2023 IEEE International Conference on Big Data (BigData) | 979-8-3503-2445-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/BigData59044.2023.10386300

Mining Large-Scale Knowledge Graphs for Chemical Reaction Fingerprints

Blake B. Gaines
Department of Computer Science
University of Connecticut
Storrs, CT, USA
blake.gaines@uconn.edu

Minghu Song
Department of Biomedical Engineering
University of Connecticut
Storrs, CT, USA
minghu.song@uconn.edu

Jinbo Bi*
Department of Computer Science
University of Connecticut
Storrs, CT, USA
jinbo.bi@uconn.edu

***Abstract*—Knowledge graphs have become a popular method for representing large, relational data. Similar to citation networks and social networks, relationships in chemical reaction data can also be uniquely captured using a knowledge graph. However, relatively few studies exist concerning the application of knowledge graph mining techniques for numerical representation of chemical reactions. In this study, we develop a pipeline for transforming large-scale relational databases of chemical reactions into heterogeneous graphs, in which reactions and their reactants and products are all characterized as nodes with connecting edges. We create nodes for reaction templates, each of which links to multiple reactions to enhance the connectivity of the graph, and then employ graph representation learning methods (Node2Vec and RotatE) to generate an embedding (or fingerprint) for each reaction node. To evaluate the efficacy of this method, we construct classifiers to label the mechanisms of reactions based on these fingerprints. Experimental results show that our graph learning approach outperforms the state-of-the-art reaction fingerprints, specifically when class labels are not available during the representation learning process. When the representations can be fine-tuned for the subsequent classification task, our approach achieves comparable accuracy to a recent Transformer-based algorithm, but with a significantly lower computational cost.**

***Index Terms*—Knowledge Graphs, Graph Mining, Chemical Reaction Fingerprints, Representation Learning**

I. Introduction

Employing quantitative methods to understand and categorize molecules and chemical reactions has become increasingly important. Large databases of reactions have become common resources for chemists in the field, but their size and lack of organization make interacting with them difficult [21]. Reaction classifications and similarity rankings facilitate this process, allowing chemists to infer the properties of certain reactions based on a general description of their mechanisms [27] [16]. For instance, information about optimal reaction conditions and yields can be found for a query reaction based on related reactions, helping chemists predict the quality of certain synthesis routes [20] [3]. Creating the tools to leverage these datasets has great potential to reduce the time and cost of chemical research and drug discovery.

Molecules are commonly represented computationally by hydrogen-depleted molecular graphs or strings using the Simplified Molecular-Input Line-Entry System (SMILES) [29]. Many methods exist to encode either the graph or string representations of molecules into numerical vectors [31]. This is a more convenient format for subsequent analysis, such as property optimization [11] or molecule classification [23]. These embedding methods have also been extended to learn chemical reactions involving multiple molecules to generate "fingerprints". Traditional methods for creating fingerprints are deterministic. Some of these methods create sparse feature vectors by directly searching for chemical substructures or residues that are responsible for reactions, requiring a separate feature for each one [5]. Others try to encode all the structural features of each molecule, such as in Morgan fingerprints, which are created by performing message passing on the molecular graphs of reactive species and hashing the result [15]. These methods benefit from being highly transparent. However, these methods often fail to capture important properties determined by molecular structure, and require very high dimensional vectors to reach the appropriate granularity for describing molecular properties [18].

Machine learning (ML) has allowed for the creation of more informative fingerprints that can be tailored to the task at hand. For example, a recent method in [26] fine-tuned Transformer-based fingerprints (RXNFP) were shown to greatly outperform the best performing traditional fingerprint introduced in [23] when classifying chemical reactions. Data-driven fingerprinting without hand-crafted design frees the model to create the best possible representations for differentiating reactions. However, this comes at the cost of model interpretability and controllability. The present work aims to introduce fingerprint generation methods that retain the efficacy of ML-generated fingerprints while preserving interpretability by integrating chemical knowledge into the encoding process.

II. Methods

We propose a new approach as shown in Figure 1 to automatically learn and generate reaction fingerprints that are useful for categorization or classification of chemical reactions. We start with a relational database of labeled chemical reactions, such as Pistachio [22] or USPTO 1k TPL [26]. Using a previously published and validated method, we generate reaction templates, patterns that generalize individual

*Corresponding Author

979-8-3503-2445-7/23/$31.00 ©2023 IEEE

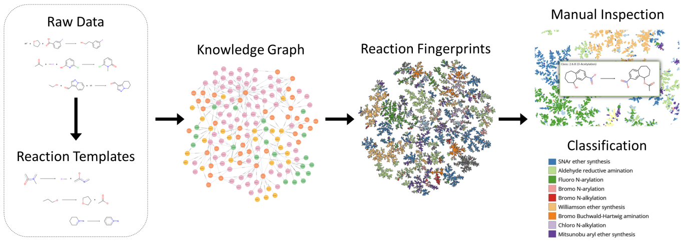

Fig. 1. Overview of the proposed approach.

reaction mechanisms by matching the fragments (i.e., chemical substructures) of each compound that are changed by the reaction. The relational reaction database can then be transformed into a knowledge graph (KG) where different types of nodes represent reactions, compounds, fragments, and templates. We can generate reaction fingerprints by employing graph representation learning techniques to embed the corresponding reaction nodes in a continuous vector space that encodes the structure of the graph. The resultant fingerprints are evaluated by visual inspection and reaction classification performance.

A. Datasets

We use the proprietary Pistachio dataset [22] and publicly available USPTO 1k TPL dataset [26]. Each dataset is split into training, validation, and test sets (80%, 10%, and 10% respectively, matching the recent work proposing RXNFP). USPTO 1k TPL has 445k reactions associated with 1000 labels for reaction types. Pistachio (version 2021Q1) contains 3,348,453 reactions, and hierarchical labels that follow the RXNO ontology [24]. These labels are algorithmically assigned by NameRXN using a large rule-base of known reaction mechanisms [25]. Each label contains a general category from one of 12 superclasses, with one representing all reactions that could not be classified by the software. Each superclass is further divided into intermediate classes, which themselves may be divided into several fine classes. In total, there are 1,385 fine classes. Duplicate reactions are removed, and RDKit [1] and Open Babel [17] are used to verify and canonicalize the reactions and compounds. We also move reagents into the part of the SMILE that is designated for reactants, since reactants and reagents are not always explicitly separated in reaction datasets.

B. Extracting Reaction Templates

We generalize reactions by finding "reactive fragments", the groups of atoms in each compound at which bonds are broken or formed during the reaction, as well as surrounding atoms that play a role. If reactive fragments are the same for two reactions, and the resultant fragments of the products are the same, then the underlying chemical processes are likely similar. Together, the reactive fragments of the reactants and products of a reaction form a sort of "template", which can be used to group similar reactions. The fragments and templates can also be expressed using strings according to the SMILES arbitrary target specification (SMARTS). We employ the algorithm described in [2] to generate template SMARTS from reaction SMILES. An example template of a reaction is illustrated in Fig. 2.

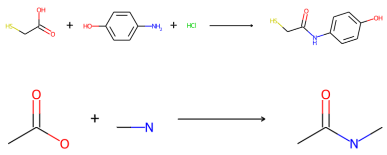

Fig. 2. An example reaction (top) and its template (bottom). This reaction involves the coupling of thioglycolic acid and 4-aminophenol, the first and second molecules from the left. The hydrochloric acid, third from the left, is a reagent and is not consumed by the reaction. The left side of the template identifies the relevant functional group from each reactant: the carboxyl group from the thioglycolic acid and the amine group from the 4-aminophenol. The right side of the template shows how these functional groups change during the reaction, producing a single amide group. Other reactions where the same functional groups combine in this way will have the same template.

Although the templates do not perfectly correspond to actual reaction mechanisms, they still represent meaningful information about a reaction, which is used in the construction of the KG. For conventional machine learning methods, there is no straightforward way to incorporate the fact that two reactions share a template. For a large reaction database, a large number of unique templates may exist, and it can be challenging to represent these templates with numerical vectors that reflect their relations. We thus employ a graph structure to characterize these relationships among the different reactions.

C. Constructing the Knowledge Graph

Similar to methods used in natural language processing for representing highly structured data [12], KGs have been used previously by chemists to perform tasks like synthesis route prediction [13]. A KG can reflect chemical knowledge.

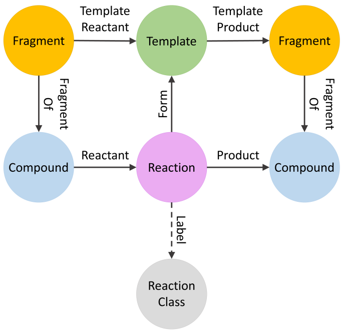

Fig. 3. The schema for our chemical knowledge graph.

The schema of our graph is shown in Fig. 3. At the center is the reaction node, which can link to compound nodes and templates. The templates are also linked to the fragments that define them, which themselves are linked to the matching compounds. We do not create separate node types for reactants and products, because the same compound can be a reactant to one reaction but a product to another. Instead, separate reactant and product edge-types are created between reaction nodes and compound nodes. Multiple fragments may be linked to a single molecule because different parts of the same molecule may be used in different reactions. A single fragment may be designated as a reactive subgraph of multiple compounds. In the resulting KG, reactions that do not directly share any information can still be identified as similar based on the number and types of paths between them.

Reaction fingerprints can be created using labels from the downstream task, which we refer to as supervised fingerprints. To generate fingerprints that differentiate reaction categories, a new type of nodes — "reaction class" nodes — can also be included to increase connectivity between reactions with the same class label. Reactions with known labels are linked to the corresponding reaction class nodes, which will force the embedding algorithm to consider the labels when creating the fingerprints, directly influencing their spatial organization. If fingerprints are not generated using any task-specific labels (i.e., the KG has no reaction class nodes), we refer to them as unsupervised fingerprints. We use Neo4j (https://neo4j.com/) to create and query a KG before exporting them into various formats for the different learning algorithms.

D. Mining the Knowledge Graph

We now use the structure of the KG to construct embeddings for each node in the graph. A variety of methods have been developed for representation learning of KGs, and any of them can be used in our pipeline. Here we employ two widely used methods: RotatE [10] and Node2Vec [8]. RotatE, implemented in the open source package OpenKE [10], represents entities as complex vectors and relationships as rotations in the complex plane. Triplets represent possible relationships in the form of a head node h connected to a tail node t by a relationship r. RotatE creates corresponding embeddings $e_h, e_r, e_t \in \mathbb{C}^n$, and optimize them to maximize the score function $-\|e_h \circ e_r - e_t\|^2$ if the triplets exist in the KG and minimize it if they do not. An advantage of this method is that we can classify unlabeled reactions in a knowledge graph with reaction label nodes without training another classifier, because we can identify the 'reaction class' node whose embedding maximizes the scoring function with the embeddings of a given reaction node and the "Label" relationship (see Fig. 3).

Another popular tool for generating node embeddings is Node2Vec, which functions like Word2Vec [14] when using random walks on the KG as "sentences" and the nodes as "words". Our experiments show that using a continuous-bag-of-words (CBOW) model to learn node embeddings for our KG was not only effective, but also computationally efficient relative to deep fingerprints. Compared with an existing transformer-based model, RXNFP, our KG mining approach is substantially more efficient. For example, fine-tuning the RXNFP fingerprints on the Pistachio dataset took 107 hours on a Tesla V100 graphics card with 32GB of RAM. In contrast, using the same group of CPUs without GPUs, Node2Vec was trained on Pistachio in 19 hours (including the time needed to generate random walks).

E. Prediction

The reaction fingerprints constructed from our reaction KG can be used in subsequent analysis. To evaluate the quality of the fingerprints, we test whether they can help classify the different reactions into correct categories. We compare our KG based fingerprints with two other reaction fingerprints: traditional Morgan fingerprints and RXNFP fingerprints, the latter of which are generated by the encoder of a Bidirectional Encoder Representations (BERT) model [4]. In the pretraining phase, the BERT model is trained on a masked language modeling task, where it must predict missing tokens in a partially observed SMILES string describing the reaction. We refer to these as "untuned" RXNFP fingerprints, since the BERT decoder can subsequently be replaced by a classification head in order to fine-tune the fingerprints for predicting reaction labels. The fingerprints generated in this supervised way are singly referred to as "RXNFP" in our result table. The supervised fingerprinting model is both pretrained and fine-tuned on the training set, and then the resultant model is applied to the reactions in the test set to create their fingerprints and report the classification performance.

For both supervised and unsupervised fingerprints, separate classifiers are created and tuned to evaluate their utility in reaction classification for pair comparison. Based on each fingerprint, we create a k-nearest neighbor (k-NN) classifier (tested with k=5, using cosine similarity) and a one-vs-rest logistic regression (LR) classifier. The training sets for these classifiers are the same reactions used to fine-tune the supervised fingerprints. TMAP [19] is used to visualize high dimensional data as the minimum spanning tree of its 10-nearest neighbor graph based on cosine similarity.

III. Results and Discussion

In this section, we demonstrate the performance of KG-based reaction fingerprints and discuss our experimental results on the two benchmark datasets.

A. Classification Performance

Tables I to IV show the results of our experiments with 5-NN and LR as classifiers. In the case of the fine-tuned RXNFP model, we include the accuracy for the classification head with the name "Head", and in the case of supervised RotatE, we include the accuracy for classification based on the enforced constraints with the name "RotatE" since no additional classifier was used. In our experiments, all fingerprints tested had a fixed size of 256 dimensions for fair comparison. For fair comparison, we compared with Transformer RXNFP fingerprint learning in the same unsupervised and supervised settings (the codes were also provided by the original authors). Note that the classic hand-crafted Morgan fingerprinting approach does not use any supervision label, so we only show its performance in the unsupervised learning setting.

All classifiers were compared according to three well-established metrics: classification accuracy, the confusion entropy of the confusion matrix (CEN) [30], and the Matthews Correlation Coefficient (MCC) [7]. Tables 1 and 2 provide

TABLE I
PISTACHIO: UNSUPERVISED FINGERPRINT PERFORMANCE METRICS

Fingerprint	Classifier	Accuracy	Overall MCC	Overall CEN
Morgan Fingerprints	5-NN	0.688	0.675	0.218
	LR	0.671	0.651	0.199
Untuned RXNFP	5-NN	0.811	0.802	0.126
	LR	0.781	0.768	0.141
Unsupervised Node2Vec	5-NN	0.815	0.809	0.114
	LR	**0.946**	**0.944**	**0.039**

TABLE II
PISTACHIO: SUPERVISED FINGERPRINT PERFORMANCE METRICS

Fingerprint	Classifier	Accuracy	Overall MCC	Overall CEN
RXNFP	**5-NN**	**0.986**	**0.986**	**0.012**
	Head	0.978	0.976	0.017
	LR	0.986	0.985	0.012
Supervised RotatE	5-NN	0.843	0.841	0.102
	LR	0.754	0.755	0.158
	RotatE	0.696	0.701	0.173
Supervised Node2Vec	5-NN	0.851	0.850	0.088
	LR	0.962	0.960	0.031

TABLE III
USPTO 1K TPL: UNSUPERVISED FINGERPRINT PERFORMANCE METRICS

Fingerprint	Classifier	Accuracy	Overall MCC	Overall CEN
Morgan Fingerprints	5-NN	0.692	0.691	0.182
	LR	0.856	0.856	0.074
Untuned RXNFP	5-NN	0.699	0.698	0.150
	LR	0.564	0.561	0.207
Unsupervised Node2Vec	5-NN	0.883	0.882	0.042
	LR	**0.943**	**0.943**	**0.020**

TABLE IV
USPTO-1K-TPL: SUPERVISED FINGERPRINT PERFORMANCE METRICS

Fingerprint	Classifier	Accuracy	Overall MCC	Overall CEN
RXNFP	**5-NN**	**0.984**	**0.984**	**0.008**
	Head	0.937	0.936	0.016
	LR	0.981	0.981	0.009
Supervised RotatE	5-NN	0.901	0.900	0.039
	LR	0.890	0.889	0.040
	RotatE	0.870	0.870	0.045
Supervised Node2Vec	5-NN	0.942	0.942	0.022
	LR	0.978	0.978	0.010

classifier accuracy for different methods on Pistachio in the unsupervised and supervised representation learning settings respectively. Tables 3 and 4 show the same classification results for USPTO 1k TPL. It is important to note that on the USPTO 1k TPL dataset, our approach has had an advantage over other methods in the unsupervised learning setting because it uses the algorithm from [2] to create the templates for the KG. At the same time, the class labels of this dataset are actually hashes of templates generated by the algorithm from [28], which is a slightly modified version of our algorithm that has been tailored to the USPTO dataset, leading to higher-quality templates. Experiments on the Pistachio dataset did not have this problem, as the reaction labels were created completely independently of the templates.

In the unsupervised setting, the reaction node embeddings from Node2Vec achieve substantially higher classification accuracy with a LR classifier than that of RXNFP. The low CEN and high MCC indicate that this high accuracy is not just due to overfitting the imbalance in our data sets. In the supervised setting, the Node2Vec embeddings performed slightly better than their unsupervised counterpart, and were comparable to RXNFP which had slightly higher accuracy. This is especially significant given that generating the Node2Vec embeddings only requires a fraction of the computation time needed by RXNFP. Even RotatE, a simpler, pairwise embedding method, achieved reasonable performance. Simply checking the geometric relationship that it enforces between reactions and labels yields higher accuracy than using LR or 5-NN classification with traditional Morgan fingerprints.

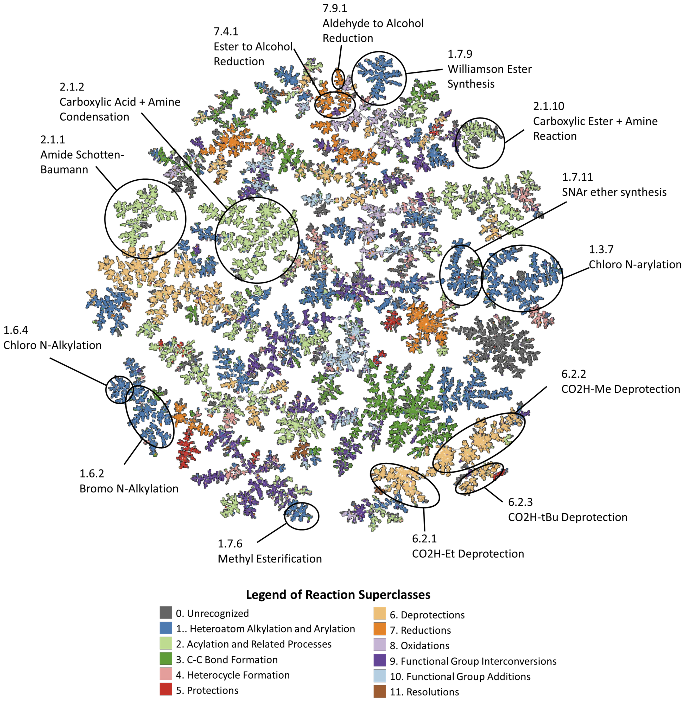

Fig. 4. Reaction atlas of unsupervised Node2Vec embeddings from the Pistachio test set, colored by superclass.

B. Visualizing the KG-based Fingerprints

Here we focus on visualizing the unsupervised KG-based fingerprints generated by Node2Vec, because they achieve the best results among all unsupervised approaches. Fig. 4 shows the TMAP diagram for a sample of the unsupervised embeddings from the Pistachio test set, with each point representing one reaction. Distance between the embeddings was defined as their cosine similarity. Except for the assignment of colors to tree nodes, the reaction labels were not used in the process of generating this figure. The only probabilistic model trained in the process was the 1-layer neural network in Node2Vec.

The colors represent one of eleven reaction superclasses [6], a general grouping over the specific labels in the database. These class assignments were not present in either of the Pistachio KGs, and were not seen at any point when generating these representations. The gray points represent uncategorized reactions. Based on Fig. 4, reactions in the same superclass tend to be clustered together in the embedding space learned

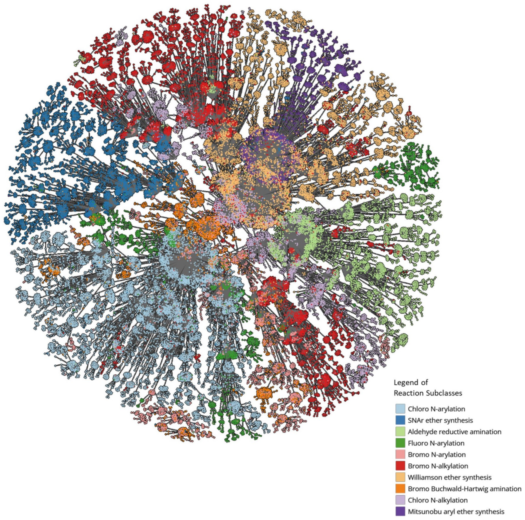

Fig. 5. Reaction atlas of heteroatom alkylation and arylation reactions from the Pistachio test set based on the reaction embeddings learned by the unsupervised Node2Vec, colored by subclass.

by Node2Vec. Further exploration of the diagram reveals that the subclasses within each superclass (1,385 in total) often separate themselves into distinct areas within the regions of their respective superclasses. For example, examining the cluster of deprotection reactions (labeled 6.x.x) depicted by the yellow branch at the bottom of the diagram, we can see that it is actually split into distinct subclusters of CO2H-Me deprotections, CO2H-tBu deprotections, and CO2H-Et Deprotections. To further demonstrate this, in Fig. 5 we focus on "Heteroatom Alkylation and Arylation" reactions (labeled 1.x.x). We plot reactions belonging to the ten most common subclasses out of the total 138, which each correspond to a different color in the diagram.

The spatial cohesion of the fingerprints indicates that their organization aligns with existing chemical knowledge. More specifically, the location of each fingerprint encodes information about the corresponding reaction's mechanism. Only rudimentary heuristics were used to explicitly relate reactions in the original data, indicating that this additional knowledge appears in the structure of the graph itself.

C. KG-Assisted Interpretation of Results

Using Node2Vec and RotatE, the spatial arrangement of node embeddings encodes the topology of the graph. In the case of Node2Vec, the embedding network learns about the graph topology through random walks, so for each reaction random walks starting from the reaction's node can be used to directly visualize the information being encoded.

Starting from reaction nodes in the unsupervised KG, we take 1000 random walks of length 10, traversing edges in both directions. During each step of these random walks, all edges are assigned equal probability except the edge traversed to find the current node, which is assigned probability 0. The nodes reached during these walks form an induced subgraph of the KG that describes the local neighborhood of the reaction, with the number of times each node was found signifying a type of "reachability" from the reaction. Under these conditions, the walks may reach several thousand nodes, so nodes that were reached less than 3 times were excluded. Then, to facilitate visualization, we perform breadth-first search from the start node to obtain a tree . The result is shown in Fig. 6.

Fig. 6 shows that reaction nodes with matching classes were found more frequently than those reactions from any other class during random walks from the original reaction node. The matching reactions were reached in several different ways: some directly shared reactants with the original reaction, others shared the same template, and some had more complex relationships.

IV. Discussion and Conclusion

These results provide compelling evidence that graphs built upon chemical data can encode far more than the information used to construct them. This fingerprint generation method is able to incorporate abstract chemical knowledge in a way that is straightforward, interpretable, and that can be tailored to specific tasks in the KG generation step, giving chemists greater control over the final fingerprints being generated. Furthermore, by enabling transductive learning (including both training and test reactions in a KG but without labels), this method can outperform all existing methods on unsupervised classification performance. With fine-tuning, it achieves comparable performance to the state-of-the-art supervised classification with significantly less computation.

This work has several limitations. In the current approach, similarities and differences between molecular parts that do not change during the reaction are not considered in the KG, so they may not be reflected in the relative distances of the reaction fingerprints. However, since such connectivity is less relevant to the chosen classification task, we do not observe performance drop. In other fingerprint applications, the KG generation process may be modified to incorporate the relevant edges. More generally, this approach uses transductive learning and cannot generate embeddings for unseen reactions. To extend this method to the inductive setting where a model can be created and applied to unseen reactions to predict their types, graph nodes can be associated with attributes, such as the structural information (e.g., SMILES), chemical properties, or even fingerprints for each compound, reaction, fragment, and template. Then, other graph representation learning methods capable of induction such as GraphSage [9] can be used to embed the nodes. This way, we can integrate both attributes of individual nodes and connectivity among nodes in the graph.

References

[1] RDKit: Open-source cheminformatics. https://www.rdkit.org.

[2] Connor W. Coley, Regina Barzilay, Tommi S. Jaakkola, William H. Green, and Klavs F. Jensen. Prediction of Organic Reaction Outcomes Using Machine Learning. *ACS Central Science*, 3(5):434–443, 2017.

[3] Connor W. Coley, William H. Green, and Klavs F. Jensen. Machine Learning in Computer-Aided Synthesis Planning. *Accounts of Chemical Research*, 51(5):1281–1289, May 2018.

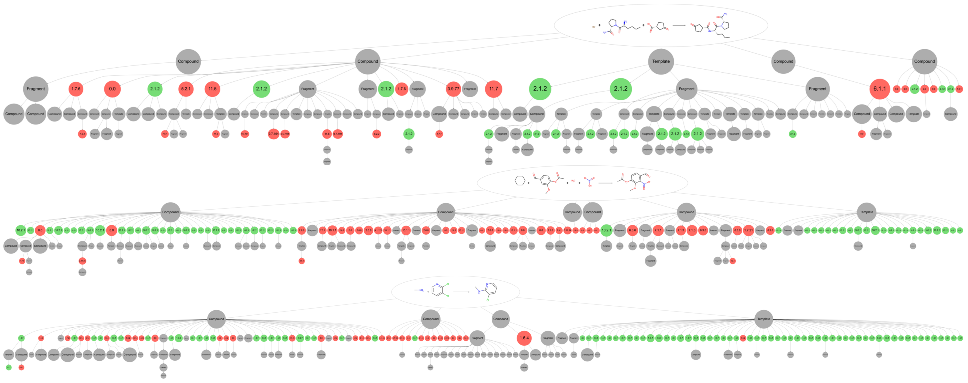

Fig. 6. Subgraph of random walks from three selected reactions (classes from top to bottom: carboxylic acid + amine condensation, nitration, chloro n-arylation). Node size corresponds to the number of times the node was reached during random walks. Reaction nodes are labeled with their finest-level classes. The color of a node is green if it matches the label of the original reaction and is red otherwise. Non-reaction nodes are gray.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. arXiv:1810.04805 [cs].

[5] Joseph L. Durant, Burton A. Leland, Douglas R. Henry, and James G. Nourse. Reoptimization of MDL Keys for Use in Drug Discovery. *Journal of Chemical Information and Computer Sciences*, 42(6):1273–1280, November 2002.

[6] Diego Garay-Ruiz and Carles Bo. Chemical reaction network knowledge graphs: the OntoRXN ontology. *Journal of Cheminformatics*, 14(1):29, December 2022.

[7] J. Gorodkin. Comparing two K-category assignments by a K-category correlation coefficient. *Computational Biology and Chemistry*, 28(5):367–374, December 2004.

[8] Aditya Grover and Jure Leskovec. node2vec: Scalable Feature Learning for Networks. *arXiv:1607.00653 [cs, stat]*, July 2016. arXiv: 1607.00653.

[9] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive Representation Learning on Large Graphs, September 2018. arXiv:1706.02216 [cs, stat].

[10] Xu Han, Shulin Cao, Xin Lv, Yankai Lin, Zhiyuan Liu, Maosong Sun, and Juanzi Li. OpenKE: An Open Toolkit for Knowledge Embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 139–144, Brussels, Belgium, November 2018. Association for Computational Linguistics.

[11] Samuel C. Hoffman, Vijil Chenthamarakshan, Kahini Wadhawan, Pin-Yu Chen, and Payel Das. Optimizing molecules using efficient queries from property evaluations. *Nature Machine Intelligence*, 4(1):21–31, December 2021.

[12] Luyang Huang, Lingfei Wu, and Lu Wang. Knowledge Graph-Augmented Abstractive Summarization with Semantic-Driven Cloze Reward. *arXiv:2005.01159 [cs]*, May 2020. arXiv: 2005.01159.

[13] Joonsoo Jeong, Nagyeong Lee, Yongbeom Shin, and Dongil Shin. Intelligent generation of optimal synthetic pathways based on knowledge graph inference and retrosynthetic predictions using reaction big data. *Journal of the Taiwan Institute of Chemical Engineers*, 130:103982, January 2022.

[14] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. Technical Report arXiv:1301.3781, September 2013.

[15] H. L. Morgan. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *Journal of Chemical Documentation*, 5(2):107–113, May 1965.

[16] Nina Nikolova and Joanna Jaworska. Approaches to Measure Chemical Similarity – a Review. *QSAR & Combinatorial Science*, 22(9-10):1006–1026, December 2003.

[17] Noel M. O'Boyle, Michael Banck, Craig A. James, Chris Morley, Tim Vandermeersch, and Geoffrey R. Hutchison. Open Babel: An open chemical toolbox. *Journal of Cheminformatics*, 3(1):33, October 2011.

[18] Daniel Probst and Jean-Louis Reymond. A probabilistic molecular fingerprint for big data settings. *Journal of Cheminformatics*, 10(1):66, December 2018.

[19] Daniel Probst and Jean-Louis Reymond. Visualization of very large high-dimensional data sets as minimum spanning trees. *Journal of Cheminformatics*, 12(1):12, February 2020.

[20] Daniel Probst, Philippe Schwaller, and Jean-Louis Reymond. Reaction classification and yield prediction using the differential reaction fingerprint DRFP. *Digital Discovery*, 1(2):91–97, 2022.

[21] Syed Sauban Ghani. A comprehensive review of database resources in chemistry. *Eclética Química Journal*, 45(3):57–68, July 2020.

[22] Roger Sayle, John Mayfield, and Ingvar Lagerstedt. Pistachio. https://www.nextmovesoftware.com, 2021.

[23] Nadine Schneider, Daniel M. Lowe, Roger A. Sayle, and Gregory A. Landrum. Development of a Novel Fingerprint for Chemical Reactions and Its Application to Large-Scale Reaction Classification and Similarity. *Journal of Chemical Information and Modeling*, 55(1):39–53, January 2015.

[24] Nadine Schneider, Daniel M Lowe, Roger A Sayle, Michael A Tarselli, and Gregory A Landrum. Big data from pharmaceutical patents: a computational analysis of medicinal chemists' bread and butter. *Journal of medicinal chemistry*, 59(9):4385–4402, 2016.

[25] Nadine Schneider, Daniel M. Lowe, Roger A. Sayle, Michael A. Tarselli, and Gregory A. Landrum. Big Data from Pharmaceutical Patents: A Computational Analysis of Medicinal Chemists' Bread and Butter. *Journal of Medicinal Chemistry*, 59(9):4385–4402, May 2016.

[26] Philippe Schwaller, Daniel Probst, Alain C. Vaucher, Vishnu H. Nair, David Kreutter, Teodoro Laino, and Jean-Louis Reymond. Mapping the space of chemical reactions using attention-based neural networks. *Nature Machine Intelligence*, 3(2):144–152, February 2021.

[27] Philippe Schwaller, Alain C. Vaucher, Ruben Laplaza, Charlotte Bunne, Andreas Krause, Clemence Corminboeuf, and Teodoro Laino. Machine intelligence for chemical reaction space. *WIREs Computational Molecular Science*, 12(5):e1604, September 2022.

[28] Amol Thakkar, Thierry Kogej, Jean-Louis Reymond, Ola Engkvist, and Esben Jannik Bjerrum. Datasets and their influence on the development of computer assisted synthesis planning tools in the pharmaceutical domain. *Chemical Science*, 11(1):154–168, 2020.

[29] Andrey A. Toropov, Alla P. Toropova, Dilya V. Mukhamedzhanoval, and Ivan Gutman. Simplified molecular input line entry system (SMILES) as an alternative for constructing quantitative structure-property relationships (QSPR). *Indian Journal of Chemistry, Section A, 44A(08)*, 2005.

[30] Jin-Mao Wei, Xiao-Jie Yuan, Qing-Hua Hu, and Shu-Qin Wang. A novel measure for evaluating classifiers. *Expert Systems with Applications*, 37(5):3799–3809, May 2010.

[31] Peter Willett. Similarity Searching Using 2D Structural Fingerprints. In Jürgen Bajorath, editor, *Chemoinformatics and Computational Chemical Biology*, volume 672, pages 133–158. Humana Press, Totowa, NJ, 2010. Series Title: Methods in Molecular Biology.