Supplementary Information – Structural Descriptors and Information Extraction from X-ray Emission Spectra: Aqueous Sulfuric Acid

E. A. Eronen,[1,*] A. Vladyka,[1] Ch. J. Sahle,[2] and J. Niskanen[1,†]

[1] *University of Turku, Department of Physics and Astronomy, FI-20014 Turun yliopisto, Finland*
[2] *ESRF, The European Synchrotron, 71 Avenue des Martyrs, CS40220, 38043 Grenoble Cedex 9, France*

(Dated: August 8, 2024)

SUPPLEMENTARY INFORMATION

Spectrum simulation plane wave cutoff convergence check

We checked the convergence of the spectrum with respect to the plane wave cutoff used in the simulations. The results from one 0.9 M structure, seen in Figure 1, show that convergence is reached after 500 eV. We used 600 eV cutoff in the simulations of this work.

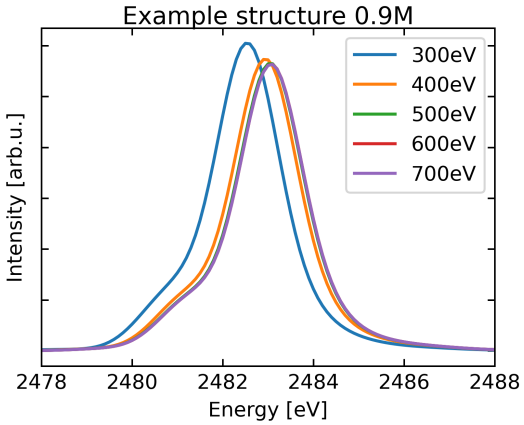

FIG. 1. Spectrum from one sample structure calculated using different plane wave cutoff values. The chosen 600 eV spectrum is on top of both the 500 eV one and the 700 eV one: convergence has been reached.

On the role of standardization of the input

A feedforward NN passes information from a layer (contained by input $\mathbf{x}$) to obtain activation values of the neurons of the next layer (contained by output $\mathbf{z}$) by applying function

$$\mathbf{z} = g(\mathbf{A}\mathbf{x} + \mathbf{b}) \tag{1}$$

where the weight matrix $\mathbf{A}$ and bias vector $\mathbf{b}$ are characteristic to the particular neuron layers, and are optimized during training of the network. The activation function g is taken element-wise for $\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{b}$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_{N_1} \end{bmatrix} = \begin{bmatrix} a_{11} & \dots & a_{1N_{\text{in}}} \\ \vdots & \ddots & \vdots \\ a_{N_1 1} & \dots & a_{N_1 N_{\text{in}}} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{N_{\text{in}}} \end{bmatrix} + \begin{bmatrix} b_1 \\ \vdots \\ b_{N_1} \end{bmatrix}.$$

* eemeli.a.eronen@utu.fi

† johannes.niskanen@utu.fi

The activation of the output neuron k is the activation function $z_k = g(y_k)$ of the summation

$$y_k = a_{k1}x_1 + \cdots + a_{kN_{\text{in}}}x_{N_{\text{in}}} + b_k = \sum_{i=1}^{N_{\text{in}}} a_{ki}x_i + b_k. \tag{2}$$

Now, say that these $\mathbf{A}$ and $\mathbf{b}$ have been found for each layer during training with non-standardized feature values. When the input features i are then z-score standarized using respective mean μ_i and standard deviation $\sigma_i > 0$, the input vector $\mathbf{x}'$ has components

$$x'_i = \frac{x_i - \mu_i}{\sigma_i}, \qquad i = 1 \ldots N_{\text{in}}. \tag{3}$$

In the first layer, each element a_{ki} of the kth row of $\mathbf{A}$ have the freedom to take new value $a'_{ki} = \sigma_i a_{ki}$. Likewise, the component b_k of the vector $\mathbf{b}$ has its freedom take value $b'_k = b_k + \sum_i a_{ki}\mu_i$. These adaptations for the newly formed $\mathbf{A}'$ and $\mathbf{b}'$ for the input layer yield

$$\begin{aligned}
y'_k &= \mathbf{A}'\mathbf{x}' + \mathbf{b}' && (4)\\
&= \sum_{i=1}^{N_{\text{in}}} a'_{ki}x'_i + b'_k && (5)\\
&= \sum_{i=1}^{N_{\text{in}}} \sigma_i a_{ki}\frac{x_i - \mu_i}{\sigma_i} + b_k + \sum_{i=1}^{N_{\text{in}}} a_{ki}\mu_i && (6)\\
&= \sum_{i=1}^{N_{\text{in}}} a_{ki}x_i + b_k + \sum_{i=1}^{N_{\text{in}}} a_{ki}\mu_i - \sum_{i=1}^{N_{\text{in}}} a_{ki}\mu_i && (7)\\
&= \sum_{i=1}^{N_{\text{in}}} a_{ki}x_i + b_k = y_k && (8)
\end{aligned}$$

and the same $z_k = g(y_k)$ as without standardization. Thus the same activation of the first-layer neurons $\mathbf{z}$ (and the output of the whole NN) can always be achieved by adaptation of $\mathbf{A}$ and $\mathbf{b}$ of the input layer to the new scaling of data. In other words, transformation (3) will not exclude any solution for training, *i.e.* the process of the optimization of matrices $\mathbf{A}$ and vectors $\mathbf{b}$ guided and decided by emulation performance of the NN for the training data set. We note that linear scaling of feature values x_i will be a special case of Equation (3). Moreover the result remains regardless of whether scaling is done individually for each i or in coordination for groups of features i. This is because for each output neuron k and each input feature i there is a specific weight matrix element a_{ki} available to adapt.

However, feature scaling has an effect of the L_2 regularization during NN training. This regularization punishes for large $|a_{nm}|$ during the optimization of the weight matrices $\mathbf{A}$. Therefore the input feature absolute values $|x_i|$ must be kept in the same order of magnitude for all features i, for them all to have an equal chance to affect the summation of Equation (2). In other words, non-standardized input features would generate a systematic bias towards the information in input features larger in their numeric absolute values. We note that the z-score standardization is not only standard practice, but also a bijective mapping. Thus information is not lost in the process of z-score standardization, and it can be converted back by the respective inverse transformation.

Descriptor hyperparameter grids

The model selection for each descriptor type was carried out as a joint randomized grid search of the relevant descriptor and neural network (emulator) parameters. Every atom (S, O and H) of any molecule within 6 Å of the emission site was included in constructing the descriptor vector, except in the case of CM. A single center, the emission site S_{em}, was used with the descriptors LMBTR, SOAP, ACSF and GT. When applicable, the cut of radius "r cut" was selected as 6.5 Å to include all the atoms which were part of the spectrum calculations.

LMBTR

We used the implementation of the DScribe package [1, 2] for the LMBTR descriptor [3]. The studied parameter grid is presented in Table I. Our previous knowledge of the descriptor [4] helped with the selection of the grid. Some unnecessary features (always zero as our system is non-periodic) were removed from the output.

TABLE I. The LMBTR parameter grid. Where applicable, the selected parameter value is shown in bold.

Parameter	Distances (k2)	Angles (k3)
geometry function	distance	angle
grid min	0.7 Å	0°
grid max	6.5 Å	180°
grid n	20,**40**,60,80,100	5,10,**20**,30,40
grid σ	0.2,0.4,**0.6**,0.8,1.0 Å	9,12,**15**,18,21°
weighting function	unity	exp
weighting scale	—	0.8,1.0,1.2,**1.4**,1.6
weighting threshold	—	1e–8

SOAP

We used the implementation of the DScribe package [1, 2] for the SOAP [5] descriptor. The studied parameter grid is presented in Table II.

TABLE II. The SOAP parameter grid. Where applicable, the selected parameter value is shown in bold.

Parameter	Values
r cut	6.5 Å
n max	4,5,6,7,**8**
l max	4,5,6,7,**8**
sigma	0.25,**0.5**,0.75,1.0 Å
rbf	gto
weighting	pow
c	**0.25**,0.5,1.0,2.0,4.0
d	0.25,0.5,**1.0**,2.0,4.0
m	**2**,4,6,8
r0	1,2,3,**4**

ACSF

We used the implementation of the DScribe package [1, 2] for the ACSF [6] descriptor. The studied parameter grid, inspired by Nguyen and co-workers [7], is presented in Table III. We used G^2 and G^4 to include both two-body and three-body (radial and angular) interactions. For G^2 we used "R_s n" linearly spaced values from 0.7 Å to 6.5 Å and $\eta = 12.5/(R_s^2)$. For G^4 we always had both $\lambda = -1$ and $\lambda = 1$, and $\zeta = 2^x$ where x is a range of integers from 0 to some value with increments of 1. The parameter η had "η n" values from "η min" to "η max" placed on a logarithmic grid.

TABLE III. The ACSF parameter grid. Where applicable, the selected parameter value is shown in bold. "R_s n" is for the G^2 function and the rest are for the G^4.

Parameter	Values
r cut	6.5 Å
R_s n	10,20,30,40,50,**60**,70,80
η min	0.0001,0.001,**0.01**,0.1
η max	**1.0**,2.0,3.0,4.0
η n	**5**,10,15,20,25,30
ζ min	1
ζ max	32,**64**,128,256
λ	-1 and 1

GT

The parameter grid for the self-implemented descriptor GT [8] is presented in Table IV. Each of the three components (scalar, vector and tensor) had the same minimum and maximum gaussian width values. The number of linearly spaced widths between the minimum and the maximum ("n") was unique for each component.

TABLE IV. The GT parameter grid. Where applicable, the selected parameter value is shown in bold.

Parameter	Values
r cut	6.5 Å
width min	0.025,0.05,0.1,0.2,**0.4** Å
width max	4,5,6,**7**,8,9,10 Å
scalar n	10,25,50,75,100,**125**,150
vector n	10,20,40,**60**,80,100
tensor n	10,20,40,60,**80**,100

CM

The parameter grid for the self-implemented descriptor CM [9, 10] is presented in Table V. In our implementation the columns and rows of the initial full Coulomb matrix are first arranged by grouping the elements. These groups are then individually distance sorted with respect to the emission site. For each element S, O and H, only a specific number of the closest atoms (with respect to the emission site S_{em}) are included in the final matrix. If a given structure had less atoms of a given element than specified in the grid, the corresponding matrix elements were zeroed. We constructed the used descriptor vector by flattening the upper triangle (excluding the diagonal) of the Coulomb matrix.

TABLE V. The CM parameter grid. The selected parameter value is shown in bold.

Parameter	Values
sulfur n	1,2,3,4,5,6,**7**,8,9,10,11,12,13,14,15
oxygen n	4,6,8,10,12,**14**,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50
hydrogen n	2,6,10,**14**,18,22,26,30,34,38,42,46,50,54,58,62,66

MBDF

The parameter grid for the MBDF developed and implemented by Khan *et al.* [11] is presented in Table VI. Excluding the cutoff radius "r cut", the internal parameters of the descriptor selected by the original authors were assumed to be suitable. Ideally, we would have included these in the search, but this would have made the model selection computationally too heavy as building the feature vector required significantly more CPU time than the other descriptors. Instead, we opted for an approach similar to that applied to CM. For each atom (row), the initial matrix had a total of six features (columns). The rows of the matrix are first arranged by grouping the elements. These groups are then individually distance sorted with respect to the emission site $\mathrm{S_{em}}$. For each element S, O and H, only a specific number rows (obtained from the closest atoms with respect to the emission site) were included in the final matrix. In the end, all the atoms (within 6 Å of the emission site) are included in the descriptor through at least one row, as each row contains information about the local environment within "r cut" of one specific center atom. If a given structure had atoms of a given element than specified in the grid, the corresponding matrix elements were zeroed.

The authors also introduced a grid-based variant, which is independent of the number of atoms. The variant caused an increase in the dimensionality of the feature vector including a notable portion of zeros, and was left out of this work. At the time of download (2023-11-08), the provided code produced six features for each included center atom.

TABLE VI. The MBDF parameter grid. Where applicable, the selected parameter value is shown in bold.

Parameter	Values
r cut	6.5 Å
sulfur n	1,2,3,4,5,6,7,8,9,10,11,12,**13**,14,15
oxygen n	4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,**36**,38,40,42,44,46,48,50
hydrogen n	2,**6**,10,14,18,22,26,30,34,38,42,46,50,54,58,62,66

Hyperparameter grid search R^2-score distributions

The cross validation R^2-score distributions and the number of tested hyperparameter combinations for each of the six descriptors is shown in Figure 2.

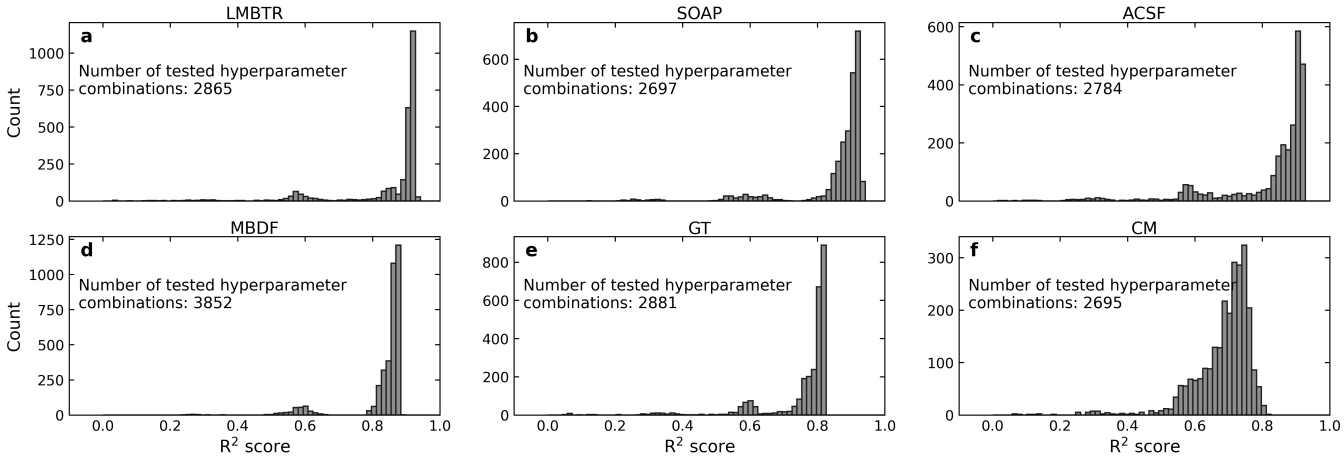

FIG. 2. **a–f:** The CV R^2-score distributions and the number of tested hyperparameter combinations for each of the six descriptors. We used the same bins for each panel.

Visualization of the performance of the best model with standardized spectra

Visualization of the performance of the best NN–LMBTR model on the z-score standardized test set spectra similar to Figure 2 of the main text is shown in Figure 3.

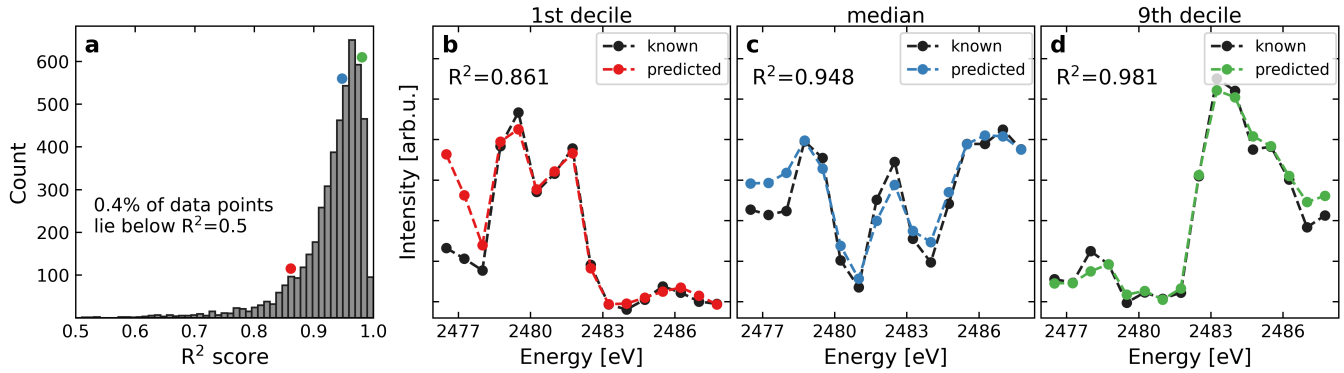

FIG. 3. Performance of the best NN–LMBTR on the test set with z-score standardized spectra. **a**: Distribution of R^2 scores for each data point. Examples of known and predicted spectra with R^2 score closest to **b:** the 1st decile, **c:** the median, and **d:** the 9th decile. For each of the three cases, the location along the R^2 distribution is shown in panel **a** as a correspondingly colored circle.

[1] Lauri Himanen, Marc O. J. Jäger, Eiaki V. Morooka, Filippo Federici Canova, Yashasvi S. Ranawat, David Z. Gao, Patrick Rinke, and Adam S. Foster. DScribe: Library of descriptors for machine learning in materials science. Computer Physics Communications, 247:106949, 2020. doi:10.1016/j.cpc.2019.106949.

[2] Jarno Laakso, Lauri Himanen, Henrietta Homm, Eiaki V. Morooka, Marc O. J. Jäger, Milica Todorović, and Patrick Rinke. Updates to the DScribe library: New descriptors and derivatives. The Journal of Chemical Physics, 158(23):234802, 2023. doi:10.1063/5.0151031.

[3] Haoyan Huo and Matthias Rupp. Unified representation of molecules and crystals for machine learning. Machine Learning: Science and Technology, 3(4):045017, 2022. doi:10.1088/2632-2153/aca005.

[4] Eemeli A Eronen, Anton Vladyka, Florent Gerbon, Christoph J Sahle, and Johannes Niskanen. Information bottleneck in peptide conformation determination by x-ray absorption spectroscopy. Journal of Physics Communications, 8(2):025001, 2024. doi:10.1088/2399-6528/ad1f73.

[5] Albert P. Bartók, Risi Kondor, and Gábor Csányi. On representing chemical environments. Physical Review B, 87:184115, 2013. doi:10.1103/PhysRevB.87.184115.

[6] Jörg Behler. Atom-centered symmetry functions for constructing high-dimensional neural network potentials. The Journal of Chemical Physics, 134(7):074106, 2011. doi:10.1063/1.3553717.

[7] Thuong T. Nguyen, Eszter Székely, Giulio Imbalzano, Jörg Behler, Gábor Csányi, Michele Ceriotti, Andreas W. Götz, and Francesco Paesani. Comparison of permutationally invariant polynomials, neural networks, and Gaussian approximation potentials in representing water interactions through many-body expansions. Journal of Chemical Physics, 148(24):241725, 2018. doi:10.1063/1.5024577.

[8] Anand Chandrasekaran, Deepak Kamal, Rohit Batra, Chiho Kim, Lihua Chen, and Rampi Ramprasad. Solving the electronic structure problem with machine learning. npj Computational Materials, 5(1):22, 2019. doi:10.1038/s41524-019-0162-7.

[9] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld. Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. Physical Review Letters, 108:058301, 2012. doi:10.1103/PhysRevLett.108.058301.

[10] Katja Hansen, Franziska Biegler, Raghunathan Ramakrishnan, Wiktor Pronobis, O. Anatole von Lilienfeld, Klaus-Robert Müller, and Alexandre Tkatchenko. Machine Learning Predictions of Molecular Properties: Accurate Many-Body Potentials and Nonlocality in Chemical Space. The Journal of Physical Chemistry Letters, 6(12):2326–2331, 2015. doi:10.1021/acs.jpclett.5b00831.

[11] Danish Khan, Stefan Heinen, and O. Anatole von Lilienfeld. Kernel based quantum machine learning at record rate: Many-body distribution functionals as compact representations. The Journal of Chemical Physics, 159(3):034106, 2023. doi:10.1063/5.0152215.