# Spatial-Aware Deep Learning for Reaction Property Prediction

Evan Lim Hong Jun[1], Mr Chong Yihui[2], Mr Alvin Liew[2] , Dr Chieu Hai Leong[2]
[1]Raffles Institution (Junior College), 1 Raffles Institution Lane, Singapore 575954
[2]DSO National Laboratories, 12 Science Park Drive, Singapore 118225

## 1. Introduction

Predicting chemical reaction properties has a wide range of applications, including the design of new materials, the discovery of novel drugs, and the optimization of industrial processes. (REFERENCE) Automating predictions can save costs, drastically reduce the time required in the drug development cycle, and reduce use of solvents and chemicals to carry out experiments (REFERENCE). Traditionally, this has been accomplished with rule-based expert systems that manually or algorithmically encode reaction heuristics, but these systems are limited because they are not generalizable [1]. The other approach is to use quantum mechanical simulations, like density functional theory (DFT), but the computational cost is prohibitively expensive, especially in systems with large numbers of atoms [1, 2]. In recent years, the creation of large chemical datasets [3] has led to greater interest in using machine learning (ML) to predict chemical properties [4]. ML methods are generalizable because they learn universal lower dimensional embeddings of reactions and are not restricted by the chemical space of rule-based systems [5]. Furthermore, ML methods have significantly lower computational overhead than quantum methods, reducing the computation time for predicting reaction properties from hours to milliseconds. A conventional approach uses template libraries to encode reactions into their molecular fingerprints before feeding them into an ML model [6, 7, 8, 9]. However, template-based methods are limited for several inherent reasons: (1) they cannot describe novel reactions, (2) they must choose between generalisation and specificity, and (3) current algorithms neglect the global chemical environment of molecules. (REFERNCE)

Recent research has turned to template-free ML models that learn reaction representations adaptively. These models can be divided into graph-based and sequence-based models. Graph neural networks (GNNs) treat molecules as 2D graphs and aggregate atom and bond information to obtain a reaction encoding [10, 11, 12, 13]. On the other hand, sequence-based models reframe the problem as a machine translation task by using text representations of the reactants and products, usually the simplified molecular-input line-entry system (SMILES) [14, 15]. This

enables one to use sophisticated language models based on the powerful transformer architecture [16] to predict chemical reaction properties, which can achieve impressive results [17, 18]. However, sequence-methods are invariably limited because they neglect chemical domain knowledge and are blind to additional atom and bond features. Moreover, SMILES representations are not bijective mappings to molecular structures, meaning they are not permutation invariant.

By treating reactions as 2D graphs or as text, both methods neglect 3D spatial information. Some research has explored incorporating 3D coordinates as atom information in molecular property prediction [19, 10]. However, no previous work has successfully utilised 3D information in reaction property prediction. Furthermore, present research simply uses spatial information as an additional atom feature, and there are few architectures that effectively integrate spatial information into the model itself.

In this work, we propose ~~RXNformer~~ to predict chemical reaction properties. ~~RXNformer~~ first builds a condensed graph of reaction (CGR), which is a superposed graph of reactant and product molecules [20, 21]. A directed-message passing neural network (D-MPNN), which is a type of GNN, aggregates local neighbourhood information on the reaction graph for each atom. We then feed these encodings into a transformer. This leverages the representational power of the transformer, but maintains the atom and bond features from the reaction graph and is permutation invariant. Moreover, the inclusion of the transformer unlocks the possibility of embedding spatial information in the positional encodings of each atom, enabling the model to learn spatial relationships between atoms that may not necessarily be captured by the molecular graph. Finally, by analysing the attention weights in the transformer, ~~RXNformer~~ is interpretable and can help generate chemical insights. Our main contributions are as follows:

1. We present a novel architecture that synthesises GNNs and transformers to learn both low-level neighbourhood features and high-level global interdependencies for chemical reaction property prediction.
2. We propose a method to integrate spatial information of reactions into the structural representation of the model.
3. We demonstrate that integrating spatial features achieves state-of-the-art results in three reaction property regression datasets.

## 2. Related Work

**Graph Neural Networks.** GNNs have become the most ubiquitous ML technique in chemical property prediction because information flow is governed by the underlying graph structure of the molecule [11, 22, 13]. However, GNNs aggregate information in a local neighbourhood of each atom, which limits them from capturing long-range dependencies in the molecule. This is crucial since some atom pairs may be topologically distant but have important interactions, such as in intramolecular hydrogen bonding or aromatic systems. One way to address this is to increase the depth of the GNN, but this may lead to over-smoothing and poor performance [23]. Another method is to construct a virtual supernode [24] or draw virtual edges between every pair of atoms [11], but this bypasses the intrinsic structure of the molecule and creates an averaging effect. Instead, more recent work has successfully used the attention mechanism to learn global patterns in molecules [25, 26, 27].

**Transformers.** The transformer [16], originally built for natural language processing (NLP), uses multi-headed self-attention and has seen extensive application in a variety of tasks. Recently, a flood of research has explored using off-the-shelf transformer architectures to predict chemical properties from SMILES strings [28, 29, 17, 30, 18, 31, 32, 33]. These models are successful because they apply multiple layers of self-attention that allow every atom to attend to every other atom in the reaction. However, the use of SMILES strings as input neglects chemical domain knowledge and is not permutation invariant. In contrast, Maziarka et al. [34] treat molecules as a list of atoms, and augment the attention mechanism in the transformer to also consider atom features, distances and adjacency. They show that this attains a better performance than using SMILES strings.

**3D features in reactions.** Reaction properties depend on spatial distances and directions in 3D space, which cannot be captured by SMILES strings or 2D graphs. In molecular property prediction, some research has explored supplementing GNNs with 3D information such as bond distances, bond angles and torsion angles [19, 35, 36]. These studies consistently show that using 3D features improves accuracy of molecular property prediction. Recently, Ying et al. [37] showed

that by replacing the positional encoding of text with a well-designed structural encoding of a graph, it was possible for transformers to represent graphs. This is crucial because it enables one to also integrate the 3D spatial encoding of a molecule into the transformer [38]. Thus far, no work has been conducted on using 3D features for reaction property prediction. Unlike molecular property prediction, reactions involve disjoint molecules from the reactants and products. Furthermore, there may be multiple reactants and products. Thus, it is not straightforward to use 3D information of reactions. However, 3D features are especially important for reactions since it can predict effects like steric hindrance that may significantly influence reaction properties. Hence, we extend on the work by Shi et al. [38] for reaction property prediction.

## 3. RXNformer

**Graph representation.** For reaction property prediction, Heid and Green [20] achieved state-of-the-art results by feeding the condensed graph of reaction (CGR) [21] into a D-MPNN. The CGR uses the atom-mapping of a reaction to superpose the graph of reactant atoms onto the graph of product atoms. Following the recommendation of Heid and Green, we construct the atom and bond features of the CGR as the concatenation of the reactant features and the difference between the reactant features and product features. Each atom retains the bonds from both the reactant graph and the product graph. The D-MPNN architecture used is based on the chemprop library developed by Yang et al. [13]. Briefly, each edge aggregates information from neighbouring edges at the previous time step:

$$h_{vw}{}^{t+1} = \tau(h_{vw}{}^{0} + W \sum_{k \in \{N(v) \setminus w\}} h_{kv}{}^{t})$$

where $h_{vw}{}^{t}$ are the edge features for the edge between nodes v and w at message-passing timestep t, $\tau$ is a non-linear activation function, $N(v)\backslash w$ is the set of neighbours of v excluding w, and $W \in R^{dxd}$, d being the size of the hidden dimension. Further details of the D-MPNN can be found in appendix A.

<Fig 2: Schema for constructing condensed graph of reaction>

**Spatial representation.** We use RDKit to obtain 3D coordinates of every atom. However, we cannot directly use these coordinates because the spatial features should be invariant to rotation

and translation of the molecule. Thus, we construct a matrix of the euclidean distance between every pair of atoms in the same molecule. For atoms that are in different molecules, we set the distance to be 0. This represents the equivalent of the atoms being infinitely far apart.

$$d_{ij} = \|r_i - r_j\|_{L2}$$

To find the distance matrix for the CGR, we take the difference in the distance in the product matrix and the distance in the reactant matrix. Intuitively, <>

$\Delta_{diff} = d_{prod} - d_{reac}$

To learn distributions in the 3D distances and obtain spatial encodings, we use Gaussian basis kernel functions [39]. Since the relative position of atoms are highly dependent on the size of their atomic radii, we first perform an affine transformation on the distances based on the atom types $a_i$ and $a_j$ of atoms $i$ and $j$, $\gamma_{(a_i,a_j)}\Delta_{ij} + \beta_{(a_i,a_j)}$ where $\gamma$ and $\beta$ are learnable scalars for each pair of atom types. We then pass this through a Gaussian density function to obtain the 3D spatial encoding for each kernel:

$$\psi_{ij}{}^{k} = \frac{1}{\sigma^k\sqrt{2\pi}} exp(-\frac{(\gamma_{(a_i,a_j)}\Delta_{ij} + \beta_{(a_i,a_j)} - \mu^k)^2}{2(\sigma^k)^2})$$

where k is the kernel number, $\sigma^k$ is the learnable kernel centre of the k-th kernel, and $\mu^k$ is the learnable scaling factor of the k-th kernel. Further details can be found in appendix A.

**Transformer readout.** The D-MPNN learns local neighbourhood features in the CGR. We extract the final atom representations after 3 message-passing steps. In NLP, the BERT architecture uses a special [CLS] token to capture the representation of the entire sentence. Similarly, we append a virtual [CLS] node to represent the entire reaction. It is initially set to the mean of all atoms. We integrate the spatial encodings of the reaction into both the absolute positional encoding [16] and relative positional encoding [40] of the transformer. The absolute spatial encoding is the sum of 3D spatial encodings to all other atoms, which we add to the input X:

$$X_i' = X_i + \varphi_i{}^{sum}$$

The relative spatial encoding is a linear projection of the 3D spatial encodings, which is used as an attention bias in the self-attention step:

$$Attention(Q_i^{l,h}, K_j{}^{l,h}, V_j{}^{l,h}) = softmax(\frac{Q_i{}^{l,h}(K_j{}^{l,h})^T}{\sqrt{d}} + \phi_{ij}{}^{h,spatial\ encoding})V_j{}^{l,h}$$

Further details can be found in appendix A.

## 4. Experiments

We conduct experiments on our model with four reaction property regression datasets. These datasets comprise three different tasks: predicting activation energies [2, 41], reaction enthalpies [42], and reaction rate constants [43]. Details on the datasets can be found in appendix C. We compare our model to the baselines and state-of-the-art established by Heid and Green [20]. This is shown in table 1. Each experiment was conducted with five-fold cross validation to obtain the average score and standard deviations. Further implementation and experimental details can be found in appendix B.

| | *Ea* ωB97X-D3 (kcal/mol) | *Ea* E2/SN2 (kcal/mol) | ΔH Rad-6-RE\| (eV) | log(k) rate constants (unitless) |
|---|---|---|---|---|
| **Template-based methods** | | | | |
| Morgan Diff default | 13.08 ± 0.98 | 4.75 ± 0.24 | 1.11 ± 0.03 | 0.88 ± 0.09 |
| Morgan Diff opt | 11.39 ± 0.39 | 4.53 ± 0.23 | 0.72 ± 0.02 | 0.75 ± 0.08 |
| ISIDA default | 9.21 ± 0.55 | 3.14 ± 0.09 | 0.48 ± 0.02 | 0.73 ± 0.06 |
| ISIDA opt | 7.55 ± 0.48 | 3.00 ± 0.10 | 0.43 ± 0.03 | 0.59 ± 0.06 |
| **Sequence methods** | | | | |
| Bert default | 14.73 ± 0.52 | 3.54 ± 0.12 | 0.76 ± 0.02 | 0.99 ± 0.03 |
| Bert opt | 10.94 ± 0.29 | 3.37 ± 0.10 | 0.65 ± 0.01 | 0.88 ± 0.02 |
| **Graph methods** | | | | |
| Grambow default | 6.35 ± 0.26 | 2.76 ± 0.08 | 0.40 ± 0.01 | 1.00 ± 0.14 |
| Grambow opt | 5.26 ± 0.15 | 2.86 ± 0.07 | – [a] | 0.76 ± 0.26 |
| CGR default | 4.84 ± 0.29 | 2.64 ± 0.10 | 0.16 ± 0.01 | 0.66 ± 0.29 |
| CGR opt | 4.25 ± 0.19 | 2.65 ± 0.09 | **0.13 ± 0.01** | 0.66 ± 0.24 |
| RXNformer (ours) | **2.68 ± 0.61** | **2.53 ± 0.11** | 0.14 ± 0.02 | **0.33 ± 0.08** |

Table 1: Comparison of mean absolute error (MAE) score on each dataset. Lower score is better. The best performance per dataset is highlighted in bold. The root mean squared error (RMSE) scores and model sizes can be found in appendix D. [a]As highlighted by Heid and Green, grambow opt simply memorises enthalpies of molecules that repeat in the training and test set, thus we exclude it from this comparison.

**Discussion.** RXNformer outperforms previous methods in predicting activation energies and reaction rates. However, RXNformer does not observe significant improvement in predicting reaction enthalpies. This could be because reaction enthalpies are mostly dependent on bond enthalpies and not on the 3D environment, making global interactions and spatial features less significant. Thus, a simple GNN which can learn bond enthalpies would be able to predict reaction enthalpies. This may suggest that spatial information is less important for thermodynamic properties like enthalpy but more important for kinetic properties like activation energy. We conduct further ablation studies to evaluate the contributions of each component of our model. This can be found in appendix D.

The transformer component of RXNformer also allows us to obtain attention weights, revealing which atom neighbourhoods influence the final prediction the most. This makes RXNformer more interpretable than GNNs, which acts as a sanity check that increases trust in the model prediction by allowing chemists to verify if the prediction aligns with chemical intuition. Furthermore, it may reveal chemical insights about the reaction that help chemists identify the substructures or functional groups which contribute the most to the property of interest, allowing chemists to make modifications for optimization.

## 5. Conclusions

In this work, we presented the ~~RXNformer~~ as an effective architecture for reaction property prediction. RXNformer can represent local and global interactions and incorporates 3D spatial information into the structure of the model, achieving state-of-the-art results in predicting activation energies and reaction rate constants. Moreover, RXNformer is interpretable, giving it more real-world applicability than previous 'black-box' models which can only provide the user with a numerical prediction.

However, RXNformer also has limitations. Compared to previous architectures, RXNformer requires more data pre-processing steps to generate atom-mapping and 3D information. Furthermore, if the atom-mapping or 3D information is noisy or incomplete, it may hinder model performance. Nonetheless, our work highlights that spatial features are critical for reaction property prediction, especially for kinetic properties, and that future work should explore building models that can make greater use of spatial information like bond angles.

## Acknowledgements

This paper would not be possible without the invaluable guidance from my expert-mentors, Mr Chong Yihui, Mr Alvin Liew and Dr Chieu Hai Leong from DSO National Laboratories.

## References

[1] A. Puliyanda, K. Srinivasan, K. Sivaramakrishnan and V. Prasad, “A review of automated and data-driven approaches for pathway determination and reaction monitoring in complex chemical systems,” *Digital Chemical Engineering,* vol. 2, 2022.

[2] C. A. Grambow, L. Pattanaik and W. H. Green, “Reactants, products, and transition states of elementary chemical reactions based on quantum chemistry,” *Sci Data,* vol. 7, p. 137, 2020.

[3] Z. Wu, B. Ramsundar, E. N. Feinberg, J. Gomes, C. Geniesse, A. S. Pappu, K. Leswing and V. Pande, “MoleculeNet: A benchmark for molecular machine learning,” *Chemical Science,* vol. 9, no. 2, 2018.

[4] C. W. Coley, W. H. Green and K. F. Jensen, “Machine Learning in Computer-Aided Synthesis Planning,” *Accounts of Chemical Research,* vol. 51, no. 5, 2018.

[5] T. F. Cova and A. A. Pais, “Deep Learning for Deep Chemistry: Optimizing the Prediction of Chemical Patterns,” *Frontiers in Chemistry,* vol. 7, 2019.

[6] C. W. Coley, L. Rogers, W. H. Green and K. F. Jensen, “Computer-Assisted Retrosynthesis Based on Molecular Similarity,” *ACS Central Science,* vol. 3, no. 12, 2017.

[7] M. H. Segler and M. P. Waller, “Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction,” *Chemistry - A European Journal,* vol. 23, no. 25, 2017.

[8] A. S. Christensen, L. A. Bratholm, F. A. Faber and O. Anatole Von Lilienfeld, “FCHL revisited: Faster and more accurate quantum machine learning,” *Journal of Chemical Physics,* vol. 152, no. 4, 2020.

[9] S. Choi, Y. Kim, J. W. Kim, Z. Kim and W. Y. Kim, “Feasibility of Activation Energy Prediction of Gas-Phase Reactions by Machine Learning,” *Chemistry - A European Journal,* vol. 24, no. 47, 2018.

[10] K. T. Schütt, H. E. Sauceda, P. J. Kindermans, A. Tkatchenko and K. R. Müller, “SchNet - A deep learning architecture for molecules and materials,” *Journal of Chemical Physics,* vol. 148, no. 24, 2018.

[11] S. Kearnes, K. McCloskey, M. Berndl, V. Pande and P. Riley, “Molecular graph convolutions: moving beyond fingerprints,” *Journal of Computer-Aided Molecular Design,* vol. 30, no. 8, 2016.

[12] D. Duvenaud, D. Maclaurin, J. Aguilera-Iparraguirre, R. Gómez-Bombarelli, T. Hirzel, A. Aspuru-Guzik and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," *Advances in Neural Information Processing Systems,* Vols. 2015-January, 2015.

[13] K. Yang, K. Swanson, W. Jin, C. Coley, P. Eiden, H. Gao, A. Guzman-Perez, T. Hopper, B. Kelley, M. Mathea, A. Palmer, V. Settels, T. Jaakkola, K. Jensen and R. Barzilay, "Analyzing Learned Molecular Representations for Property Prediction," *Journal of Chemical Information and Modeling,* vol. 59, no. 8, 2019.

[14] D. Weininger, "SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules," *Journal of Chemical Information and Computer Sciences,* vol. 28, no. 1, 1988.

[15] J. Nam and J. Kim, "Linking the Neural Machine Translation and the Prediction of Organic Chemistry Reactions," 12 2016.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems,* Vols. 2017-December, 2017.

[17] P. Schwaller, D. Probst, A. C. Vaucher, V. H. Nair, D. Kreutter, T. Laino and J. L. Reymond, "Mapping the space of chemical reactions using attention-based neural networks," *Nature Machine Intelligence,* vol. 3, no. 2, 2021.

[18] R. Irwin, S. Dimitriadis, J. He and E. J. Bjerrum, "Chemformer: A pre-trained transformer for computational chemistry," *Machine Learning: Science and Technology,* vol. 3, no. 1, 2022.

[19] J. Gasteiger, J. Groß and S. Günnemann, "Directional Message Passing for Molecular Graphs," 3 2020.

[20] E. Heid and W. H. Green, "Machine Learning of Reaction Properties via Learned Representations of the Condensed Graph of Reaction," *Journal of Chemical Information and Modeling,* 2021.

[21] F. Hoonakker, N. Lachiche and A. Varnek, "Condensed Graph of Reaction: Considering a Chemical Reaction as One Single Pseudo Molecule," *International Journal on Artificial Intelligence Tools,* vol. 20, no. 2, 2011.

[22] C. W. Coley, R. Barzilay, W. H. Green, T. S. Jaakkola and K. F. Jensen, "Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction," *Journal of Chemical Information and Modeling,* vol. 57, no. 8, 2017.

[23] C. Cai and Y. Wang, "A Note on Over-Smoothing for Graph Neural Networks," 6 2020.

[24] J. Li, D. Cai and X. He, "Learning Graph-Level Representation for Drug Discovery," 9 2017.

[25] Z. Xiong, D. Wang, X. Liu, F. Zhong, X. Wan, X. Li, Z. Li, X. Luo, K. Chen, H. Jiang and M. Zheng, "Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism," *Journal of Medicinal Chemistry,* vol. 63, no. 16, 2020.

[26] C. Shang, Q. Liu, K.-S. Chen, J. Sun, J. Lu, J. Yi and J. Bi, "Edge Attention-based Multi-Relational Graph Convolutional Networks," 2 2018.

[27] M. Tavakoli, A. Shmakov, F. Ceccarelli and P. Baldi, "Rxn Hypergraph: a Hypergraph Attention Model for Chemical Reaction Representation," 1 2022.

[28] P. Karpov, G. Godin and I. V. Tetko, "A Transformer Model for Retrosynthesis," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics),* vol. 11731 LNCS, 2019.

[29] P. Schwaller, A. C. Vaucher, T. Laino and J. L. Reymond, "Prediction of chemical reaction yields using deep learning," *Machine Learning: Science and Technology,* vol. 2, no. 1, 2021.

[30] P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. A. Hunter, C. Bekas and A. A. Lee, "Molecular Transformer: A Model for Uncertainty-Calibrated Chemical Reaction Prediction," *ACS Central Science,* vol. 5, no. 9, 2019.

[31] X. Wang, Y. Li, J. Qiu, G. Chen, H. Liu, B. Liao, C. Y. Hsieh and X. Yao, "RetroPrime: A Diverse, plausible and Transformer-based method for Single-Step retrosynthesis predictions," *Chemical Engineering Journal,* vol. 420, 2021.

[32] J. Ross, B. Belgodere, V. Chenthamarakshan, I. Padhi, Y. Mroueh and P. Das, "Large-Scale Chemical Language Representations Capture Molecular Structure and Properties," 6 2021.

[33] S. Honda, S. Shi and H. R. Ueda, "SMILES Transformer: Pre-trained Molecular Fingerprint for Low Data Drug Discovery," 11 2019.

[34] Ł. Maziarka and J. Tabor, "Molecule-Augmented Attention Transformer," *NeurIPS Workshop on Machine Learning and the Physical Sciences,* no. NeurIPS, 2019.

[35] H. Cho and I. S. Choi, "Three-Dimensionally Embedded Graph Convolutional Network (3DGCN) for Molecule Interpretation," *ChemMedChem,* 11 2019.

[36] R. Li, S. Wang, F. Zhu and J. Huang, "Adaptive graph convolutional neural networks," *32nd AAAI Conference on Artificial Intelligence, AAAI 2018,* 2018.

[37] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen and T.-Y. Liu, "Do Transformers Really Perform Bad for Graph Representation?," *NeurIPS 2021,* 6 2021.

[38] Y. Shi, S. Zheng, G. Ke, Y. Shen, J. You, J. He, S. Luo, C. Liu, D. He and T.-Y. Liu, "Benchmarking Graphormer on Large-Scale Molecular Modeling Datasets," 3 2022.

[39] M. Shuaibi, A. Kolluru, A. Das, A. Grover, A. Sriram, Z. Ulissi and C. L. Zitnick, "Rotation Invariant Graph Neural Networks using Spin Convolutions," 6 2021.

[40] P. Shaw, J. Uszkoreit and A. Vaswani, "Self-attention with relative position representations," *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference,* vol. 2, 2018.

[41] G. F. von Rudorff, S. N. Heinen, M. Bragato and O. Anatole von Lilienfeld, "Thousands of reactants and transition states for competing E2 and SN2 reactions," *Machine Learning: Science and Technology,* vol. 1, no. 4, 2020.

[42] S. Stocker, G. Csányi, K. Reuter and J. T. Margraf, "Machine learning in chemical reaction space," *Nature Communications,* vol. 11, no. 1, 2020.

[43] P. L. Bhoorasingh, B. L. Slakman, F. Seyedzadeh Khanshan, J. Y. Cain and R. H. West, "Automated Transition State Theory Calculations for High-Throughput Kinetics," *Journal of Physical Chemistry A,* vol. 121, no. 37, 2017.

## Appendix A – Architecture details

**Building the CGR.** Let the molecular graph be an undirected graph $G$ with atoms $V$ and bonds $E$. Each atom $v \in V$ has a feature vector $x_v \in R^a$ and each bond $(u, v) \in E$ from atom $u$ to $v$ has a feature vector $e_{vw} \in R^b$. To form the CGR, we superpose the atoms from the products onto the

corresponding atoms from the reactants based on the atom-mapping. We concatenate the features from the reactant graph with the difference in features in the reactant and product graphs to obtain the new features:

$$x_v = cat(x_v^{reac}, x_v^{diff})$$

$$e_{vw} = cat(e_{vw}^{reac}, e_{vw}^{diff})$$

The list of bonds now comprises bonds from both the reactant graph $G^{reac}$ and product graph $G^{prod}$, $E = E^{reac} \cup E^{prod}$. All four datasets used are balanced, so the list of atoms is simply $V = V^{reac} = V^{prod}$.

**D-MPNN.** The D-MPNN consists of a message passing phase and a readout phase. The message passing phase consists of $T$ time steps, where each edge is updated based on the information from neighbouring edges. We initialize the hidden state in each edge at time step 0 by concatenating the features of the first atom $x_v$ to the bond features $e_{vw}$ and passing it through a single layer:

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

where $W_i \in R^{d \times d_i}$, and $d$ being the hidden size (300), $d_i$ being the size of $cat(x_v, e_{vw})$, and $\tau()$ being a nonlinear activation function (ReLU). The message $m_{vw}^{t+1}$ at time step $t+1$ in edge $vw$ is the sum of hidden states from neighbouring edges directed towards $v$ at the previous time step $t$:

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

where $N(v)\backslash w$ is the set of neighbours of $v$ excluding $w$. To obtain the new hidden state values:

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

where $W_m \in R^{d \times d}$. We repeat this for 3 times steps. Finally, we transform the hidden states back into atom representations:

$$h_v = \tau(W_0(cat(x_v \sum_{w \in N(v)} h_{wv}^T))$$

where $W_0 \in R^{d \times d_0}$, where $d_0$ is the size of $cat(x_v, h_{vw})$. At the end, we obtain representations $h_v$ of each atom in the CGR of shape $n \times d$, where $n$ is the number of atoms.

**Getting the spatial features.** We use RDKit to generate the 3D coordinates of each atom in each molecule in the reaction. Within each molecule, we calculate the Euclidean distance between every pair of atoms $d_{ij}$. For atoms in different molecules, we set the distance to be 0. We repeat this for the reactants and the products separately.

$$d_{ij} = \{\|r_i - r_j\|_{L2} \quad , \qquad \&if\ same\ molecule\ 0, \qquad otherwise$$

We first pad the distance matrixes to the maximum number of atoms and append the virtual [CLS] atom. To find the distance matrix for the CGR, we take the difference in the distances in the product matrix and the distances in the reactant matrix.

$$\Delta_{diff} = d_{prod} \ - \ d_{reac}$$

We perform an affine transformation on the distances based on the atom types $a_i$ and $a_j$ of atoms $i$ and $j$, $\gamma_{(a_i,a_j)}\Delta_{ij} \ + \ \beta_{(a_i,a_j)}$ where $\gamma$ and $\beta$ are learnable scalars for each pair of atom types. We then pass this distance matrix through a set of gaussian density functions with $k$ kernels:

$$\psi_{ij}{}^{k} = \frac{1}{\sigma^k\sqrt{2\pi}} exp(-\frac{(\gamma_{(a_i,a_j)}\Delta_{ij} \ + \ \beta_{(a_i,a_j)} - \mu^k)^2}{2(\sigma^k)^2})$$

where k is the kernel number, $\sigma^k$ is the learnable kernel centre of the k-th kernel, and $\mu^k$ is the learnable scaling factor of the k-th kernel. For the virtual [CLS] atom, and all other padded atoms, we set $\psi_{ij}^k$ to be 0. The shape of $\psi$ is $N \times N \times K$. The absolute spatial encoding of atom $i$ is calculated as the sum of the spatial encodings to all other atoms $j$ and passed through a linear layer:

$$\varphi_i^{sum} = W_k^0 \sum_{j \in V} \quad \psi_{ij}^k$$

where $W_k^0 \in R^{d \times K}$. The shape of $\varphi^{sum}$ is $N \times d$, where N is the max number of atoms (after padding). The relative spatial encoding between atoms $i$ and $j$ is a linear projection of the 3D spatial encodings:

$$\phi_{ij}^{spatial\ encoding} = \ GELU\left(\psi_{ij}W_k^1\right)W_k^2$$

where $W_k^1 \in R^{K \times K}$ and $W_k^2 \in R^{K \times H}$, and $H$ is the number of heads in the transformer. The final shape of $\phi$ is $N \times N \times H$.

**Transformer readout.** We take the final atom representations from the D-MPNN $h$ and pad it to the maximum number of atoms $N$. We also append a virtual [CLS] atom to represent the entire reaction. We first briefly introduce the transformer architecture. The transformer consists of N attention layers, where each layer is composed of a multi-head self-attention block, followed by a feed-forward block, with both block having residual connections and layer normalization. The multi-headed self-attention comprises H heads. Let the input to the $l$-th layer be $X^l$. Head $i$ computes the vectors $Q_i = XW_i^Q$, $K_i = XW_i^K$, $V_i = XW_i^V$. The attention operation is:

$$Attention(Q, K, V) = softmax()$$

Our implementation of the transformer encoder is inspired by The Annotated Transformer[1].

The absolute spatial encoding is the sum of 3D spatial encodings to all other atoms, which we add to the input X:

$$X' = X + {\varphi_{ij}}^{sum}$$

The relative spatial encoding is a linear projection of the 3D spatial encodings, which is used as an attention bias in the self-attention step:

$$Attention({Q_i}^{l,h}, {K_j}^{l,h}, {V_j}^{l,h}) = softmax(\frac{{Q_i}^{l,h}({K_j}^{l,h})^T}{\sqrt{d}} + {\phi_{ij}}^{h,spatial\ encoding}){V_j}^{l,h}$$

padding

## Appendix B – Experimental details

**Training details.** All models are trained on a single NVIDIA <> GPU for 100 epochs. This takes approximately

The transformer has 2 layers and each layer has 3 self-attention heads. We do not use dropout. The batch size is 50

<Fig: Atom and bond features used>

[1] http://nlp.seas.harvard.edu/annotated-transformer/

Training Details

**Obtaining 3D Information.** Spatial data was obtained with
Conformers definition, lowest energy and UFF rdkit
skipped molecules
pre-processing steps, for highly strained molecules may not work.

## Appendix C – Dataset details

datasets

## Appendix D – Additional Results

rmse scores, model size

ablation studies

Attention and interpretability

**Ablation studies.** We conduct ablation studies on the ωB97X-D3 dataset to evaluate the contributions of each component of our model. In table 2, we compare not using atom-mapped information, drawing virtual edges between mapped-atoms, or building a CGR. We find that the CGR has a stronger inductive bias than drawing virtual edges. In table 3, we compare the effects of adding spatial features and the transformer. We find that both improve model performance.

- Example reaction that can demonstrate importance of spatial features and transformer/global interactions for my diagram
- What other analysis on which reactions were guessed right / guessed wrong should I do?

Get attention weights on e2sn2

Reviews