Reaction Chemistry & Engineering

AI-driven Hypergraph Network of Organic Chemistry: Network Statistics and Applications in Reaction Classification

Journal:	*Reaction Chemistry & Engineering*
Manuscript ID	RE-ART-08-2022-000309.R1
Article Type:	Paper
Date Submitted by the Author:	23-Nov-2022
Complete List of Authors:	Mann, Vipul; Columbia University, Chemical Engieering Venkatasubramanian, Venkat; Columbia University, Chemical Engineering

AI-driven Hypergraph Network of Organic Chemistry: Network Statistics and Applications in Reaction Classification

Vipul Mann [1] **Venkat Venkatasubramanian** [1]

Abstract

Rapid discovery of new reactions and molecules in recent years has been facilitated by the advances in high throughput screening, accessibility to a highly complex chemical design space, and the development of accurate molecular modeling frameworks. A holistic study of the growing chemistry literature is, therefore, required that focuses on understanding the recent trends in organic chemistry and extrapolating them to infer possible future trajectories. To this end, several network theory-based studies have been reported that use a directed graph representation of chemical reactions. Here, we perform a study based on representing chemical reactions as hypergraphs where the nodes represent the participating molecules and hyperedges represent reactions between nodes. We use a standard reactions dataset to construct a hypergraph network of organic chemistry and report its statistics such as degree distribution, average path length, assortativity or degree correlations, PageRank centrality, and graph-based clusters (or communities). We also compute each statistic for an equivalent directed graph representation of reactions to draw parallels and highlight differences between the two. To demonstrate the AI applicability of hypergraph reaction representation, we generate dense hypergraph embeddings and use them in the reaction classification problem. We conclude that the hypergraph representation is flexible, preserves reaction context, and uncovers hidden insights that are otherwise not apparent in a traditional directed graph representation of chemical reactions.

[1]Department of Chemical Engineering, Columbia University, New York, USA. Correspondence to: Venkat Venkatasubramanian <venkat@columbia.edu>.

1. Introduction

With the accelerated discovery of new reactions and complex molecules due to advances in computational methods, chemistry literature has been growing rapidly. The major drivers for this growth are the advances in molecule optimization, reaction engineering and optimization resulting in the discovery of novel reactions that were either unknown earlier or were infeasible, and high-throughput screening methods that have led to the re-engineering (or re-wiring) of existing reactions to make them more cost-effective and sustainable from an environmental standpoint. Hybrid AI models have a central role to play in driving chemistry growth by combining domain knowledge in the form of symbolic AI with numeric machine learning methods [1], thus leveraging the expertise of a chemist and the numeric stronghold of AI methods. Consequently, several hybrid AI-based methods have been reported for problems including thermodynamic property estimation [2, 3], reaction prediction and retrosynthesis [4, 5], and chemical product design among several others as presented in the excellent review articles [6, 7, 8, 9].

To condense (and make sense of) the huge amount of chemistry literature that is available to us mostly in an unstructured format, we require tools that could be used to represent this knowledge in a structured format, compute coarse-grained statistics that summarize the information effectively, identify general trends on the evolution and growth of the domain, and discover new chemistry insights that were unknown earlier. While a framework that addresses these requirements could be custom-developed, network theory naturally offers tools and techniques such as – structural statistics [10, 11], centrality measures [12], clustering [13], network embedding [14, 15], link prediction [16, 17] – that could be used to tackle these requirements. There are several variations of graph-based representations for chemical reactions, but the most common is a directed graph representation where nodes represent molecules and directed edges from reactant nodes to product nodes represent reactions. Studies based on such dyadic representations have reported several interesting properties of the reactions network such as their scale-free network structure similar to the World Wide Web (WWW) [18], the existence of core (most useful)

and peripheral molecules across organic chemistry reactions [19], the small-world nature of reaction networks [20], which is shown to make a network robust towards node/edge deletions [21]. [22, 23] demonstrated applications of network theory-based studies in parallel synthesis, reactivity estimation, and rewiring of synthetic pathways.

The traditional directed graph representation for chemical reactions has several limitations. First, a directed graph representation does not capture the complete reaction context, i.e. it introduces independent directed edges for multi-reactant (or multi-product) reactions from each reactant to each product, thus losing contextual information on the presence of other reactants (or products). As a result, several seemingly independent, directed edges might correspond to the same reaction. Second, a dyadic graph representation does not allow for reaction (or edge)-specific molecular (or node) properties such as relative molecular complexity, reactivity, stoichiometry, reaction kinetics, and other properties that might be useful for making the graph representation more complete, rich, and chemistry-aware. Third, due to the above limitations, the analyses generally could not be analyzed in a self-contained manner to draw inferences and identify the trends in chemistry that are not an artifact of the reaction representation, as observed for degree correlations in [20].

To address these limitations, we propose an alternative hypergraph representation where molecules are represented as vertices and an entire reaction is represented as a hyperedge. Since hypergraphs allow for an edge (or hyperedge) to connect multiple vertices together (and not just two), the entire reaction is represented using just a single, unique hyperpedge. To address the issue of incorporating reaction-specific node attributes, we use the recently proposed annotated hypergraph framework [24], which allows for each node to have hyperedge-specific annotations and makes the representation flexible to allow for reaction-specific contextual information. Therefore, compared with directed graphs, annotated hypergraphs are much more frugal in terms of the number of hyperedges, flexible in capturing reaction-level context, and due to the one-to-one correspondence between hyperedges and reactions, the statistics are self-contained, which correspond to underlying chemistry trends.

In this work, we compare and contrast the directed graph representation of chemical reactions with an annotated hypergraph representation using a standard organic chemistry reactions database containing nearly half a million reactions. Our work is the first attempt to study the network of organic chemistry using a hypergraph framework that we show to be frugal, rich, and chemistry-aware in nature, making them suitable for deriving chemistry inferences. To allow for a one-to-one comparison between the dyadic representation and the hypergraph representation, we compute standard network properties for the directed graph representation and an equivalent hypergraph representation using the same reactions dataset. At the same time, we also report the time-evolution of these properties. We also show how a hypergraph could be transformed into a weighted directed graph to allow for computation of dyadic network properties that may be ill-defined or difficult to compute for hypergraphs (at the moment). Finally, to demonstrate the use-case of such hypergraph representations not just for understanding chemistry trends but also for reaction engineering, we show how the hypergraph representation could be used in the reaction classification problem, i.e., predicting the reaction type given participating molecules which has applications in reaction mechanism generation, retrosynthetic planning, and feasibility analysis.

The rest of the paper is organized as follows – in Section 1, we first provide a mathematical and visual description of the directed graph and hypergraph representations using an example set of four reactions in Section 2.1, followed by a tutorial-like description of network statistics such as degree distributions, average path length, and assortativity in Sections 2.2, 2.3, and 2.4; a description of the dataset used to construct the organic chemistry networks is provided in Section 3.1 and detailed network statistics along with their time-evolution and chemistry-inferences derived are provided in Sections 3.2, 3.3, and 3.4; additional analysis based on PageRank and community detection are presented in Section 4. The application of hypergraph in reaction classification using reaction embeddings generated via random hyperwalks is presented in Section 5; finally, we present the conclusions of this study and future direction of our work in Section 6.

2. Properties of directed graphs and hypergraphs

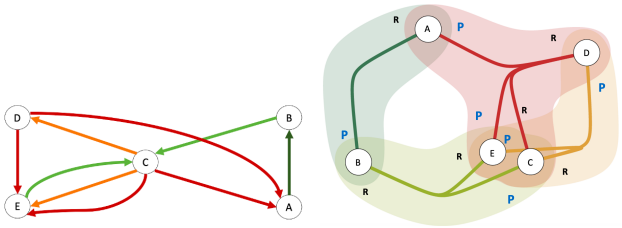
In this section, we formally define directed graphs, annotated hypergraphs, and the various network statistics that we use to characterize the hypergraph network of organic chemistry. The following sections could also be treated as a tutorial that motivates various network properties using an example set of four simple reactions containing five different molecules.

2.1. Mathematical representation

A directed graph is an ordered pair $G = (V, E)$ of a set of vertices V and a corresponding set of edges E. Each edge e_i in E connects a source node s_i to a target node t_i, giving directionality to the set of edges, thus resulting in a *directed* graph as opposed to an undirected graph. Chemical reactions could also be represented using such directed graphs where the reactants and products are represented as vertices, and directed edges from reactants to products representing

reactions. For reactions with multiple reactants and products, the directed graph is typically constructed using all-to-all wiring with all reactants of a given reaction connecting individually to all products in the reaction through independent directed edges. Figure 1(a) shows a directed graph representation for the set of four reactions ($R1, R2, R3, R4$) with 5 different molecules (A, B, C, D, E) shown in Equation 1.

$$\begin{aligned} R1: \quad & A \longrightarrow B & \qquad R3: \quad & C \longrightarrow E + D \\ R2: \quad & B + E \longrightarrow C & \qquad R4: \quad & D + C \longrightarrow A + E \end{aligned} \tag{1}$$



(a) (b)

Figure 1. (a) Directed graph-based representation (b) Annotated hypergraph-based representation where an entire reaction is represented using a single hyperedge and the annotations indicate the vertex 'roles' as product (P) or reactant (R)

On the other hand, a hypergraph is a generalization of a graph where each edge is not limited to connecting just two vertices but could connect any number of vertices via hyperedges. Mathematically, a hypergraph is a pair $H = (V, E)$ where V is a set of vertices and E is the set of edges (or *hyperedges*) where each edge contains a non-empty subset of V. Since each chemical reaction has contextual information about molecules along with an inherent directionality, we use *annotated* hypergraphs [24] with hyperedge-specific annotations (or roles) for nodes in a hyperedge. An annotated hypergraph is defined as $A = (V, E, X, l)$ where V is the set of nodes, E is a labeled hyperedge set where each hyperedge is a subset of V, X is a finite label set containing the possible set of labels (or annotations/roles), and l is a role labeling function for assigning roles to each edge in the label. It should be noted that each node v would have a given role x in given edge e, written as $l(v, e) = x$. Roles are contextual and they are assigned to node-edge pairs, unlike node attributes that are defined a priori for each node in dyadic graphs. For a set of chemical reactions, the set of vertices would be nodes, reactions containing the set of vertices participating in the reaction are represented as hyperedges, and the node-edge pair role could either be 'product (P)' or 'reactant (R)' for nodes that play the role of reactants or products in a reaction, respectively. Figure 1(b) shows the equivalent hypergraph representation for the set of four reactions in Equation 1.

Remark 1: Observe that the number of (hyper)edges in a hypergraph representation is the same as the number of reactions, but this is not the case with edges in a directed graph representation.

Remark 2: One of the primary benefits of using annotated hypergraphs is the incorporation of contextual information about reactions and molecules through hypergraph annotations or roles.

2.2. Degree distributions

Degree distributions provide a general sense of the network structure and its connectivity pattern. Generating a degree distribution involves computing the degree (or number of edges) for each node and estimating the underlying probabilistic distribution that they follow. For a directed graph, each node has two kinds of degrees – incoming degree (number of incoming edges, d_{in}) and outgoing degree (number of outgoing edges, d_{out}). The sum of the incoming and outgoing degrees, total degree ($d_{in} + d_{out} = d_{total}$), is the same as the degree of an equivalent undirected graph with directionality removed from directed edges.

For an annotated hypergraph, equivalent degree distributions could be defined. The incoming degree for a node in the annotated hypergraph would involve counting the number of hyperedges where the node participates with a role 'product' ($d_{product}$ or d_{in}) since products have incoming edges, and the outgoing degree would involve counting the number of hyperedges where the node participates with a role 'reactant' ($d_{reactant}$ or d_{out}) since reactants have outgoing edges. The sum of the incoming and outgoing degrees would be the total degree ($d_{product} + d_{reactant} = d_{total}$).

Table 1 shows the incoming and outgoing degrees for each node in the set of reactions in Equation 1 for directed graph and hypergraph representations.

Table 1. Degree distributions for the example set of reactions in Equation 1

Node	Directed graph		Annotated hypergraph	
	in	out	in (P)	out (R)
A	2	1	1	1
B	1	1	1	1
C	2	4	1	2
D	1	2	1	1
E	3	1	2	1

2.3. Average shortest path length

The average shortest path length of a network measures the separation between nodes (on average) in term of the number of edges between nodes. Since this measure involves computing the separation between *all nodes*, the network is

required to be connected, i.e., there must exist a path from any node to any other node in the network. For a directed graph, the average shortest path length is the number of directed edges between nodes with the constraint that the distance should be measured along the direction of the edges. For undirected graphs, this is simply the average number of edges between nodes, irrespective of the directionality. This is often referred to as the all pairs shortest path (APSP), and is defined as,

$$l = \sum_{s,t \epsilon V} \frac{d(s,t)}{n(n-1)} \tag{2}$$

where $d(s,t)$ is the distance between nodes s and t, and n is the total number of nodes in the network.

To define connectivity for hypergraphs, we introduce two new concepts – dual hypergraphs and linegraphs. First, the dual hypergraph H^* of a hypergraph H is a hypergraph with nodes and edges interchanged. Therefore, in an H^*, the nodes represent reactions and the hyperedges represent the set of molecules common between the nodes that it connects. Second, a linegraph $L(H)$ of a hypergraph H is defined as a graph whose vertex set is the set of vertices of H with two vertices adjacent and connected in $L(H)$ when their corresponding hyperedges have a non-empty intersection, i.e. they have common hyperedges (or reactions in our context). Therefore, a hypergraph H is said to be connected if its linegraph $L(H)$ is connected. A generalization of linegraphs is the s-linegraph where s (an integer, ≥ 1) indicates the minimum size of the intersection, thus giving rise to s-linegraphs. Because of the duality property of hypergraphs, an equivalent linegraph $L(H^*)$ could be created for the dual hyeprgraph H^* where the set of vertices represent hyperedges and adjacent vertices are connected if they have non-empty intersections, i.e. common molecules in our context. The s-linegraphs for the example set of reactions in Equation 1 for different values of s is shown in Figure 2 for H and H^*.

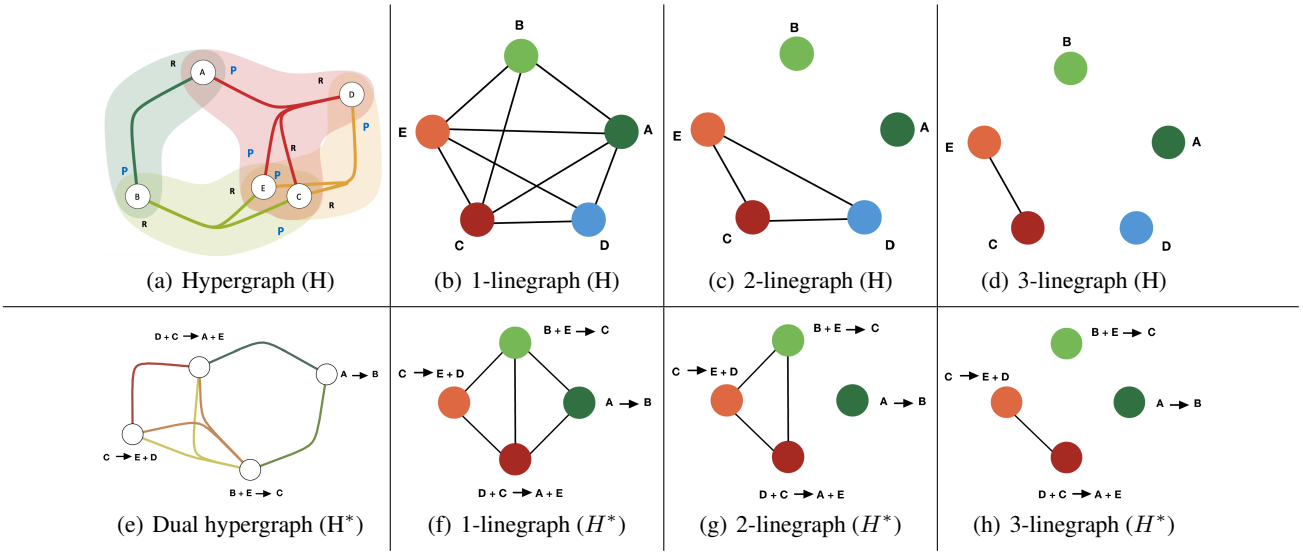
Now, for hypergraphs, the average shortest path length could be defined in the same manner as for dyadic graphs by computing the distance between nodes in an s-linegraph of H (known as s-distance). For our purpose, we generate the 1-linegraph and compute the average shortest 1-distance between the nodes using Equation 2.

Since the computation of the average shortest path length requires the graph to be connected, we find out the largest connected subcomponent both for the directed graph and the hypergraph and report their respective average shortest path lengths. For the example set of four reactions in Equation 1, since both the directed graph representation and the hypergraphs's 1-linegraph representations are connected, their largest connected subcomponents are the same as their respective graphs (or hypergraphs). The average path lengths computed for the regular (undirected) graph and the hypergraph is show in Table 2.

Remark 3: It is evident from the above table that in a hypergraph, the distances between nodes correspond exactly to the number of reactions that separate the nodes (or molecules), whereas in the case of a directed graph representation, the distance between nodes corresponds only to partial reactions separating the nodes and not the complete reactions.

2.4. Assortativity

Assortativity is a measure of the *mixing patterns* in networks that indicates the general mixing behavior of nodes with



(a) Hypergraph (H) (b) 1-linegraph (H) (c) 2-linegraph (H) (d) 3-linegraph (H)

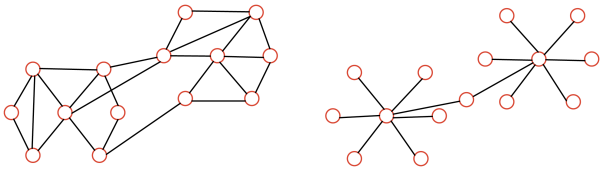
(e) Dual hypergraph (H*) (f) 1-linegraph (H^*) (g) 2-linegraph (H^*) (h) 3-linegraph (H^*)

Figure 2. The hypergraph (H), dual hypergraph (H^*), and their respective s-linegraphs for the example set of four reactions in Equation 1

Table 2. All pairs shortest distance for the example reactions

Node pairs	Graph	Hypergraph
	in	s=1
d_{A-B}	1	1
d_{A-C}	1	1
d_{A-D}	1	1
d_{A-E}	2	1
d_{B-C}	1	1
d_{B-D}	2	2
d_{B-E}	2	1
d_{C-D}	1	1
d_{C-E}	1	1
d_{D-E}	1	1
Average	**1.3**	**1.1**

other nodes in the network to give rise to a bigger network. Assortativity is defined as the degree correlations between nodes, and therefore, the mixing pattern could either be assortative (positive correlation) or diassortative (negative correlation). The assortativity is often computed as the Pearson correlation coefficient between the degrees of a pair of nodes and takes values between -1 and 1 – a network with an assortativity coefficient of -1 indicates a perfectly disassortative mixing, an assortativity coefficient of 1 points towards a perfectly assortative mixing, and an assortativity coefficient of 0 indicates a non assortative graph. Figure 3 shows an example of assortative and disassortative networks.



(a) (b)

Figure 3. Different mixing patterns (a) Assortative (b) Disassortative. Assortative networks have mixing patterns that arise due to nodes with similar degree connecting to other nodes with similar degrees, whereas disassortative networks are a result of mixing patterns where nodes with dissimilar degrees connect to each other

For a directed graph, the in-assortativity ($r_{in,in}$), out-assortativity ($r_{out,out}$), and in-out assortativity ($r_{out,in}$) measures the tendencies of nodes to connect with other nodes that have similar in-degrees, out-degrees, and out-in degrees, respectively. For $\alpha, \beta \in \{in, out\}$, the assortativity $r_{\alpha,\beta}$ for directed graphs is defined as

$$r(\alpha,\beta) = \frac{\sum_i (j_i^\alpha - \bar{j^\alpha})(k_i^\beta - \bar{k^\beta})}{\sqrt{\sum_i (j_i^\alpha - \bar{j^\alpha})^2}\sqrt{\sum_i (k_i^\beta - \bar{k^\beta})^2}} \quad (3)$$

where j_i^α is the α-degree of the source node for edge i, k_i^β is the β-degree of the target node for edge i, $\bar{j}^\alpha$ is the average α-degree of source nodes, and $\bar{k}^\alpha$ is the average β-degree of target nodes. For the annotated hypergraph, we define assortativity with respect to the roles (or annotations) in a manner similar to the directed graph representation in Equation 3, replacing the concept of edges with hyperedges and in-out degrees with role-specific (or annotation-specific) node degrees.

For the example set of reactions in Equation 1, the assortativity coefficients for the directed graph and the annotated hypergraph are reported in the Table 3 below.

Table 3. Degree assortativity coefficients for the directed and hypergraph representations for the example set of four reactions

roles pair	directed	hypergraph
p-p	-0.19	-0.43
r-r	-0.53	-0.43
r-p	0.27	0.15

Remark 4: These assortativity values could be used to answer questions such as – how likely is it for products with high degree to connect to other products with high degrees, or how likely is it that the reactants would connect to other reactants of similar degrees (appear in reactions together), and so on.

3. Network statistics on organic chemistry dataset

In this section, we study the network of organic chemistry through the lens of various network statistics defined in the previous section using a standard organic chemistry reactions database. The primary objective is to highlight the differences and similarities between the network statistics for the directed graph and the hypergraph representations. At the end of each section, we present chemistry insights that are drawn from such analyses along with the time-evolution of these properties.

3.1. Dataset description

The Jin's USPTO-reactions dataset [25] derived from Lowe's text mining work [26] for chemical reactions on the US patents office applications (1976-2016) is the primary dataset that we use to report and compare network statistics. We performed minimal preprocessing (removed incorrect, incomplete, and duplicate reactions) to allow for the network statistics to capture network properties without possibly losing information due to such preprocessing exercises. Along with information on reactants and products, the dataset also contained information on the year in which the reaction was reported, allowing us to investigate the time-evolution of the network properties. The final dataset contained 487,724 *single-product reactions* containing in-

Table 4. Network structure overview for the directed and hypergraph representation for the USPTO dataset

	all		1976-1985		1985-2005		after 2005	
	graph	hypergraph	graph	hypergraph	graph	hypergraph	graph	hypergraph
Num reactions	487,724		69,692		259,214		158,818	
Num (hyper)edges	1,245,533	487,724	106,977	69,692	389,072	259,214	289,623	158,818
Num nodes	440,207	440,207	71,268	71,268	238,872	238,872	180,348	180,348

formation on participating reactants, major products of each reaction, and the year in which the reactions were reported.

Using this dataset, we construct directed graph and an annotated hypergraph-based networks of organic chemistry. The directed graph representation was constructed using the all-to-all node connectivity for each reaction. The other wiring possibilities are one-to-one or many-to-one but it has been shown previously that the actual connectivity pattern does not change the network structure and properties [18, 20]. The annotated hypergraph, on the other hand, represents all the reactants and products as part of the same hyperedge with node annotations based on

- reaction roles: 'reactant' or 'product'
- relative length of SMILES strings *in a reaction* with respect to the median SMILES length per reaction: 'SMILES_short', 'SMILES_medium', 'SMILES_long'
- molecular weight *across the entire dataset*: 'molwt_light', 'molwt_medium', 'molwt_heavy'

To perform an analysis of the time-evolution of network properties over different stages of chemistry research, we split the data into three time regimes – regime 1 with reactions reported from 1976 to 1985, regime 2 with reactions reported after 1985 until 2005, and regime 3 with reactions reported from 2005 until 2016. An overview of the directed graph and hypergraph representation obtained using the entire dataset and also using dataset in the three time-regimes is presented in Table 4.

Remark 5: Note that in the case of the hypergraph, the number of hyperedges exactly equals the number of reactions in the dataset, whereas for the graph representation, the number of edges is much higher. Of course, the number of nodes remain the same in both the representations since each node corresponds to a unique molecule in both the representations.

3.2. Degree distributions

3.2.1. Degree distribution comparison

We first compare the degree distributions of both the incoming and outgoing degrees for the directed graph and annotated hypergraph representations. Recall from Section 2.2 that for the annotated hypergraph, the incoming degree is the same as the node-degree for annotation 'product' and the outgoing degree is the same as the node-degree for annotation 'reactant'. The degree distributions for the directed graph and for the various annotations in the hypergraph (based on reaction roles, relative SMILES length, and molecular weights as defined in the foregoing section) are presented respectively in Figures 4 and 5.

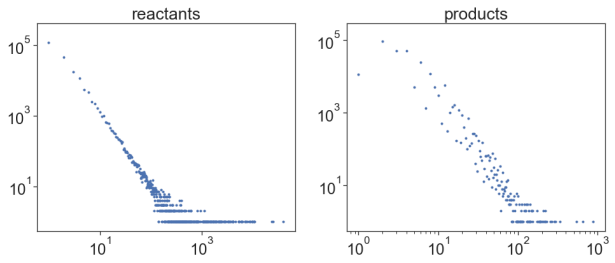


Figure 4. Degree distributions for outgoing (reactants) and incoming (product) edges in a directed graph

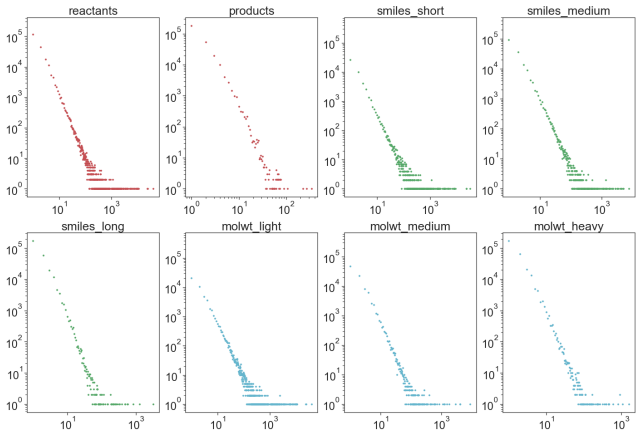


Figure 5. Degree distributions for the various hypergraph node-annotations (or roles)

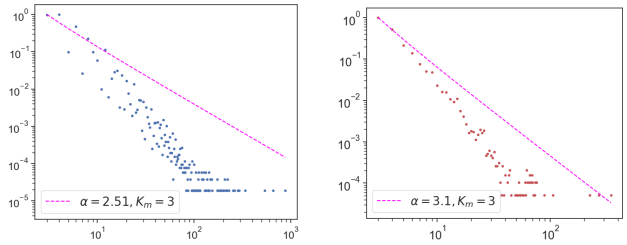
Remark 6: Note that since our dataset only contains single-product reactions, the outgoing degree distribution (reactants) is the same in both representations, and only the incoming (product) degree distributions differ. This is because the directed graph representation for a reaction would have as many incoming edges for the product as the number of reactants whereas in the hypergraph representation the product would have just one incoming edge.

3.2.2. Power law fit for degree distributions

A visual inspection of the degree distributions indicates a possible power law distribution, which is defined as

$$p(k) \propto k^{-\alpha} \tag{4}$$

where $p(.)$ is the degree distribution, k is the degree, and α is the scale-free or power law distribution parameter. The existence of a power law distribution points towards an underlying network structure known as the scale-free network structure [11], ubiquitous in real-world networks that often results in 'small-world' behavior. We perform a mathematically rigorous fit to ensure the existence of a power law using the powerlaw package in Python and estimate the underlying scale-free distribution parameter. The power law fit for the incoming degrees (products) for the directed graph and the hypergraph-based network of organic chemistry are shown in Figure 6.



(a) (b)

Figure 6. Scale-free distribution fit on incoming (products) degrees for (a) Directed graph (b) Hypergraph; K_m is the minimum degree cutoff threshold that is required as a hyperparameter in the powerlaw package

We observe that the degree distributions for both the directed graph and hypergraph incoming degrees could be assumed to be coming from a power law distribution, thus pointing towards an underlying scale-free network structure, agreeing with several other studies that have shown that chemistry networks exhibit a scale-free or small-world behavior [20, 18, 22]. However, the scale-free parameter, α differs in both the cases – α is 2.51 for the directed graph (close to 2.7 reported in [22, 18] on another reactions dataset) and 3.1 for the hypergraph.

In order to ascertain the difference in α values for the degree distributions, we estimate the scale-free parameter by randomly sub-sampling different fractions of the network in a step-forward manner in time, i.e., by utilizing the reaction year information, we sample reactions starting from 1976 sequentially sampling additional reactions from the following years. We sub-sample $0.1 - 1.0$ fraction of the network in steps of 0.1 and repeat this 10 times to perform bootstrapping and compute the deviations in α. The results are presented in Table 5. It is clear from the table that the scale-free distribution is indeed different in the two representations and remains the same irrespective of the fraction of network sub-sampled for estimating the distribution.

Table 5. Scale-free distribution parameter values, α, for different fractions of the network sampled using step-forward sampling in time using 10 bootstrapped samples for each fraction

frac	graph α		hyeprgraph α	
	mean	std	mean	std
0.1	2.54	0.0009	3.1	0.0014
0.2	2.54	0.0003	3.18	0.0013
0.3	2.48	0.0008	3.13	0.0021
0.4	2.48	0.0005	3.1	0.0013
0.5	2.47	0.0001	3.02	0.0003
0.6	2.48	0.0004	3.03	0.0014
0.7	2.49	0.0005	3.04	0.0014
0.8	2.5	0.0003	3.07	0.0005
0.9	2.51	0.0001	3.1	0.0001
1.0	2.52	0.00014	2.97	0.0034

3.2.3. Time-evolution of scale-free network property

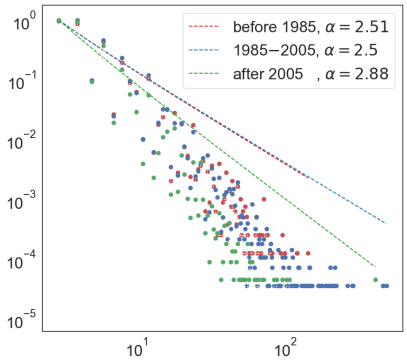
Next, we study the time-evolution of the scale-free parameter α by computing it across the three time regimes – before 1985, 1985 – 2005, and after 2005. The degree distributions, power law fit, and the estimated α values for the power law fit are show in Figure 7. We observe that the scale-free parameter α has been increasing over the years with significant increase post 2005, pointing towards accelerated growth nature of the hypergraph network [10] and a similar observation has been made on reactions dataset in [22]. The accelerated growth of the network of chemistry is also evident from the average path length analysis presented in Section 3.3.

3.2.4. Inferences from degree distributions analysis

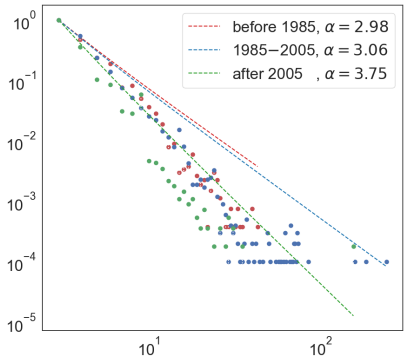
First, we observe that the degree distributions in both the cases follow a scale-free distribution, pointing towards an underlying mechanism of 'preferential attachment' or 'preferential linking' where new nodes attach to existing nodes in the network with probability proportional to their connectivity or node degrees. Mathematically, preferential attachment is characterized by

$$\Pi(k) \sim k^c \tag{5}$$

where $\Pi(k)$ is the probability of a new node attaching to an existing node with degree k, and c is a constant controlling the degree of non-linearity in preferential attachment. This expression translates to the inference that chemistry growth



(a) Directed graph incoming degrees



(b) Hypergraph 'product' degrees

Figure 7. Scale-free fit for reactions reported in the three regimes with estimated α in inset

is largely driven by a relatively small set of highly important molecules that are highly connected (higher degree, k) and they have a higher likelihood of playing a central role in the discovery of new molecules or reactions because of the underlying phenomenon of preferential linking.

Second, for the directed and hypergraph representations, the parameter characterizing the scale-free distributions is higher for the latter. This could be related to chemistry by looking at the concept of 'initial attractiveness' in scale-free networks that assigns a non-zero probability of connecting to an isolated node, given by

$$\Pi(k) = A + k^c \tag{6}$$

which ensures that, for non-zero values of A, $\Pi(k) \neq 0$ for disconnected nodes. The presence of A in the expression for preferential attachment $\Pi(k)$ does not affect the scale-free structure of the network but has a direct-impact on the α parameter as,

$$\alpha = 2 + (k1 + A)/k2 \tag{7}$$

where $k1, k2$ are constants with values depending on the underlying generating model and A characterizes the initial attractiveness of nodes. Thus, it could be inferred from Equation 7 that the initial attractiveness based on the hypergraph representation is higher than that of the directed-graph representation since the former has a higher α of 3.1 characterizing the scale-free distribution compared with α of 2.51 for directed-graph representation. Moreover, the gradually increasing α values for the scale-free distribution in both directed and hypergraph representations indicates that the initial attractiveness has been increasing over time, with the trend being much more evident in the latter representation where α grew from 2.98 in regime 1 to 3.75 in regime 3.

Third, a higher initial attractiveness translates to a higher likelihood of discovering new connections (or reactions) to isolated nodes (rare or complex molecules). Since the initial attractiveness is the highest and much different in regime 3 (after 2005) than the other two regimes, it could be inferred that in the recent years, there has been an emphasis on the rewiring of existing reactions to create connections between previously disconnected nodes, or the synthesis of rarer molecules. It will become clear from the analysis in the next section on average shortest path length that the major driver of chemistry evolution in the recent years is the rewiring of existing reactions.

3.3. Average path length

3.3.1. Average path length comparison

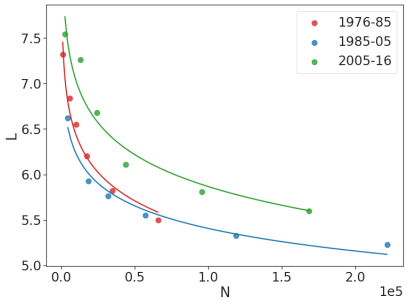
The average separation between the molecules (vertices) in terms of number of reactions (edges) is captured by the average path length of the network. We compute the average path length on the largest connected subgraph for both the representations. Recall from the Section 2.3 that for the hypergraph, in order to make a one-to-one comparison, we choose $s = 1$ to generate a 1-linegraph and compute the 1-distance between nodes to compute the average shortest path length for the hypergraph. The average shortest path lengths for the largest connected subgraph obtained for the two representations for different fraction of nodes sampled from the entire dataset using step-forward sampling is shown in Table 6.

Table 6. All pairs shortest path (or APSP) on the entire dataset

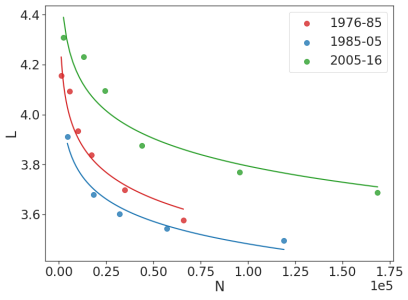
Fraction	Reactions	Directed		Hypergraph	
		nodes	APSP	nodes	APSP
1%	4,877	7,528	6.62	7,516	3.99
5%	24,386	26,222	5.98	26,176	3.75
10%	48,772	47,043	5.69	47,069	3.64
20%	97,544	91,903	5.36	91,870	3.52
50%	243,862	209,790	5.16	209,790	3.37[2]
100%	**487,724**	**411,396**	**5.11**	**411,396**	**3.25**[2]

3.3.2. Time evolution of Average path length

Similar to the degree distribution analysis, we study the time-evolution of the average path lengths of the networks in the three time regimes. The average path length as a function of the number of nodes in the network using time-based step-forward sampling is shown in Figure 8 for both the representations for different fractions of the networks, namely $1\%, 5\%, 10\%, 20\%, 50\%$ and 100% of the network in each regime. We observe that across both the representations, the



(a) Graph



(b) Hypergraph

Figure 8. Average shortest path lengths for various regimes as function of the number of nodes in the sub-sampled graph

average path length between the nodes decreases exponentially as the number of nodes in the networks is increased. Moreover, in both the cases, the average path lengths for the time regimes 1 and 2 are very similar to each other but the average path length for regime 3 is significantly higher than those in other two across all values of N. The phenomenon of decreasing path length as number of nodes is increased has been reported in the literature as network densification [27] where the network grows more and more dense over time – this makes sense for the USPTO dataset containing patented reactions where the nodes are mostly sparsely connected and they get more connected over time after either new reactions discovered, or existing nodes become more connected.

[2]extrapolated values since the network size was prohibitively large for the hypernetX package in python with no C-optimized libraries

In terms of differences, the average path length is much smaller for the hypergraph representation than for the directed graph. This could be an outcome of the frugal representation of hypergraphs where number of edges is the same as number of reactions but that is not the case with graphs. This is one of the major advantages of using hypergraphs – the edges-based analysis has a one to one correspondence with reactions-based analysis, meaning that the separation in terms of hyperedges between nodes corresponds exactly to the separation between molecules in terms of reactions. Therefore, the average separation between nodes in a hypergraph not only differs from a dyadic graph representation, but the separation corresponds to the number of reactions (on average) the separate the nodes in the network.

3.3.3. Inferences from average path lengths analysis

First, we observe that as expected, the average shortest path length for the directed graph representation is higher than that of the hypergraph representation. This again is an artifact of the directed graph representation which introduces several additional edges for each reaction depending on the number or reactants and products per reaction. On the other hand, in the hypergraph representation, since each hyperedge connects all the molecules taking part in a reaction using a single hyperedge, the separation exactly equals the number of reactions separating any two given nodes. Thus, in the directed graph average shortest path length for the entire network is 5.11 while in the hypergraph it is 3.25.

Second, the average all pairs shortest distance for the hypergraph could be interpreted as separation between nodes (or molecules) in terms of number of reactions. Recall that since each hyperedge corresponds to a unique reaction, there exists a one to one mapping between the number of reactions and the number of hyperedges separating the molecules. Thus, the hypergraph network of organic chemistry indicates that the network of organic chemistry is much more compact than previously understood with nearly 3.25 degrees of separation between molecules, pointing towards an even stronger small-world nature than previously observed with five degrees of separation [20].

Third, in both the cases, the time-evolution of the network suggests that over time, network densification takes place primarily due to the creation of links between existing nodes in the network rather than by the addition (or discovery) of new nodes. A characteristic of network densification is shrinking diameter [27], i.e., the average separation between nodes decreases as the network grows, similar to the exponential decrease in average shortest path length reported in Table 6 and Figure 8. This phenomenon is observed for both the representations and across time-regimes, pointing towards an underlying process causing the densification.

There exist models for explaining such densification such as the community guided attachment similar to preferential attachment but at a bigger community (or cluster) level with separation between the communities [27]. However, the exact quantitative model guiding densification in reaction chemistry network needs further studies. Nevertheless, densification suggests that chemistry has been evolving mostly based on the rewiring of existing reactions (edges) rather than the discovery of completely new molecules (nodes addition), that has brought the molecules closer to each other over time. This is intuitive for the reaction patents dataset that we worked with since most molecules are initially well separated given that they are patented molecules/reactions which get more connected (reachable) over time due to the discovery of new reactions over the years.

Third, the time-evolution analysis of the average shortest path length in Figure 8 suggests that in regime 1 and 2, the average separation between molecules was nearly the same for a given number of nodes, N in the network. However, in regime 3, there was a significant upward shift of average separation across all values of N. This suggests that the time-regime post 2005 is characterized by the discovery of complex chemistry leading to the synthesis of molecules via complex routes that has led to the increase in their average separation, possibly due to significant advances in computational capabilities around this time. This increase in average separation is more evident in the hypergraph representation than the directed-graph representation.

3.4. Assortativity

3.4.1. Assortativity comparison

To understand the mixing patterns of nodes in the two network representations, we compute the assortativity values between different node-type (or role) combinations – 'in' and 'out' degree roles for directed graphs and pairwise roles-based node degrees for the annotated hypergraph. Table 7 shows the assortativity values for the two representations on the entire network. The assortativity values for the two representations agree qualitatively with each other but differ in terms of their relative strengths. From Table 7, we see that in the hypergraph representation, the reactant nodes exhibit strong assortative mixing (out-out). On the other hand, the product-product and reactant-product exhibit very weakly assortative or non-assortative behavior pointing towards a lack of degree correlation between such nodes. Owing to the flexibility offered by annotations in the hypergraph, we computed additional assortativity between node roles of reactant/product with roles based on molecular weights and relative SMILES length of molecules, as shown in Tables 8 and 9. We observe that reactant-MW_{light} and reactant-$\text{SMILES}_{\text{short}}$ exhibit strong assortative mixing whereas this is not usually the case with other node-role pairs.

Table 7. Assortativity values on the entire dataset

node-pairs	directed graph	hypergraph
in-in	0.0107	0.0734
out-out	0.0049	0.1159
out-in	0.0187	0.0032

Table 8. Hypergraph assortativity between reactant & product roles with roles based on molecular weights

	MW_{light}	MW_{medium}	MW_{heavy}
reactant	0.1337	0.0074	0.0061
product	0.0119	0.0003	0.0004

Table 9. Hypergraph assortativity between reactant & product roles with roles based on relative SMILES lengths

	SMILES_{short}	SMILES_{medium}	SMILES_{long}
reactant	0.1782	0.0713	0.0015
product	0.0237	0.0083	-0.0009

3.4.2. Time evolution of assortativity

To study the evolution of mixing patterns in the network over time, we study the time-evolution of assortativity during the three time regimes. The assortativity values for the in-in, out-out, and out-in node-role pairs are shown in Table 10 below.

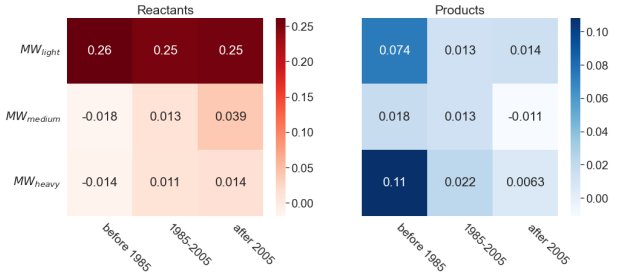
We observe from the table above that the directed-graph representation does not show any strong trend in various assortativity values, an observation also reported in [20]. On the other hand, the hypergraph representation shows a decreasing assortativity of in-in nodes over time and an increasing assortativity of out-in nodes. A further analysis on additional assortativity values with different node-role pairs reveal additional trends as shown in Figure 9. We observed that reactants show assortative mixing with nodes with MW_{light} across time regimes, whereas products show assortative mixing with MW_{heavy} before 1985. Similarly, we also observe that reactants show assortative mixing with nodes with SMILES_{short} and SMILES_{medium} across time regimes, whereas products show assortative mixing with SMILES_{short} and SMILES_{long} before 1985 and with SMILES_{medium} from 1985-2005.

3.4.3. Inferences from assortativity analysis

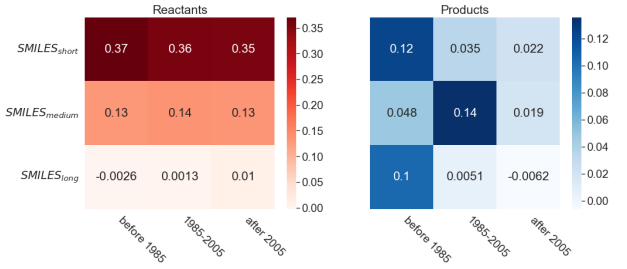
The assortativity analysis highlights another limitation of the directed-graph representation in terms of obscuring the underlying network characteristics induced by the network wiring scheme. Based on the observations in Table 7 for the directed-graph representation, it appears that the network is

Table 10. Time evolution of assortativity for the directed graph and hypergraph for various node-role pairs

node-pairs	Directed graph			Hypergraph		
	before 1985	1985-2005	after 2005	before 1985	1985-2005	after 2005
in-in	0.0630	0.0157	0.0175	0.3623	0.2302	0.1674
out-out	0.0047	0.0042	0.0069	0.2368	0.2241	0.2259
out-in	0.0274	0.0257	0.0286	0.0005	0.0057	0.0076



(a) Assortativity between MW and reactant/product roles. Reactants show assortative mixing with nodes with MW_{light} across time regimes, whereas products show assortative mixing with MW_{heavy} before 1985.



(b) Assortativity between relative SMILES length and reactant/product roles. Reactants show assortative mixing with nodes with SMILES_{short} and SMILES_{medium} across time regimes, whereas products show assortative mixing with SMILES_{short} and SMILES_{long} before 1985 and with SMILES_{medium} from 1985-2005.

Figure 9. Time evolution of assortativity for reactants and products with respect to additional node annotations (or roles)

non assortative or very weakly assortative with respect to all the node-role pairs. It was shown in [20] that this is an artifact of the network preprocessing and the assortativity values change drastically if one chooses to perform network preprocessing to remove parallel edges. On the contrary, the hypergraph representation shows that the network is assortative with respect to certain node-role pairs such as out-out degree assortativity indicating that commonly used reactants tend to take part in reactions together.

Second, due to the flexibility of the hypergraph representation in terms of allowing additional node annotations, we performed additional assortativity analysis with respect to different node-role pairs as shown in Tables 8 and 9. It was observed that reactants are assortative with molecules of light molecular weight and relatively short/medium SMILES length, highlighting the mixing patterns of reactant nodes in the network. Products, on the other hand, seem to be non-assortative with these properties, thus highlighting the wide spectrum of products with varying degrees of complexity present in the dataset.

An analysis of the time-evolution of assortativity presented in Table 10 shows node-mixing trend across time-regimes, with no clear trend in assortativity for directed-graphs. However, from the hypergraph representation it is observed that the reactants exhibit assortative mixing at nearly the same level across time regimes, whereas the products show a decreasing assortativity over time. The latter points towards the general trend in earlier years (regime 1) to discover several different routes for synthesizing a given molecule, which has been decreasing over the years (but still significant) due to the synthesis of new products molecules with different chemistry.

Finally, based on the time-evolution of assortativity with respect to additional node annotations in Figure 9, we observe that reactants are assortative at the same level with heavy molecular weight as well as relative molecular complexity across time regimes, with decreasing assortativity as the molecular weight or complexity is increased. Products on the other hand, show a positive assortativity before 1985 with heavy and complex molecules, in regime 2 assortative with medium complexity, and non-assortative in regime 3 with all roles. The latter indicates towards the diversity of products synthesized in the recent years.

4. Additional hypergraph statistics

Even though many dyadic network properties could also be defined equivalently for hypergraphs, sometimes it is necessary to work with the directed graph framework for reasons among – interpretability from a traditional graph-theoretic standpoint, easy availability of tools for computation of dyadic properties, or aversion towards adopting hypergraphs due to their seemingly high complexity. The annotated hypergraph could, therefore, be projected as a directed graph with edge-weights defined using a role-interaction kernel [24]. The role-interaction kernel defines the mapping of the annotated hypergraph to a projected-directed graph, that

maps various nodes to annotations in the hypergraph using weighted edges. We work with the following three kernels:

- $R1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$: each hyperedge split into multiple weighted directed edges from reactants to products each with weight 1; emphasis is on forward reactions only
- $R2 = \begin{bmatrix} 0 & 0.75 \\ 0.25 & 0 \end{bmatrix}$: each hyperedge split into multiple weighted directed edges with directed edges from reactants to products with weight 0.75 and also directed edges in the reverse direction (from products to reactants) with weight 0.25; unequal emphasis on forward and inverse reactions
- $R3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$: each hyperedge split into multiple weighted directed edges in the reverse direction (from products to reactants) each with weight 1; emphasis on inverse reactions only

Using such projected dyadic graphs, we perform two additional studies on the entire network – first, a PageRank [12] analysis of reaction nodes to identify the most important molecules, and second, a graph-based community-detection (or clustering) [13] to identify clusters in the reaction networks based on their connectivity patterns.

4.1. PageRank analysis

The PageRank algorithm was originally proposed for ranking of webpages on the internet [12] based on the number and quality of links to webpages and is based on a random-surfer model that performs random walks along incoming and outgoing edges from webpages. A page that has a higher likelihood of being visited by a random surfer is therefore considered more important by PageRank, thus requiring both higher connectivity as well as connectivity to other important webpages for higher PageRank.

Extending the idea of PageRank to chemical reactions and molecules, we could find the set of molecules that are most important based on their connectivity (high reactivity) as well as their connectivity to other important molecules (chemical importance due to ease of synthesizability or criticality for other compounds). Thus, a molecule with high PageRank in a network of chemical reactions should be crucial both from a reactivity/synthesizability as well as reachability/criticality standpoint. In contrast, a molecule with merely the highest degree does not say much about the molecule except that the molecule participates in many reactions.

Using the three role-interaction kernels – R1, R2 and R3 defined above, we compute the PageRank and degree centrality of nodes in the resulting network defined as d_v/d_{max} where d_v is the degree of node v and d_{max} is the maximum degree across all nodes in the network. Since PageRank and degree centrality are two different measures, their absolute values should not be compared and only the relative values or ranked order of molecules should be compared. The top-5 molecules based on PageRank and degree centralities computed using the weighted directed reaction networks obtained using different role-interaction kernels are shown in Figure 10.

Based on the above ranked order of molecules, we first observe that the molecules that are important from a PageRank standpoint are not the same as those important from a degree centrality standpoint. Second, across the role interaction kernels, the ranked order changes, i.e., molecules critical based on R1 kernel-based projection (forward edges) of hypergraph differ from those based on R3 kernel-based projection (retrosynthetic edges). This highlights the flexibility of the hypergraph reaction representation in incorporating custom importance for forward and retrosynthetic reaction directions through role-interaction kernels. Such an analysis of molecular importance in a reaction network would have application in optimizing reaction networks, designing robust supply chain networks, and performing efficient product design.

4.2. Community detection analysis

To study the formation of communities or clusters in the reaction network based on the mutual connectivity patterns and node-densities, we perform graph-based clustering on the network of reactions. We use the Leiden algorithm [13] to perform optimal graph partitioning that results in well-connected set of dense nodes in the network (called communities) and is a suitable algorithm for weighted, directed networks. For this study, we use the R2 role-interaction kernel to preserve both forward and retrosynthetic edges but with unequal weights in the network. The applications of such a graph-based community detection exercise is to get a general sense of the distribution and connectivity patterns of reactions in a large reactions dataset and understand the possible different types of reactions in the absence of any other information about the reactions. Note that the projected hypergraph is a dyadic, weighted directed graph obtained by using a role-interaction kernel that decomposes a hyperedge into a set of weighted directed edges. We perform community detection on such projected hypergraph. The alternative is to perform clustering directly using the hypergraph representation. However, given the scale of the hypergraph network of organic chemistry, the current clustering methods are computationally prohibitive.

For the entire network, the Leiden algorithm identifies nearly $65,000$ communities with a size-distribution as

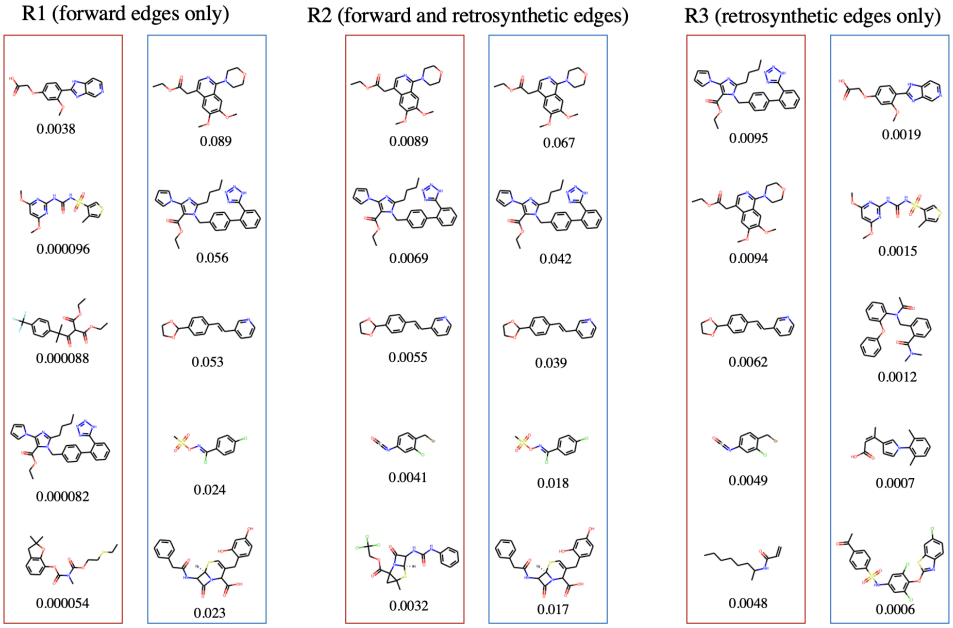


Figure 10. PageRank and degree centrality analysis for the three role interaction kernels with $R1 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$, $R2 = \begin{bmatrix} 0 & 0.75 \\ 0.25 & 0 \end{bmatrix}$, and $R3 = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$ corresponding to forward edges only, forward and retrosynthetics edges, and retrosynthetic edges only.

shown in Figure 11(a); the top-8 largest communities are shown in Figure 11(b) with different color for each identified community, and the top-100 communities are show in Figure 11(c).

We observe from Figure 11(a) that most of the communities are really small in size consisting of less than 10 reactions, whereas there are around 8 biggest communities containing over 60 reactions in each of them, as shown in Figure 11(b). The close-knit nature of these communities point towards a possible segregation of different types of reactions just based on their connectivity patterns and the nodes (molecules) that take part in those reactions. This is the idea that we utilize to perform reaction type classification in the next section. Finally, the top-100 communities visualized in Figure 11 shows clear regions of high density with highly connected and localized clusters, and regions of low density further apart from the biggest clusters. In addition, it is also observed that there is a cluster that is completely separated from all the other communities and is therefore an *island* community. The existence of core-periphery regions in the reaction network was also shown in [19] but the analysis was not based on graph clustering but on identifying strongly connected components in the network by representing reactions using a single directed edge from the heaviest reactant to the heaviest product in each reaction. In the community detection algorithm that we work with, we take into account the directionality as well as the weights of the edges, making it more flexible and the results more generalizable.

5. Application in reaction class prediction problem

In the foregoing sections, we have shown how the hypergraph representation could be used to uncover hidden insights contained in large reactions datasets and study their time-evolution through network-theoretic properties. In this section, we demonstrate the usefulness of the hypergraph representation in capturing the context of reactions and thereby their reaction type or class. We, therefore, use the hypergraph representation in the reaction-type classification problem where the objective is to estimate the reaction class from a given set of reactants and products. This problem has practical applications in retrosynthetic planning where several different routes could be eliminated just by knowing the possible reaction types. The other problems where such a problem would find significance is the reaction feasibility estimation problem where the objective is to estimate the feasibility of a reaction given the possible participating molecules. Other studies that have proposed data-driven frameworks for reaction classification problems are [28, 29, 30].

5.1. Dataset description

For this problem, since we require reaction class information for reactions, we use a subset of the USPTO reactions dataset that is typically used for retrosynthesis problem, containing about 50K reactions annotated with their corresponding reaction class from 10 possible classes. We generate the equivalent reaction hypergraph network for this dataset and work with the largest connected component in the hypergraph since the hyperedge (or reaction) embedding framework that we use to represent reactions subsequently in the classification framework is dependent on the connectivity and neighborhood contextual information. The distribution of reactions across different reaction classes in the sub-hypergraph is shown in Table 11 below.

Table 11. Distribution of reactions across different reaction classes in the largest connected subcomponent

Rxn class	Rxn name	Num rxns
1	Heteroatom alkylation and arylation	11,526
2	Acylation and related processes	8,488
3	C–C bond formation	3,909
4	Heterocycle formation	588
5	Protections	646
6	Deprotections	760
7	Reductions	459
8	Oxidations	305
9	Functional group interconversion (FGI)	1,168
10	Functional group addition (FGA)	196

5.2. Reaction embeddings using random hyperwalks

To perform reaction classification by training a data-driven classifier, we need numeric representations for reactions that are generated from their hypergraph representations and would, therefore, be used as features to train a classifier. We generate hyperedge (or reaction) embeddings by adapting the deep hyperedges framework [15] and modifying it to incorporate the contextual information contained in chemical reactions, as explained in the pseudocode provide in Algorithm 1.

The hyperedge embeddings are generated by performing random hyperwalks that capture the co-member information in each vertex by traversing hyperedges in the hypergraph network of chemical reactions. For each hyperedge, the hyperwalk starts at a randomly selected reactant node that is part of the current hyperedge, and either hops to a node in the adjacent hyperedge or stays in the current hyperedge to select another reactant node in the current hyperedge. This is repeated until the desired length of the hyperwalk is achieved. The adjacent hyperedge traversal is done only with respect to reactants since this would – first, differentiate reactants from products, and second, mimic chemistry more realistically where only those reactions are accessible where either the current reactants participate as reactants or the product of the current reaction participates as reactant. Such a random hyperwalk would closely mimic a chemist performing experiments randomly.

Formally, we start at a node v_m selected at random with the annotation 'reactant' in a hyperedge e_i. The probability of traversing an adjacent hyperedge is inversely proportional to the cardinality of the current vertex; i.e. $p = min(\frac{\alpha}{|v_m|} + \beta, 1)$ where α and β are tunable hyperparameters and $\mid v_m \mid$ is the cardinality of the vertex v_m. As in a random walk, if p is less than a randomly generated number, the traversal is performed to an adjacent hyperedge; otherwise the current hyperedge is added to the random walk and the next is chosen randomly from the adjacent hyperedges of the current vertex v_m. For each hyperedge e_i, we construct 50 random walks of length 50 each. Examples of such hyperedge random walks on the four example reactions in Equation 1 is shown in Table 12. The hyperwalks are then embedded into dense vectors of dimension R^{256} using skip-gram approach for generating embeddings [31]. At the end of the hyperedge embedding exercise, we would have a 256 dimensional vector for each hyperedge in the network.

The pseudocode for the hyperwalk generating algorithm is presented in Algorithm 1 and example hyperwalks using the example set of four reactions in Equation 1 is shown in Table 12. A 2D visualization of the resulting 256-dimensional hyperedge embeddings on the entire dataset of reactions is visualized in Figure 12.

Algorithm 1 Pseudocode for generating random hyperwalks for each hyperedge in the hypergraph

```
Input :
        walkLength:length of each hyperwalk
        hyperEdges: set of hyperedges in the hypergraph
        vertexMemberships: membership dictionary for each vertex indexed by vertex id and
        vertex role (product, reactant)
        α and β: probability distribution parameters
Initialize :
        walks_all = [ ] ;            // stores all the hyperwalks generated
for hyperedge_id in hyperEdges do
    curr_walk = [ ] ;       // stores hyperwalk for the current hyperedge
    hyperEdge = hyperEdges[hyperedge_id]
    curr_vertex = randomly chosen 'reactant' vertex in hyperEdge ;          // hyperwalk
      always starts from 'reactant' nodes
    curr_hyperEdge = hyperEdge
      while len(hyperWalk) < walkLength do
        proba = α/len(vertexMemberships[curr_vertex]['reactant']+
          vertexMemberships[curr_vertex]['product'])+β
          if random.random() < proba then
            adjacent_vertices = curr_hyperEdge['reactant']+curr_hyperEdge['product']
              ;       // switch to one of the adjacent vertices in
              current hyperedge
            curr_vertex = random.choice(adjacent_vertices)
          end
          curr_walk.append(curr_hyperedge)
          adjacent_hyperedges = vertexMemberships[curr_vertex]['reactant'] ; // adjacent
            hyperedges defined with respect to reactant roles
          curr_hyperedge = random.choice(adjacent_hyperedges) ; // randomly choose
            from one of the adjacent hyperedges
    end
    walks_all.append(walk_hyperedge)
end
Output : walks_all
```

Table 12. Two example hyperwalks generated for each reaction (hyepredge) in the example set of reactions. For each walk, $v_i \xrightarrow{e_k} v_j$ represents a walk along hyperedge e_k via nodes v_i and v_j. The hyperwalks for each hyperedge are the sequential collection of such e_k's starting at that hyperedge.

$\mathbf{H_{id}}$	Hyperwalk
0	$B,0 \xrightarrow{1} C \xrightarrow{2} C \xrightarrow{2} C \xrightarrow{3} D \xrightarrow{3} A \xrightarrow{0} A \xrightarrow{3} A \xrightarrow{3} A \xrightarrow{3} A$ $B,0 \xrightarrow{1} B \xrightarrow{1} C \xrightarrow{2} C \xrightarrow{2} C \xrightarrow{3} C \xrightarrow{3} C \xrightarrow{1} C \xrightarrow{2} C \xrightarrow{1} C$
1	$E,1 \xrightarrow{2} E \xrightarrow{1} C \xrightarrow{2} C \xrightarrow{1} C \xrightarrow{2} D \xrightarrow{3} D \xrightarrow{3} D \xrightarrow{2} E \xrightarrow{1} E$ $B,1 \xrightarrow{1} E \xrightarrow{3} E \xrightarrow{2} E \xrightarrow{1} E \xrightarrow{1} E \xrightarrow{2} D \xrightarrow{3} A \xrightarrow{3} A \xrightarrow{0} A$
2	$E,2 \xrightarrow{1} B \xrightarrow{1} B \xrightarrow{0} B \xrightarrow{0} A \xrightarrow{0} A \xrightarrow{0} A \xrightarrow{3} D \xrightarrow{2} D \xrightarrow{2} D$ $C,2 \xrightarrow{2} C \xrightarrow{1} C \xrightarrow{3} C \xrightarrow{3} C \xrightarrow{2} E \xrightarrow{1} E \xrightarrow{2} C \xrightarrow{2} C \xrightarrow{3} A$
3	$A,3 \xrightarrow{3} A \xrightarrow{0} A \xrightarrow{0} A \xrightarrow{0} B \xrightarrow{0} B \xrightarrow{1} B \xrightarrow{0} A \xrightarrow{3} D \xrightarrow{3} D$ $C,3 \xrightarrow{2} C \xrightarrow{2} E \xrightarrow{2} E \xrightarrow{3} C \xrightarrow{1} C \xrightarrow{3} C \xrightarrow{1} B \xrightarrow{0} A \xrightarrow{0} A$

5.3. Reaction class prediction results

To predict the reaction classes, we train a one-vs-rest classifier based on support vector machines (SVM) that learns a multi-class classification decision boundary. We used a randomized cross validation search strategy to perform hyperparameter tuning of the SVM model with a radial basis function. A detailed description of the SVM model and the mathematical framework that underlies it is provided in [2].

The precision and recall metrics for each of the reaction classes computed using the test-set containing unseen reactions at the training stage are shown in Figure 13 below.

From the results above, we observe that the trained model accurately predicts the reaction class for most of the reaction classes except for the reaction classes – reductions, deprotections, and heterocycle formation. The precision metrics across all 10 reaction classes shown in Figure 13(a) highlights the model's high precision in identifying the correction reaction class. However, since the recall shown in Figure 13(b) is lower for the three underperforming classes, there could be overlapping reaction classes in the feature (embedding) space. This is indeed observed for these classes in 2D visualization of the learned embeddings in Figure 12. The separation between various reaction classes could be addressed in future by also incorporating additional molecular descriptors that, in combination with the connectivity-specific embeddings, would more accurately distinguish the different reaction types. Nevertheless, the embeddings generated just based on the reactions and node-connectivity information in the hypergraph representation seems to have separated a majority of the reaction types into distinct clusters, consequently resulting in the model learning to predict them accurately. This again highlights the ability of hypergraphs to capture reaction context accurately.

6. Conclusions and future work

Network theory offers natural tools and techniques for understanding the growth of chemistry over time by representing reactions as time-evolving real-world networks. Though most of the work in this area has been done using a dyadic graph representation, a hypergraph representation with hyperedges between nodes for representing reactions is a more natural, intuitive, and flexible representation that allows for the incorporation of additional reaction context.

We have shown that the hypergraph representation is more flexible, allows for incorporation of reaction-specific node context, and facilitates one-to-one correspondence of network properties with chemistry. We have computed detailed network statistics of the resulting hypergraph network of organic chemistry and studied the time evolution of these properties. As with several previous studies, we observed that the network exhibits a scale-free behavior with preferential attachment of nodes, has small average path length indicative of small-world nature, and shows assortative mixing with respect to certain node types. For all the network statistics presented, namely, degree distributions, average path length, assortativity or degree correlations, PageRank analysis, and community detection, we have correlated them with chemistry inferences that could be drawn from such analysis. In addition, we discovered that the network exhibits the phenomenon of initial attractiveness and network densification as chemistry evolves over time time.

To demonstrate the AI-applications of the hypergraph representation of chemical reactions, we performed reaction classification using embeddings generated from chemistry-informed random walks on hyperedges. The embeddings resulted in well-separated clusters for different reaction classes and consequently accurate reaction classification results. In future, we plan to extend this study on diverse (and possibly bigger) datasets across various subdomains, incorporate additional molecular descriptors for generating hyperedge embeddings for reaction classification, utilize the results in a retrosynthetic planning framework, and perform hyperedge prediction to discover new reactions.

Conflicts of interest

There are no conflicts of interest to declare.

Author contributions

Vipul Mann: Conceptualization, Formal Analysis, Writing - Original draft, Writing - review & editing Methodology, Software; **Venkat Venkatasubramanian**: Conceptualization, Writing - review & editing, Supervision, Funding acquisition

Acknowledgements

This work was supported by the National Science Foundation (NSF) under Grant No. 2132142 and carried out at Columbia University.

References

[1] Venkatasubramanian V, Mann V. Artificial intelligence in reaction prediction and chemical synthesis. Current Opinion in Chemical Engineering. 2022;36:100749.

[2] Mann V, Brito K, Gani R, Venkatasubramanian V. Hybrid, Interpretable Machine Learning for Thermodynamic Property Estimation using Grammar2vec for Molecular Representation. Fluid Phase Equilibria. 2022;p. 113531.

[3] Alshehri AS, Tula AK, You F, Gani R. Next generation pure component property estimation models: With and without machine learning techniques. AIChE Journal. 2021;p. e17469.

[4] Mann V, Venkatasubramanian V. Predicting chemical reaction outcomes: a grammar ontology-based transformer framework. AIChE Journal. 2021;67(3):e17190.

[5] Mann V, Venkatasubramanian V. Retrosynthesis prediction using grammar-based neural machine translation: An information-theoretic approach. Computers & Chemical Engineering. 2021;155:107533.

[6] Venkatasubramanian V. The promise of artificial intelligence in chemical engineering: Is it here, finally. AIChE J. 2019;65(2):466–478.

[7] Zhang L, Mao H, Liu Q, Gani R. Chemical product design–recent advances and perspectives. Current Opinion in Chemical Engineering. 2020;27:22–34.

[8] Rangarajan S. Towards a chemistry-informed paradigm for designing molecules. Current Opinion in Chemical Engineering. 2022;35:100717.

[9] Schwaller P, Vaucher AC, Laplaza R, Bunne C, Krause A, Corminboeuf C, et al. Machine intelligence for chemical reaction space. Wiley Interdisciplinary Reviews: Computational Molecular Science. 2022;p. e1604.

[10] Albert R, Barabási AL. Statistical mechanics of complex networks. Reviews of modern physics. 2002;74(1):47.

[11] Barabási AL, Bonabeau E. Scale-free networks. Scientific american. 2003;288(5):60–69.

[12] Page L, Brin S, Motwani R, Winograd T. The PageRank citation ranking: Bringing order to the web. Stanford InfoLab; 1999.

[13] Traag VA, Waltman L, Van Eck NJ. From Louvain to Leiden: guaranteeing well-connected communities. Scientific reports. 2019;9(1):1–12.

[14] Cui P, Wang X, Pei J, Zhu W. A survey on network embedding. IEEE transactions on knowledge and data engineering. 2018;31(5):833–852.

[15] Payne J. Deep hyperedges: a framework for transductive and inductive learning on hypergraphs. arXiv preprint arXiv:191002633. 2019;.

[16] Lü L, Zhou T. Link prediction in complex networks: A survey. Physica A: statistical mechanics and its applications. 2011;390(6):1150–1170.

[17] Maurya D, Ravindran B. Hyperedge Prediction Using Tensor Eigenvalue Decomposition. Journal of the Indian Institute of Science. 2021;101(3):443–453.

[18] Fialkowski M, Bishop KJ, Chubukov VA, Campbell CJ, Grzybowski BA. Architecture and evolution of organic chemistry. Angewandte Chemie International Edition. 2005;44(44):7263–7269.

[19] Bishop KJ, Klajn R, Grzybowski BA. The core and most useful molecules in organic chemistry. Angewandte Chemie International Edition. 2006;45(32):5348–5354.

[20] Jacob PM, Lapkin A. Statistics of the network of organic chemistry. Reaction Chemistry & Engineering. 2018;3(1):102–118.

[21] Mann V, Sivaram A, Das L, Venkatasubramanian V. Robust and efficient swarm communication topologies for hostile environments. Swarm and Evolutionary Computation. 2021;62:100848.

[22] Grzybowski BA, Bishop KJ, Kowalczyk B, Wilmer CE. The'wired'universe of organic chemistry. Nature Chemistry. 2009;1(1):31–36.

[23] Gothard CM, Soh S, Gothard NA, Kowalczyk B, Wei Y, Baytekin B, et al. Rewiring chemistry: algorithmic discovery and experimental validation of one-pot reactions in the network of organic chemistry. Angewandte Chemie International Edition. 2012;51(32):7922–7927.

[24] Chodrow P, Mellor A. Annotated hypergraphs: Models and applications. Applied network science. 2020;5(1):1–25.

[25] Jin W, Coley C, Barzilay R, Jaakkola T. Predicting organic reaction outcomes with weisfeiler-lehman network. Advances in neural information processing systems. 2017;30.

[26] Lowe DM. Patent reaction extraction: downloads; 2014. Available from: https://bitbucket.org/dan2097/patent-reaction-extraction/downloads.

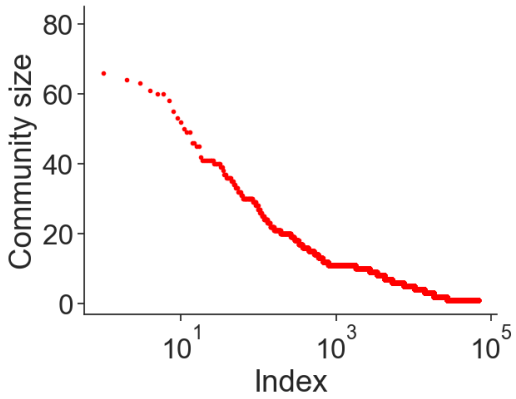
[27] Leskovec J, Kleinberg J, Faloutsos C. Graph evolution: Densification and shrinking diameters. ACM transactions on Knowledge Discovery from Data (TKDD). 2007;1(1):2–es.

[28] Probst D, Schwaller P, Reymond JL. Reaction classification and yield prediction using the differential reaction fingerprint DRFP. Digital discovery. 2022;1(2):91–97.

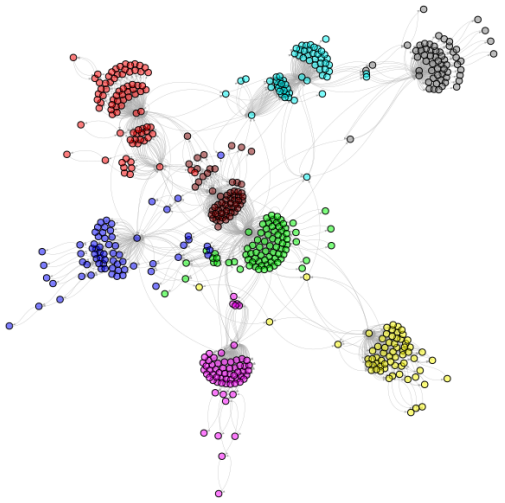
[29] Baylon JL, Cilfone NA, Gulcher JR, Chittenden TW. Enhancing retrosynthetic reaction prediction with deep learning using multiscale reaction classification. Journal of chemical information and modeling. 2019;59(2):673–688.

[30] Schneider N, Lowe DM, Sayle RA, Landrum GA. Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity. Journal of chemical information and modeling. 2015;55(1):39–53.

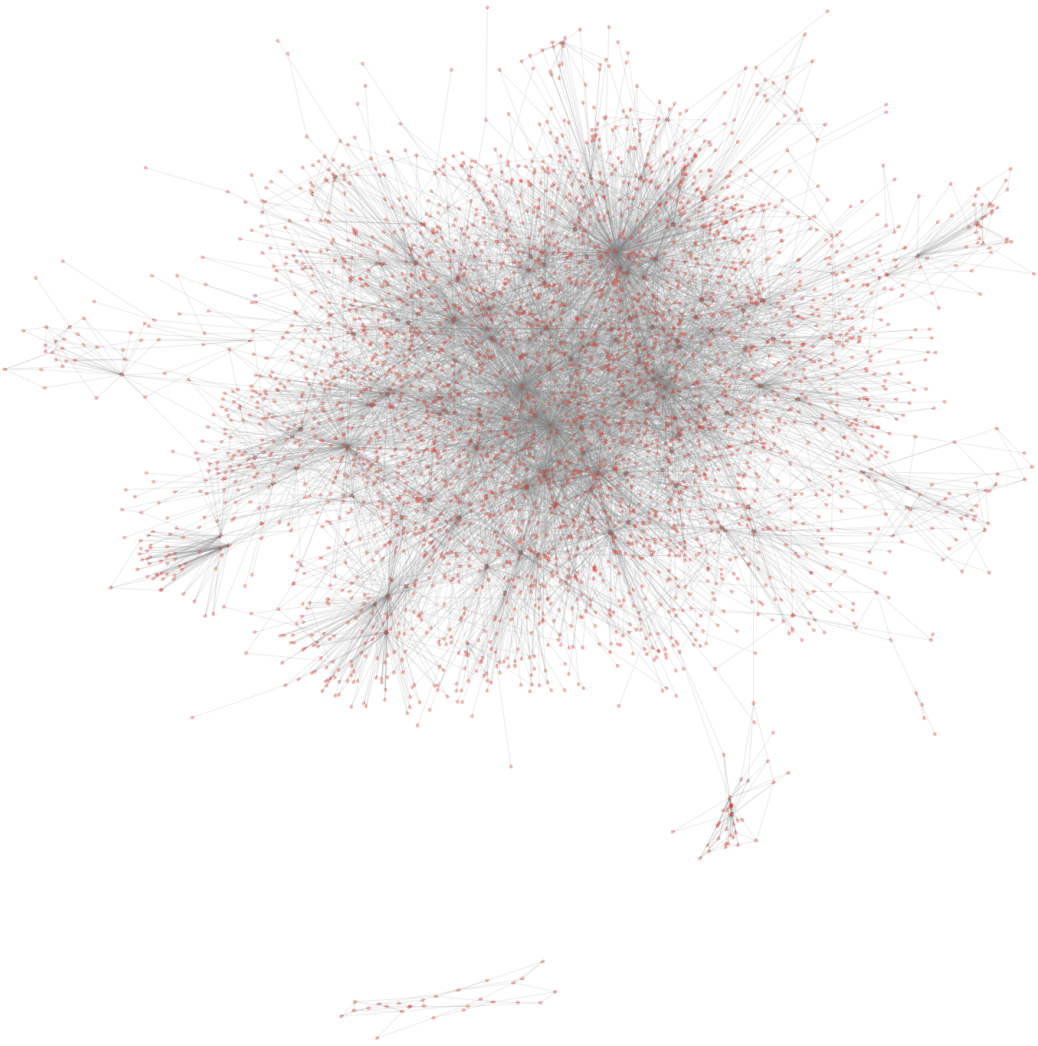
[31] Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J. Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems. 2013;26.



(a) Size distribution of identified communities



(b) Top-8 largest communities with nodes in each community in a different color



(c) Top-100 largest communities indicating showing clear regions of high and low densities along with an island community disconnected from the rest of the network

Figure 11. Community detection results on the weighted projected directed graph obtained using role-interaction kernel R2

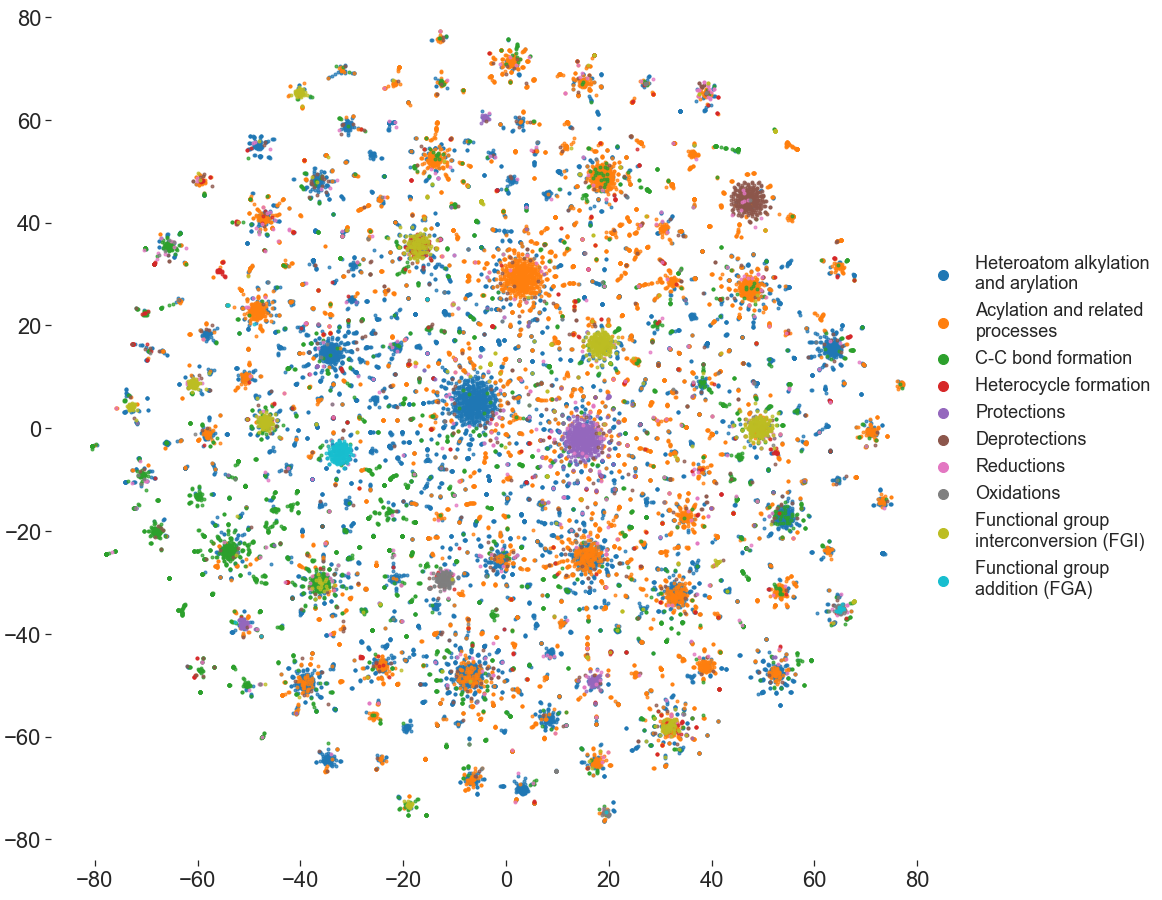
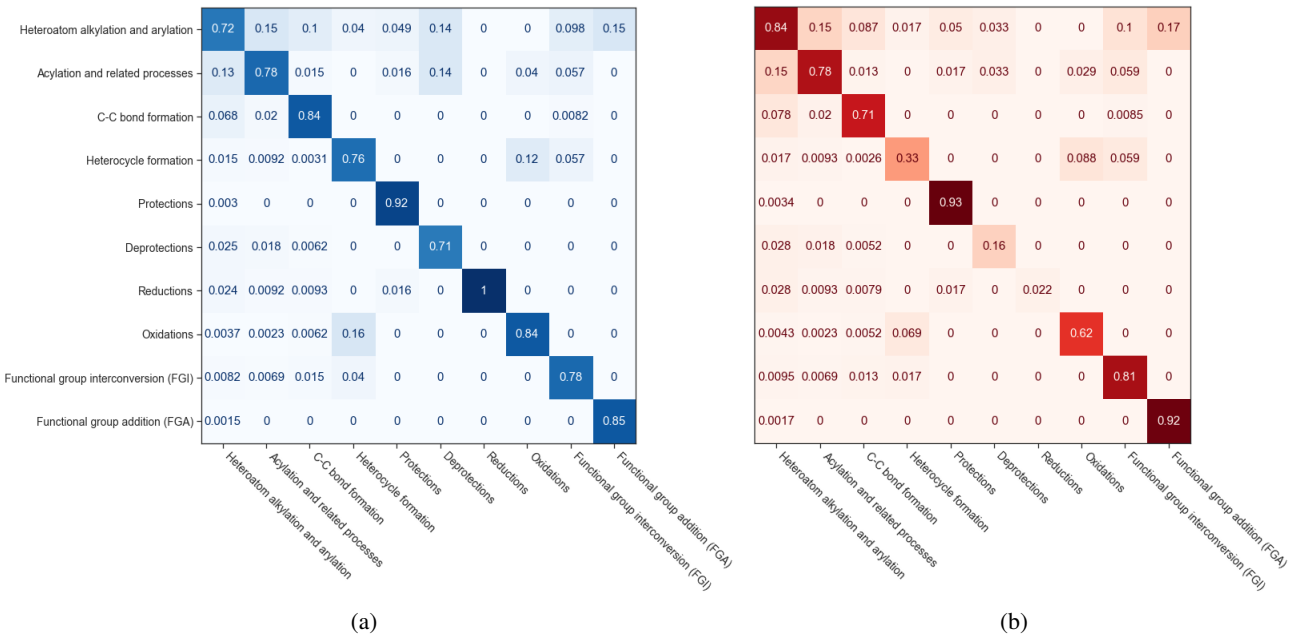


Figure 12. A 2D t-SNE projection of the 256-dimensional hyperedge embeddings



(a) (b)

Figure 13. Performance metrics for the multi-class reaction classification on the test-set (a) Precision (b) Recall