

# Supporting Information for: Deep Learning of Activation Energies

Colin A. Grambow, Lagnajit Pattanaik, and William H. Green\*

*Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge,  
Massachusetts 02139, United States*

E-mail: [whgreen@mit.edu](mailto:whgreen@mit.edu)

## S1. Neural network description

**Architecture.** The neural network architecture extends the `chemprop` directed message passing neural network (D-MPNN) framework by Yang et al.,<sup>S1</sup> which uses messages associated with directed edges instead of vertices. The following description is similar to that by Yang et al. but with important differences related to encoding reactions.

The same D-MPNN operates on the reactant and product graphs,  $G_R$  and  $G_P$ , separately to create a learned representation for each atom in the reactant and each atom in the product. Hydrogens are explicit in the graphs because they are often directly involved in the reactions. We subtract the representations for corresponding atoms between reactants and products from each other to generate a reaction embedding for each atom. We then aggregate these embeddings prior to the final activation energy prediction.

To operate on a molecular graph,  $G = (V, E)$  with vertices (atoms)  $V$  and edges (bonds)  $E$ , we require initial atom features  $\{x_v \mid v \in V\}$  and initial bond features  $\{e_{vw} \mid vw \in E\}$ . The atom features comprise a one-hot encoding of the atomic number, the degree, the formal charge, the chiral tag, the total number of hydrogens, and the hybridization; an aromaticity flag; the atomic mass; and whether the atom is in a ring of size  $s$  for  $s \in [3, 10]$ . The bond features indicate whether the bond is a single, double, triple, or aromatic bond; whether it is conjugated; whether it is in a ring; whether it is in a ring of size  $s$  for  $s \in [3, 10]$ ; and they contain a one-hot encoding of the bond stereochemistry. Since the ring membership features for atoms and bonds are one-hot vectors, they are able to encode all different-size rings that they are part of. We obtained all of the features using RDKit.<sup>S2</sup>

The following illustrates the message passing procedure. Note that some layers may include bias parameters, but the equations do not show them explicitly. We obtain the initial hidden state of a bond  $vw$  in an embedding operation given by

$$h_{vw}^0 = \tau(W_i \text{cat}(x_v, e_{vw})) \tag{S1}$$

where  $\tau(\cdot)$  is the ReLU activation function,  $W_i \in \mathbb{R}^{h \times (h_x + h_e)}$  is a learned matrix, and  $\text{cat}(x_v, e_{vw}) \in \mathbb{R}^{h_x + h_e}$  represents the concatenation of atom and bond features.  $h_x$  and  $h_e$  are the sizes of the initial atom and bond features, respectively. We determined the optimal hidden size to be  $h = 1800$  using the hyperparameter optimization procedure described later. The network calculates messages at the next time step as

$$m_{vw}^{t+1} = \sum_{k \in \{N(v) \setminus w\}} h_{kv}^t \quad (\text{S2})$$

where  $N(v)$  denotes the neighbors of atom  $v$ . The hidden state is updated by

$$h_{vw}^{t+1} = \tau(h_{vw}^0 + W_m m_{vw}^{t+1}) \quad (\text{S3})$$

where  $W_m \in \mathbb{R}^{h \times h}$  is another learned matrix and adding  $h_{vw}^0$  connects every hidden state to its original embedding. This proceeds iteratively for  $t \in \{1, \dots, T\}$ , and we set  $T = 5$ . We then convert bond fingerprints to atom fingerprints according to

$$m_v = \sum_{w \in N(v)} h_{vw}^T \quad (\text{S4})$$

$$h_v = \tau(W_a \text{cat}(x_v, m_v)) \quad (\text{S5})$$

where  $W_a \in \mathbb{R}^{h \times (h_x + h)}$  is a third learned matrix. Equations (S4) and (S5) are another message passing step, so the total number of message passing iterations is  $T + 1 = 6$ . We apply the operations in (S1)–(S5) to both the reactant and the product to yield  $h_v^{(R)}$  and  $h_v^{(P)}$ , respectively, for all atoms  $v$  in the molecular graph.

Next, we obtain the embedded difference atom fingerprints as

$$d_v = \tau(W_d (h_v^{(P)} - h_v^{(R)})) \quad (\text{S6})$$

where  $W_d \in \mathbb{R}^{h \times h}$  is a learned matrix. We sum the difference fingerprints to obtain a feature

vector for the reaction

$$r = \sum_{v \in G} d_v \quad (\text{S7})$$

Before generating an estimate for the activation energy, we calculate 200 global molecular features using RDKit<sup>S2</sup> for both the product and the reactant and append their difference to the reaction feature vector

$$\tilde{r} = \text{cat}(r, f_P - f_R) \quad (\text{S8})$$

where  $f_P$  and  $f_R$  are the product and reactant RDKit features, respectively. The purpose of these features is to capture global structural information in addition to the local information that is built up in the message passing steps. See Ref. S1 for more information.

Finally, the reaction feature vector with a linear activation enables estimation of the activation energy

$$\hat{E}_a = w_a^\top \tilde{r} \quad (\text{S9})$$

where  $w_a \in \mathbb{R}^{h+200}$  is a learned vector. We observed that a multitask prediction of both the activation energy and the enthalpy of reaction significantly improves the activation energy estimate. Therefore, the model has a second output to predict the enthalpy of reaction

$$\Delta \hat{H}_r = w_e^\top \tilde{r} \quad (\text{S10})$$

which is supplied during training but no longer used during evaluation.

**Training and hyperparameter optimization.** We partition the data into training, validation, and testing sets using a scaffold split, which bins the data based on the Murcko scaffolds of the reactants calculated by RDKit.<sup>S2</sup> Ref. S1 describes the exact partitioning procedure. To obtain a better measure of model performance, we use a 10-fold cross-validation approach. The validation data sets, used for hyperparameter optimization and early stopping, consist of 5% of the available data. Even though the model produces  $\hat{E}_a$  and  $\Delta \hat{H}_r$  as

outputs, we only use the error in  $\widehat{E}_a$  to determine early stopping. The main paper shows the variation in the training and test data fractions. We schedule the learning rate as follows: a linear learning rate increase from the initial learning rate to the maximum learning rate over a given number of warm-up epochs followed by an exponential decrease to the final learning rate over the course of the remaining epochs.

Training proceeds in two parts. First, we train the base model with the low-level B97-D3/def2-mSVP data. We then initialize the parameters of the final model using those of the base model and train the final model on the high-level  $\omega$ B97X-D3/def2-TZVP data. This transfer learning approach makes better use of all available data and enables improved accuracy of the final model.

We determine the architecture and other hyperparameters using the hyperparameter optimization code supplied with the `chemprop` package.<sup>S1</sup> In addition to the hidden size,  $h$ , and other architectural parameters, we optimize several training hyperparameters including the batch size, the number of epochs, the initial learning rate, the maximum learning rate, the final learning rate, and the number of warmup epochs. Table S1 shows the optimized parameters.

**Table S1: Optimized training hyperparameters.**

Hyperparameter	Base Model	Final Model
Batch size	50	10
Number of epochs	80	60
Initial learning rate	$10^{-5}$	$10^{-4}$
Maximum learning rate	$10^{-3}$	$10^{-4}$
Final learning rate	$10^{-5}$	$10^{-6}$
Number of warm-up epochs	3	1

## S2. RMG families

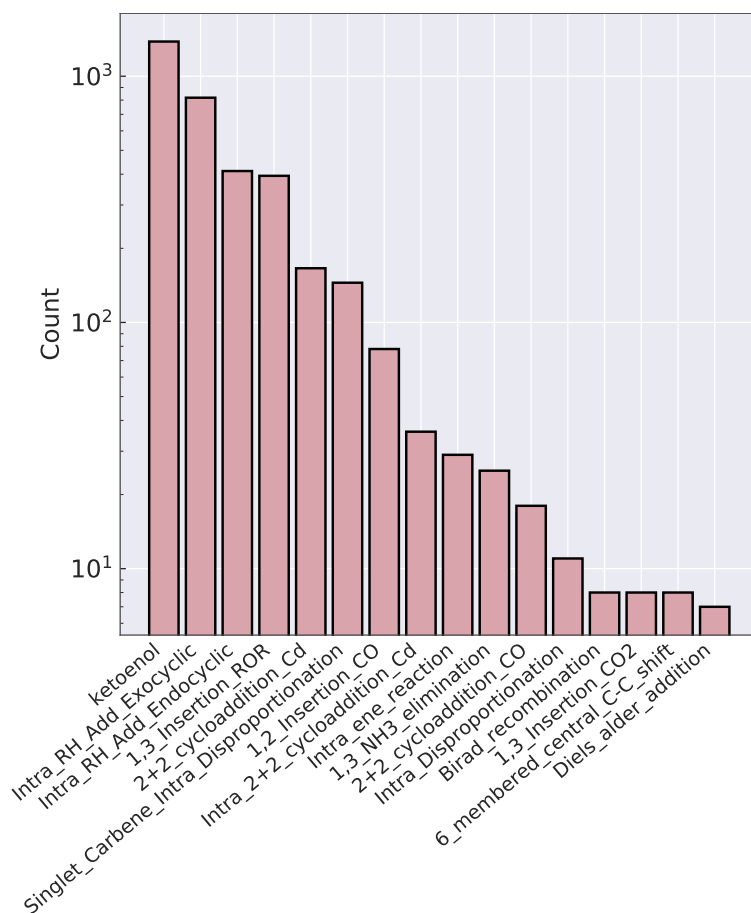


Figure S1: Number of reactions that match each RMG family.

### S3. Reaction type examples

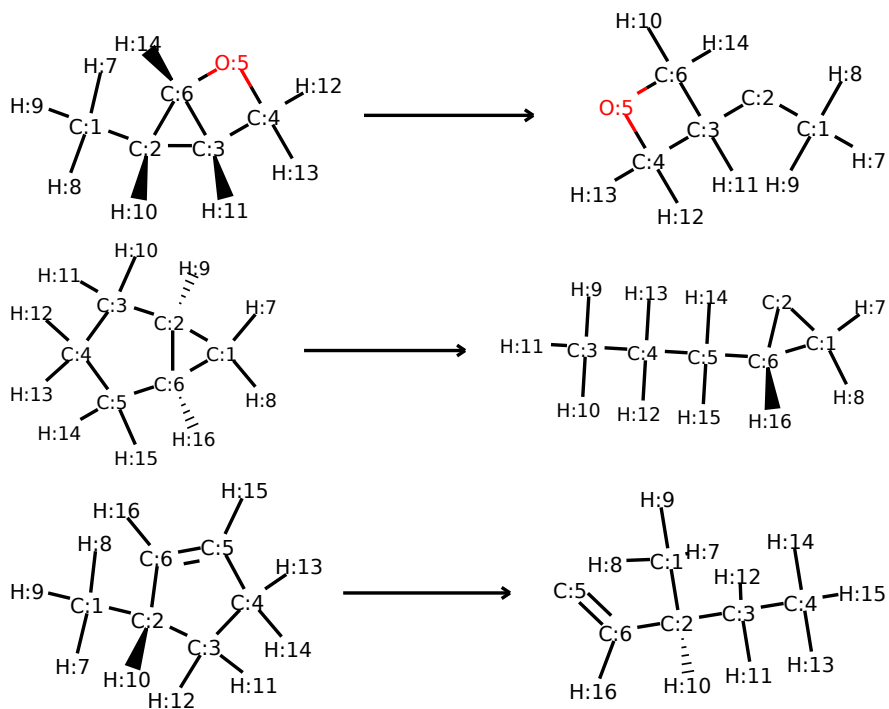


Figure S2: Reaction examples of +C-H, -C-H, -C-C type.

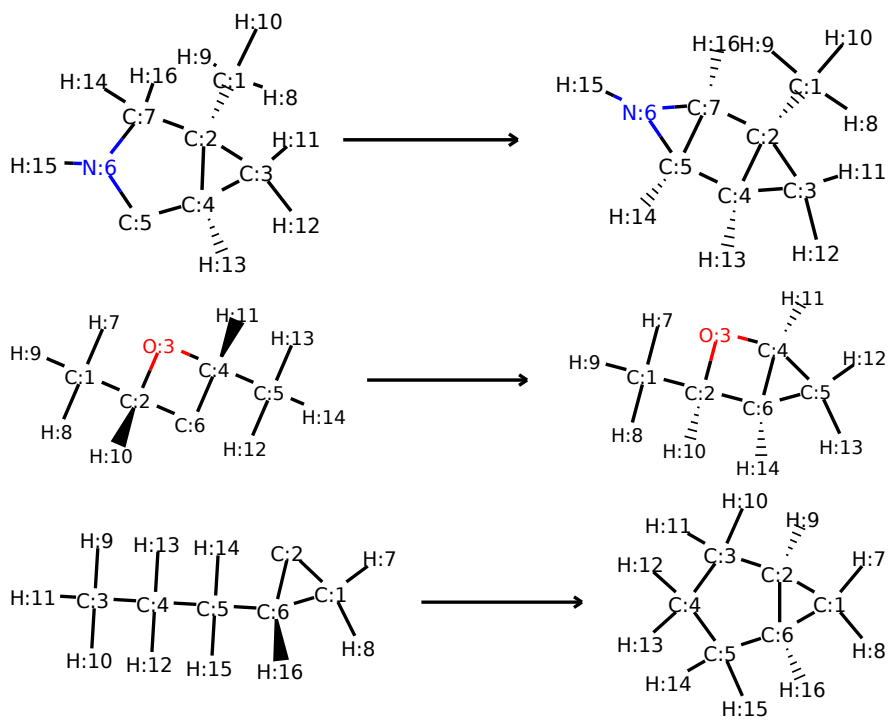


Figure S3: Reaction examples of +C-H, -C-H, +C-C type.

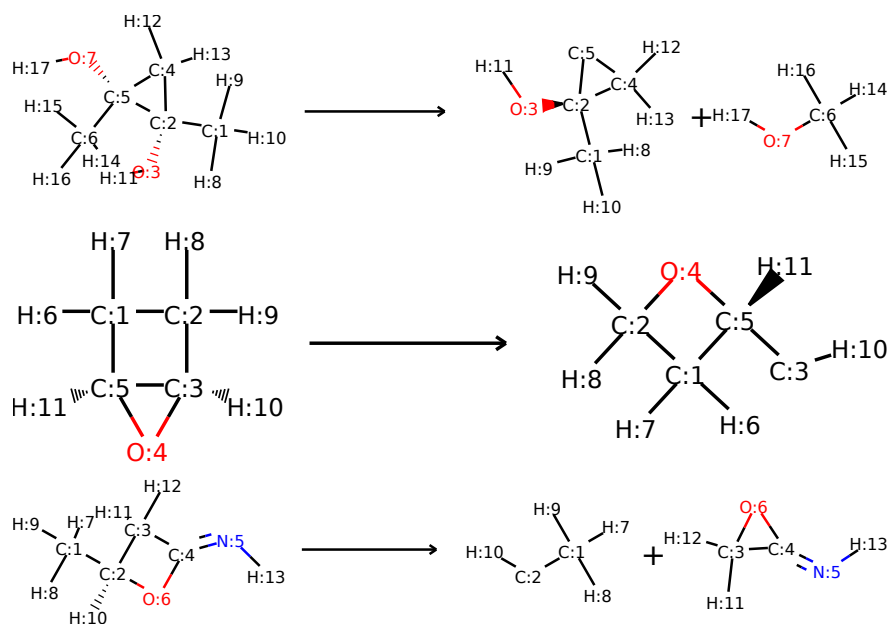


Figure S4: Reaction examples of +C-O, -C-C, -C-O type.

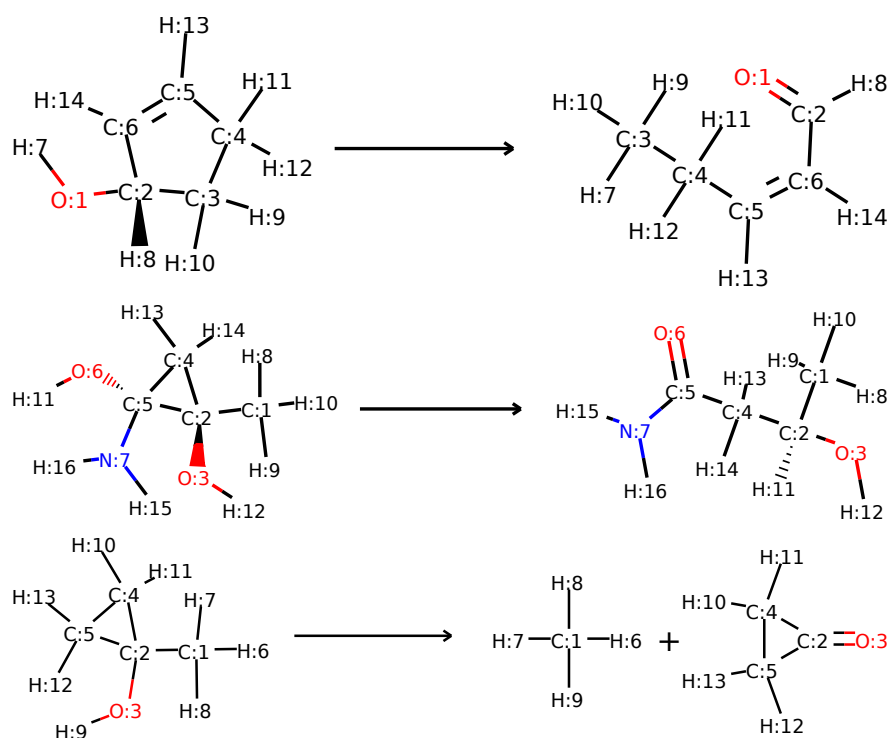


Figure S5: Reaction examples of +C-H, +C=O, -O-H, -C-C, -C-O type.



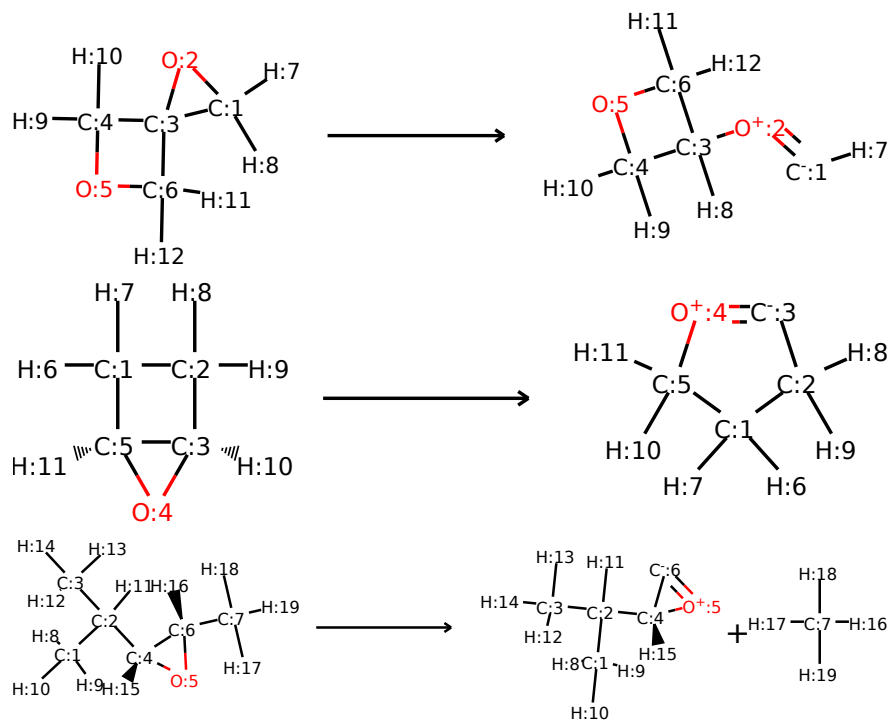


Figure S6: Reaction examples of +C-H,+C=O,-C-H,-C-C,-C-O type.

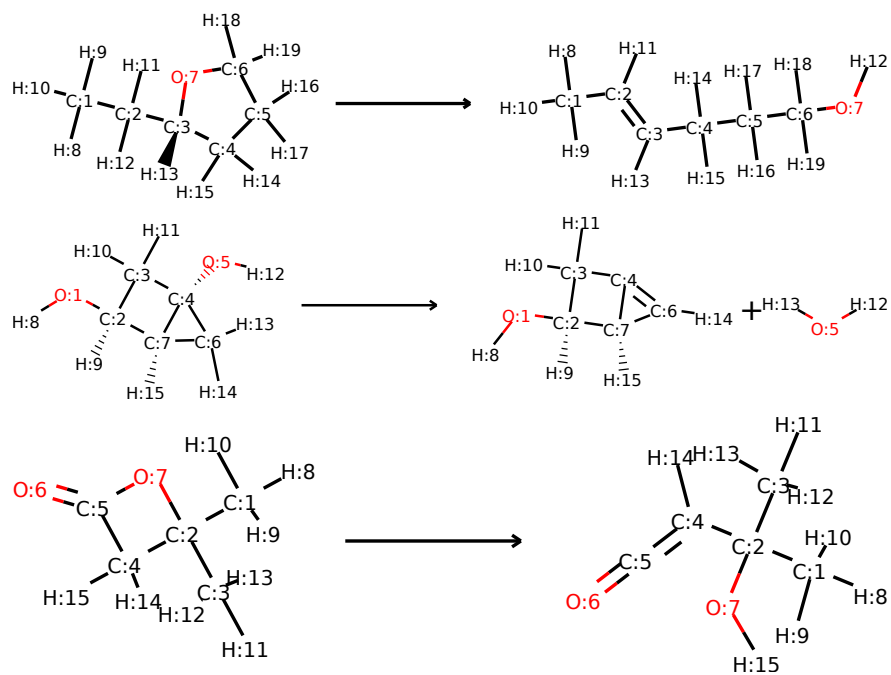


Figure S7: Reaction examples of +O-H,+C=C,-C-H,-C-C,-C-O type.

#### S4. MAE split by number of heavy atoms

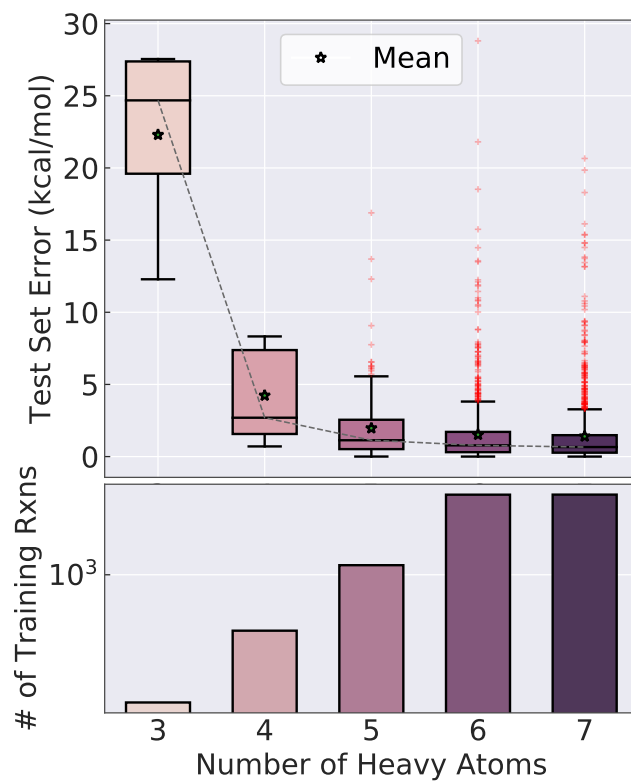


Figure S8: MAE split by the number of heavy atoms involved in each reaction.

## S5. Principal component analysis of learned reaction encodings

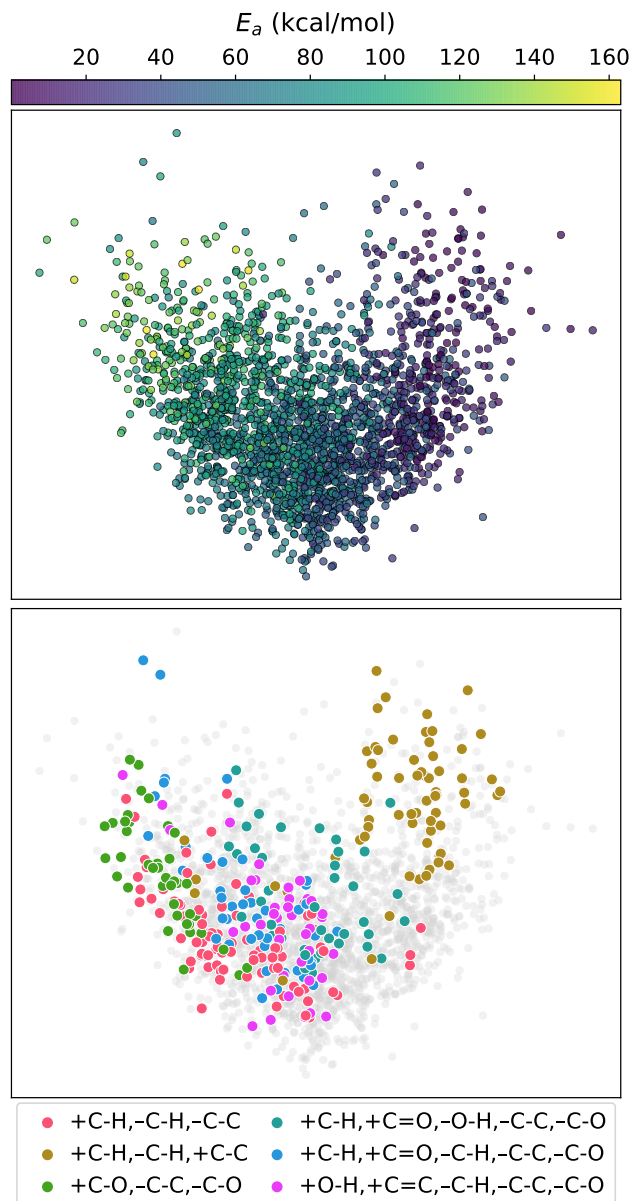


Figure S9: Principal component analysis (PCA) of the learned reaction encodings for the test set of the first fold. The first two components capture 46% of the total variance. The reactions cluster in PCA space based on their reaction type. Shown are the six most frequent reaction types (bottom). Each reaction type only includes the bond changes occurring in the reaction, e.g., +C-H,-C-H,-C-C means that a carbon-hydrogen bond is formed, a different carbon-hydrogen bond is broken, and a carbon-carbon single bond is formed in the reaction.

## S6. Side chain analysis

The side chain analysis was conducted by selecting two reactions, one with a substitutable hydrogen close to the reaction center (at a distance of 1) and one with a substitutable hydrogen far from the reaction center (at a distance of 3), and substituting the hydrogens using different functional groups (side chains). The groups were chosen as the homologous methyl, ethyl, and propyl chains; an amino group; and a hydroxy group. Figure S10 illustrates the original and substituted reactions.

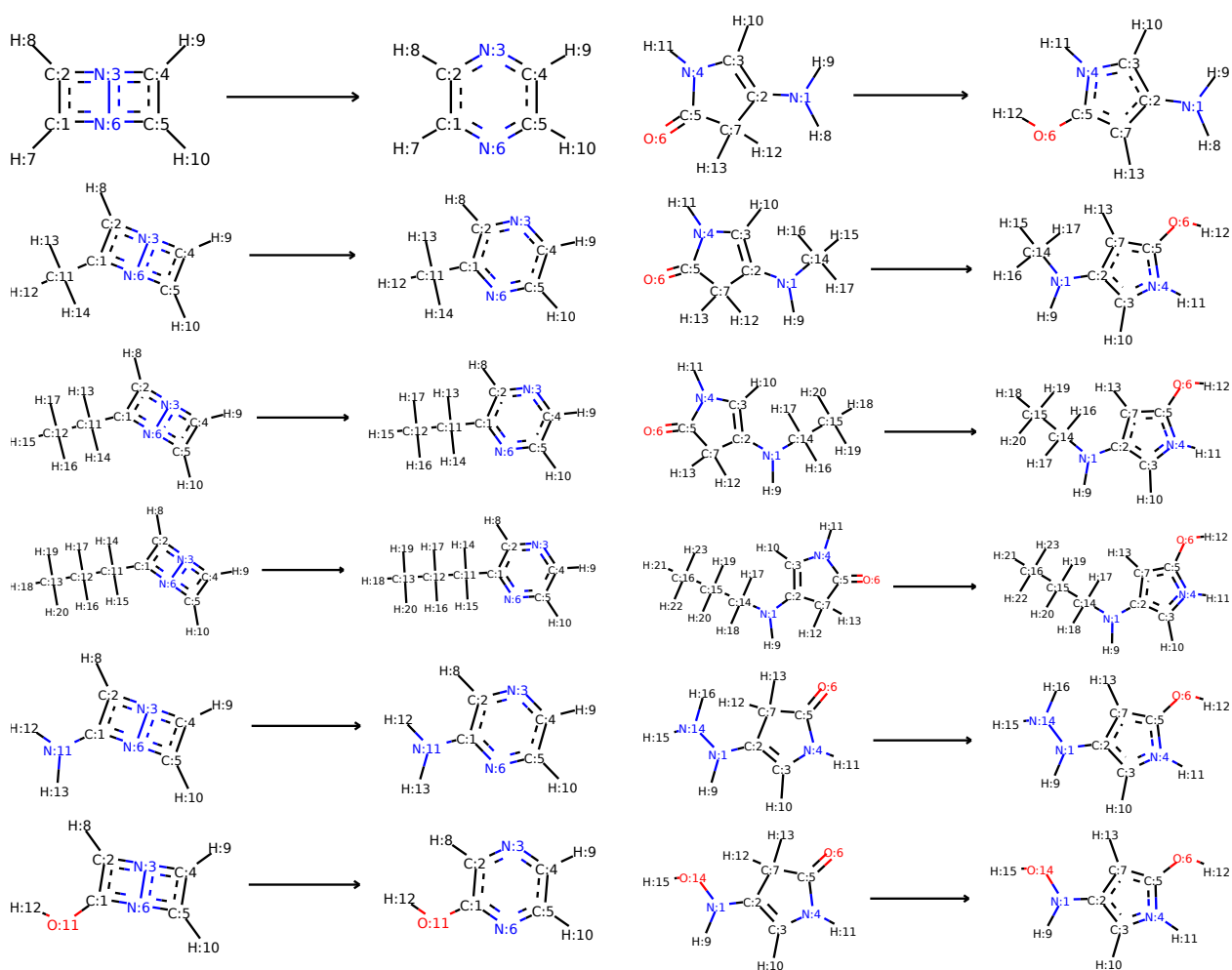


Figure S10: Substitution of side chains at a location close to the reactive center (hydrogen 7) for an example reaction (left) and at a location far from the reactive center (hydrogen 8) for a different example reaction. The topmost reactions are the original reactions and the following reactions involve different substitutions of the hydrogen atoms.

As shown in Figure S11, both the amino and hydroxy groups have a significant negative effect on the activation energy when the substitution occurs close to the reactive center. Interestingly, the more electronegative hydroxy group does not reduce the barrier as strongly as the amino group. The deep learning model agrees well with the DFT calculations, except in the case of the hydroxy group, where it predicts a barrier lower than that for the amino group. When the substitution occurs far from the reactive center, none of the side chains results in significant differences from the original barrier; and the deep learning predictions agree well with the DFT results.

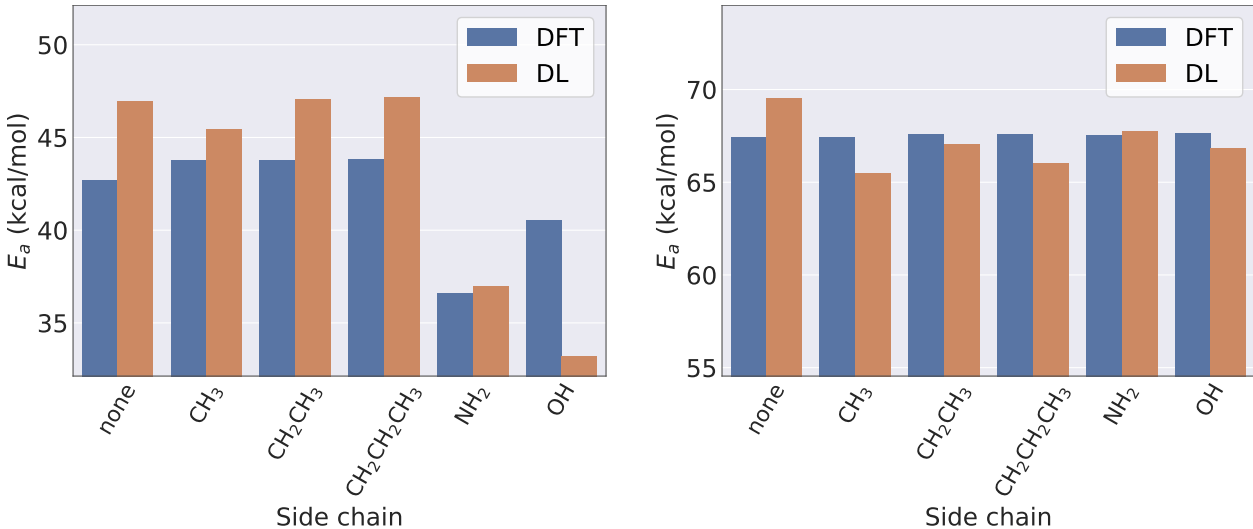


Figure S11: Change in activation energy due to the side chain substitutions illustrated in Figure S10. The left plot corresponds to the left reactions in Figure S10 and the right plot corresponds to the right reactions in Figure S10. The “true” activation energies for the substituted reactions were calculated using DFT ( $\omega$ B97X-D3/def2-TZVP) and are compared to the deep learning (DL) predictions. Note that the ordinate in both plots is scaled such that both plots have the same spacing and that its range goes from 15 kcal mol<sup>-1</sup> below the maximum barrier to 5 kcal mol<sup>-1</sup> above the maximum barrier, but does not start at zero.

The DFT results are available as a separate Supporting Information ZIP file, which contains the geometries of the reactants and transition states for each reaction in Figure S10. Each geometry file also contains the electronic energy and zero-point energy in Hartree, and a list of all harmonic vibrational frequencies in cm<sup>-1</sup> on the comment line of each XYZ-file.

## References

- (S1) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M. et al. Analyzing Learned Molecular Representations for Property Prediction. *J. Chem. Inf. Model.* **2019**, *59*, 3370–3388.
- (S2) Landrum, G. Open-Source Cheminformatics. 2006; <http://rdkit.org>.