For improving the accuracy of numerical integration over a large number of steps, the pairwise potential is modified in the track of previous studies as [35, 36]

\[ U(r) = u(r) - u(r_c) - (r - r_c) \frac{du(r)}{dr} \bigg|_{r=r_c}, \]

so that the force and the potential vanish at the cutoff length \( r = r_c \). The cutoff lengths are set differently for different combinations of pair species depending on the combination of pair species as \( r_c = 2.5\sigma_{\mu\nu} \).

The procedure of dataset generation is similar to that in the previous study [13] except that the system size is fixed. For all the simulations, the number density of the system is fixed at \( N/V = 1.2 \). Time step of numerical integration is kept at \( \Delta t = 10^{-3} \) throughout. We set four target temperature, \( T = 0.44, 0.50, 0.56, \) and 0.64, for which the respective relaxation times \( \tau_\alpha \) are summarized in Table I. For each four target temperatures, 500 independent particle configurations are generated as follows. After keeping the temperature at \( T = 5.0 \) for the time lapse of \( t_0 = 10^3 \), the system is cooled rapidly (in \( t_{cool} = 20 \)) to the target temperature. The temperature is subsequently kept constant until the system reaches the steady state up to time scale of \( 40\tau_\alpha \). The final particle configuration thus obtained is used as the initial one at \( t = 0 \) in the production run for generating the dataset. From the 500 initial configurations, we have introduced the isofngerational ensemble [6, 37] by running 32 separate microcanonical (NVE) simulations all starting from the same configuration; however, the initial velocities are randomly given from the Maxwell-Boltzmann distribution at the target temperature. These simulations are conducted approximately up to \( 30\tau_\alpha \), where the particle configurations are sampled at logarithmically increasing time points. Among the 500 initial configurations, 400 configurations (12,800 runs) are used for training and the other 100 (3,200 runs) are for testing.

B. Machine learning model architecture

In this subsection, we explain the neural network architecture of BOTAN and NT-GNN, and also provide information that was not provided in the previous study conducted by Bapst et al. [13]. Let \( G(V, E) \) be a graph that includes a set of nodes \( V \) and edges \( E \). For a pair of nodes \( v, u \in V \), an edge between \( v \) and \( u \) is represented as \( e_{vu} \). The neighboring edges of a vertex \( v \in V \) are represented as \( N(v) \subset E \). Given input feature vectors \( h_v^m \) and \( h_e^m \) for a node \( v \in V \) and an edge \( e \in E \) respectively, the goal is to compute the corresponding output features \( h_v^{out} \) and \( h_e^{out} \). As shown in a block diagram in FIG. 6 and by denoting an encoder, a multi-layer perceptron (MLP) and a decoder as \( EN(\cdot) \), MLP(\cdot) and DE(\cdot), each layer to compute feature vectors for node \( v \) is formulated as follows. First, input feature vectors are encoded by respective encoders

\[ h_v^0 = EN(h_v^m), \quad h_e^0 = EN(h_e^m). \]

Then, update of the edge and node features via MLPs are repeated for \( n \) times

\[ h_v^{m+1} = MLP(h_v^{m-1} \oplus h^m_e \oplus h_v^0), \quad h_e^{m+1} = MLP(h_e^{m-1} \oplus \sum_{e \in N(v)} h_e^m \oplus h_e^0) \]

where \( m \) stands for the iteration index of \( n \) repeat cycle \( (0 < m < n) \). In each cycle, messages are passed between nodes and edges, wherein encoded feature vectors are concatenated together before every update. The decoders placed as the final layer yields the output features on each node and edge

\[ h_v^{out} = DE(h_v^0), \quad h_e^{out} = DE(h_e^0). \]
In practice, EN(•), MLP(•) and DE(•) are all implemented by two hidden layers of 64 units with rectified linear units (ReLU) non-linearity. At every repeat cycle of update on edges and nodes in Eqs. (5) and (6), information of the neighbor shell (inside the distance of 2σAA) is propagated. We set the number of this repetition as n = 7 so that the information propagates over the whole system (with a box length of 15.06) in a single epoch of learning.

In addition to the node output feature $\hat{h}_v^{\text{out}}$ which have been exploited to learn particle propensity in previous studies [13, 14], we introduced the decoder for the edge feature $\hat{h}_e^{\text{out}}$ in this model, which is an essential innovation in the present study. It is essential that edges receive messages from both neighbor nodes v,u in each repetitive cycle of message passing in Eq. (5), though this detail does not affect the results and was not explicitly addressed in the study by Bapst et al. [13].

In this study, a graph data for learning is constructed based on the particle configuration wherein the particles are represented by its nodes. Edges are assigned to pairs of particles i and j within a distance threshold value $r_{ij}(t = 0) \leq r_e$, so that the neighborhood relation can be represented. We choose a threshold length $r_e = 2.0σ_{AA}$ that has the best predictive performance on particle propensity [13]. Then these graphs are fed into the GNN by encoding particle types (A and B) and 3D relative positions between particles independently into the nodes and edges as single floating-point numbers, wherein the types (A and B) are converted to 0 and 1.

### C. Learning and inference procedure

The learning setups and hyperparameters in the present study are similar to those used by Bapst et al. [13]. Dataset is learned to minimize the loss function, wherein the neural network is optimized by Adam algorithm without norm regularization with the learning rate of $10^{-4}$ with a standard implementation in PyTorch. For learning on output features on each node and edge, loss functions are defined by

$$L_M = pL_S + (1 - p)L_E,$$  \hspace{1cm} (8)

wherein the prediction of the particle propensity $S_i(t)$ and distance change $E_{ij}(t)$ is performed simultaneously. $p$ is treated as a hyperparameter ($0 \leq p \leq 1$) as it decides the weight of learning between nodes and edges. While on the one hand, when we employ the pair-distance changes as the target quantity, this parameter is set to $p = 0$ (corresponding to Figs. [1] and [2], on the other hand, the model reduces to the previous NT-GNN [13]. For the simultaneous learning on nodes and edges, we make the choice of $p = 0.4$ in this study, which is justified by the ablation experiment in which Pearson correlation coefficients are calculated with changing $p$ (see Supplemental Information, Fig. S3).

$L_S$ and $L_E$ are standard $L_2$-norm loss functions, which may be written down as

$$L_S = \sum_{j=1}^{N} \left[ \langle \hat{S}_j(t;\{r_i\}) - \langle S_j(t) \rangle \rangle - \langle S_j(t) \rangle \rangle \right]^2,$$  \hspace{1cm} (9)

$$L_E = \sum_{j=1}^{N} \sum_{k=1}^{N} \left[ \langle \hat{E}_{jk}(t;\{r_i\}) - \langle E_{jk}(t) \rangle \rangle \right].$$  \hspace{1cm} (10)

Here, $\langle \cdot \rangle$ represents the average over the isoconfigurational ensemble and $\langle S_j(t) \rangle$ and $\langle E_{jk}(t) \rangle$ denote the values of particle propensity and pair-distance change predicted by the GNN from the initial configuration used for generating the isoconfigurational ensemble.

The data loaded for training are augmented by applying random rotation of the simulation box to the particle positions, as all the target quantities for learning are scalar variables with rotational invariance. In the training performed with NT-GNN, the minibatch sizes of the graph input are fixed at 5 and the number of epochs for learning are fixed at 1000, to ensure that overlearning does not occur in all the cases. The training scenarios performed with BOTAN (with edge targets), without the minibatching of the graph input, requires a larger number of epochs under the present setting of hyperparameters. Consequently, we need to stop learning at an epoch before overlearning occurs.

In all these training scenarios, all the particles (particle pairs) are learned simultaneously, regardless of the particle types. The training with NT-GNN over 1000 epochs required approximately 2 h when using one NVIDIA A100 Tensor Core GPU (40 GB SXM). It is essentially the same for BOTAN, except that there may exist a substantial overhead arising from computation of interparticle distances, of which the actual elapsed time depends on its implementation.
D. Other details

In Figs. 1, 3, and 4 color maps are plotted by interpolating the propensities (and the prediction error) of particles which are inside a specific cross section within the thickness of 0.8. In Fig. 1, on the other hand, neighbor pairs are selected so that their initial distance is lower than 1.35 and fits within the same cross section. The locally-averaged displacement fields in Fig. 4 are defined as an isoconfigurational ensemble average inside the first neighbor shell.

\[
\langle \Delta r^\text{new}(t) \rangle_{IC} = \left\langle \sum_j \left[ \mathbf{r}_j(t) - \mathbf{r}_j(0) \right] \Theta \left( r_c - r_{ij}(0) \right) \right\rangle_{IC},
\]

where \( \Theta(x) \) denotes Heaviside’s step function and \( r_c \) is the cutoff length of coarse-graining \( r_c \) set at 1.35. Each of them is plotted in a cross-section with a thickness of 1.2, where the 3D vectors are enlarged for better visibility.

The local strain tensor \( \epsilon_{\alpha\beta} \) in Eqs. (2) and (3) is chosen as the one that minimizes \( D_2(t)^2 \) with respect to \( \Lambda_{\alpha\beta} \), as proposed in the original paper by Falk and Langer \[39\]. We have taken the isoconfigurational ensemble averages of \( R_s(t) \) and \( D_2(t) \), and conducted the machine learning. This results in the choice of

\[
\Lambda_{\alpha\beta} = \sum_{x=y,z} Y^{-1}_{\beta\gamma} X_{\gamma\gamma}
\]

wherein the tensors \( X_{\alpha\beta} \) and \( Y_{\alpha\beta} \) denote 3 \times 3 matrices given by

\[
X_{\alpha\beta} = \left\langle \sum_{m:n:n} \left[ r^{\alpha\beta}_m(0) - r^{\alpha\beta}_n(t) \right] [r^{\alpha\beta}_m(0) - r^{\alpha\beta}_n(0)] \right\rangle_{IC}
\]

\[
Y_{\alpha\beta} = \sum_{m:n:n} \left[ r^{\alpha\beta}_m(0) - r^{\alpha\beta}_n(t) \right] [r^{\alpha\beta}_m(0) - r^{\alpha\beta}_n(0)].
\]

DATA AVAILABILITY

The source code used for training and evaluation is available at GitHub repository https://github.com/h3-Open-BDEC/pyg_botan. This code is an implementation of BOTAN using PyTorch Geometric \[48\], with extensions from the original code for TensorFlow provided by Bapst et al. \[13\]. Data for evaluations will also be made available via the hyperlink in the same GitHub repository for three years after the publication of this paper.

AUTHOR CONTRIBUTIONS

H.S. and T.Sh conceived and designed research. H.S., M.H., and T.Su developed the machine learning model. H.S. conducted the code development, numerical simulations, and machine learning. H.S. and M.H. analyzed the data. All the authors discussed the results. H.S. and M.H. wrote the manuscript with inputs from T.Su and T.Sh.

COMPETING INTERESTS

The authors declare no competing financial interests.

\[1\] C. A. Angell, Formation of glasses from liquids and biopolymers, Science 267, 1924 (1995)

\[2\] L. Berthier, G. Biroli, J.-P. Bouchaud, L. Cipelletti, and W. van Saarloos, Dynamical Heterogeneities in Glasses, Colloids, and Granular Media (Oxford University Press, 2011).

\[3\] R. Yamamoto and A. Onuki, Heterogeneous diffusion in highly supercooled liquids, Physical Review Letters 81, 4915 (1998).

\[4\] M. D. Ediger, Spatially heterogeneous dynamics in supercooled liquids, Annu. Rev. Phys. Chem. 51, 99 (2000).

\[5\] H. Tanaka, H. Tong, R. Shi, and J. Russo, Revealing key structural features hidden in liquids and glasses, Nature Reviews Physics 1, 333 (2019).

\[6\] A. Widmer-Cooper, H. Perry, P. Harrowell, and D. R. Reichman, Irreversible reorganization in a supercooled liquid originates from localized soft modes, Nature Physics 4, 711 (2008).

\[7\] S. S. Schoenholz, A. J. Liu, R. A. Riggelman, and J. Rottler, Understanding plastic deformation in thermal glasses from single-soft spot dynamics, Physical Review X 4, 031014 (2014).

\[8\] P. Ronhovde, S. Chakraborty, D. Hu, M. Sahu, K. K. Sahu, K. F. Kelton, N. A. Mauro, and Z. Nussinov, Detecting hidden spatial and spatio-temporal structures in glasses and complex physical systems by multisresolution network clustering, European Physical Journal E 34, 105 (2011).

\[9\] E. D. Cubuk, S. S. Schoenholz, J. M. Rieser, B. D. Malone, J. Rottler, D. J. Durian, E. Kaxiras, and A. J. Liu, Identifying structural flow defects in disordered solids using machine-learning methods, Physical Review Letters 114, 108001 (2015).

\[10\] S. S. Schoenholz, E. D. Cubuk, D. M. Sussman, E. Kaxiras, and A. J. Liu, A structural approach to relaxation in glassy liquids, Nature Physics 12, 469 (2016).

\[11\] S. S. Schoenholz, E. D. Cubuk, E. Kaxiras, and A. J. Liu, Relationship between local structure and relaxation in out-of-equilibrium glassy systems, Proceedings of the National Academy of Sciences of the United States of America 114, 263 (2017).

\[12\] E. Boattini, S. Marín-Aguilar, S. Mitra, G. Foffi, F. Smallenburg, and L. Filion, Averaging local structure to predict the dynamic propensity in supercooled liquids, Physical Review Letters 127, 088007 (2021).
[15] T. Xie and J. C. Grossman, Crystal graph convolutional neural networks for an accurate and interpretable prediction of material properties, Physical Review Letters 120, 145301 (2018).

[16] L. Chanussot, A. Das, S. Goyal, T. Lavril, M. Shuaibi, M. Riviere, K. Tran, J. Heras-Domingo, C. Hu, W. Hu, A. Palizhata, A. Sriram, B. Wood, J. Yoon, D. Parikh, C. L. Zitnick, and Z. Ulissi, Open catalyst 2020 (oc20) dataset and community challenges, ACS Catalysis 11, 6059 (2021).

[17] V. Fung, J. Zhang, E. Juarez, and B. G. Sumpter, Benchmarking graph neural networks for materials chemistry, npj Computational Materials 7, 84 (2021).

[18] S. Takamoto, C. Shinagawa, D. Motoki, K. Nakago, W. Li, T. Onodera, T. Ishii, T. Kudo, H. Ono, R. Sawada, R. Ishitan, M. Ong, T. Yamaguchi, T. Kataoka, A. Hayashi, N. Charoenphakdee, and T. Ibuka, Towards universal neural network potential for material discovery applicable to arbitrary combination of 45 elements, Nature Communications 13, 2991 (2022).

[19] Q. Wang and L. Zhang, Inverse design of glass structure with deep graph neural networks, Nature Communications 12, 3539 (2021).

[20] U. Buchenau, Y. M. Galperin, V. L. Gurevich, and H. R. Schober, Anharmonic potentials and vibrational localization in glasses, Physical Review B 43, 5039 (1991).

[21] H. Mizuno, H. Shiba, and A. Ikeda, Continuum limit of the vibrational properties of amorphous solids, Proceedings of the National Academy of Sciences of the United States of America 114, E9767 (2017).

[22] Y.-C. Hu and H. Tanaka, Origin of the boson peak in amorphous solids, Nature Physics 18, 669 (2022).

[23] N. Xu, V. Vitelli, A. J. Liu, and S. R. Nagel, Anharmonic and quasi-localized vibrations in jammed solids – modes for mechanical failure, EPL 90, 56001 (2010).

[24] A. Ghosh, V. Chikkadi, P. Schall, and D. Bonn, Connecting structural relaxation with the low frequency modes in a hard sphere colloidal glass, Physical Review Letters 107, 188303 (2011).

[25] P. W. Battaglia, R. Pascanu, M. Lai, D. Rezende, and K. Kavukcuoglu, Interaction networks for learning about objects, relations and physics, Advances in Neural Information Processing Systems 29, 4502 (2016).

[26] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, and R. Pascanu, Relational inductive biases, deep learning, and graph networks, arXiv:1806.01261 [cs.LG] (2018).

[27] G. DeZoort, S. Thais, J. Duarte, V. Razavimaleki, M. Atkinson, I. Ojalo, M. Neubauer, and P. Elmer, Charged particle tracking via edge-classifying interaction networks, Computing and Software for Big Science 5, 26 (2021).

[28] P. Yunker, Z. Zhang, K. B. Aptowicz, and A. G. Yodh, Irreversible rearrangements, correlated domains, and local structure in aging glasses, Physical Review Letters 103, 115701 (2009).

[29] V. Chikkadi and P. Schall, Nonaffine measures of particle displacements in sheared colloidal glasses, Physical Review E 85, 031402 (2012).

[30] F. Shimizu, S. Ogata, and J. Li, Theory of shear banding in metallic glasses and molecular dynamics calculations, Materials Transactions 48, 2923 (2007).

[31] T. Kawasaki and A. Onuki, Slow relaxations and stringlike jump motions in fragile glass-forming liquids: Breakdown of the stokes-einstein relation, Physical Review E 87, 012313 (2013).

[32] H. Shiba, Y. Yamada, T. Kawasaki, and K. Kim, Unveiling dimensionality dependence of glassy dynamics: 2d infinite fluctuation eclipses inherent structural relaxation, Physical Review Letters 117, 245701 (2016).

[33] B. Guiselin, C. Scalliet, and L. Berthier, Microscopic origin of excess wings in relaxation spectra of supercooled liquids, Nature Physics 18, 468 (2022).

[34] W. Kob and H. C. Andersen, Testing mode-coupling theory for a supercooled binary lennard-jones mixture: The van hove correlation function, Physical Review E 51, 4626 (1995).

[35] S. Toxaerud and J. C. Dyre, Communication: Shifted forces in molecular dynamics, The Journal of Chemical Physics 134, 081102 (2011).

[36] M. Shimada, H. Mizuno, and A. Ikeda, Anomalous vibrational properties in the continuum limit of glasses, Physical Review E 97, 022609 (2018).

[37] A. Widmer-Cooper, P. Harrowell, and H. Fynneuer, How reproducible are dynamic heterogeneities in a supercooled liquid?, Physical Review Letters 93, 135701 (2004).

[38] H. Shiba, T. Kawasaki, and K. Kim, Local density fluctuation governs divergence of viscosity underlying elastic and hydrodynamic anomalies in a 2d glass-forming liquid, Physical Review Letters 123, 265501 (2019).

[39] M. L. Falk and J. S. Langer, Dynamics of viscoplastic deformation in amorphous solids, Physical Review E 57, 7192 (1998).

[40] H. L. Peng, M. Z. Li, and W. H. Wang, Structural signature of plastic deformation in metallic glasses, Physical Review Letters 106, 135503 (2011).

[41] C. Bechinger, R. D. Leonardo, H. Löwen, C. Reichhardt, G. Volpe, and G. Volpe, Active particles in complex and crowded environments, Reviews of Modern Physics 88, 045006 (2016).

[42] F. Noé, S. Olsson, J. Köhler, and H. Wu, Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning, Science 365, eaaw1147 (2019).

[43] M. Gabrie, G. M. Rotskoff, and E. Vanden-Eijnden, Adaptive monte carlo augmented with normalizing flows, Proceedings of the National Academy of Sciences of the United States of America 119, e2109420119 (2022).

[44] A. Pareja, G. Domeniconi, J. Chen, T. Ma, T. Suzumura, H. Kanezashi, T. Kaler, T. B. Schardl, and C. E. Leiserson, EvolveGCN: Evolving graph convolutional networks for dynamic graphs, Proceedings of the 34th AAAI Conference on Artificial Intelligence 34, 5363 (2020).

[45] J. Skarding, B. Gabrys, and K. Musial, Foundations and modeling of dynamic networks using dynamic graph neural networks: A survey, IEEE Access 9, 79143 (2021).

[46] Z. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, Fast graph representation learning with PyTorch Geometric, in ICLR Workshop on Representation Learning on Graphs and Manifolds (2019).