Evaluating the utility of graph similarity metrics in quantitative read-across: A tutorial

Brett Hagan[a,b], Louis Groff[b], Imran Shah[b], Grace Patlewicz*

[a]ORAU, Oak Ridge Associated Universities, Oak Ridge, TN, USA

[b]Center for Computational Toxicology and Exposure (CCTE), Office of Research and Development, US Environment Protection Agency, 109 TW Alexander Dr, Research Triangle Park, NC 27711 USA

Abstract

Read-across is a popular technique for filling data gaps for substances lacking empirical data. The approach relies on identifying source analogues with relevant data that are 'similar' to a target substance. Typically, source analogues are identified on the basis of structural similarity but the evaluation of their suitability for read-across depends on other contexts of similarity including their physical property information, chemical reactivity, bioactivity and metabolism. Whilst quantifying structural similarity is well established, relying on chemical fingerprints and using a similarity index such as Tanimoto to limit the number of analogues returned, characterizing other aspects of similarity objectively remains a challenge. The use of graph analytical approaches has become an increasingly important tool that is employed in a wide variety of domains. Since many different aspects of a substance and its associated properties lend themselves to being represented by graphs, using graph similarity offers new ways in which analogues could be identified and evaluated for read-across purposes. This manuscript considered the application of graph similarity measurements for read-across. Several different methods were discussed, which have been divided into three categories; graph kernel approaches, graph embedding approaches, and deep learning (DL) approaches. Each approach was described in brief together with a practical example to illustrate the application for read-across purposes.

Keywords: Read-Across, Graph similarity, Graph kernels, Graph convolutional networks (GCNs)

1. Introduction

1.1. Background to Read-Across

An overwhelming and ever-increasing number of substances exist in commerce, of which only a small proportion have undergone sufficient toxicological evaluation. Assessing each chemical in turn presents a significant and impractical challenge in terms of cost, animal welfare, and resources[1]. In vitro and in silico approaches have the potential to play a large role in the assessment of chemicals that lack empirical data. In silico approaches encompass (quantitative) structure activity relationships ((Q)SAR) as well as read-across (RAx), both of which relate chemical structural properties to

*Corresponding author
Email address: patlewicz.grace@epa.gov (Grace Patlewicz)

(eco)toxicological or physical property endpoints. RAx is probably the most commonly used data gap filling technique for regulatory purposes, notably it is cited as the most commonly used adaptation to address information requirements under the European Union's Registration Evaluation and Authorisation of Chemicals (REACH) regulation[2,3]. In brief, RAx describes the method for filling a data gap whereby a substance with existing data (termed the 'source analogue') is used to make a prediction of the same property for a 'target' substance with limited available empirical data. These predictions operate on the assumption that the source and target substances are 'similar' in some context with relevant information pertaining to a specific outcome[4,5]. Key to this approach is the characterization of similarity. Although structural similarity is typically the most common approach used to identify candidate source analogues, other similarity contexts namely similarity in physicochemical properties, metabolism, toxicokinetics, chemical reactivity, bioactivity and toxicological profile also play a significant role in justifying those source analogues for read-across. This is evident in the assessment frameworks that have been published such as the European Chemicals Agency's Read-Across Assessment Framework (RAAF)[6] as well as the analogue workflow underpinning the US Environmental Protection Agency's Provisional Peer Review Toxicity Values (PPRTV)[7]. For example, metabolic similarity considers the similarity of transformation pathways as determined in experimental studies or the commonality of metabolites formed. Physicochemical similarity compares certain physical property information such as the log of the octanol-water partition coefficient (logKow), melting point, boiling point etc. of source analogues relative to the target chemical to determine whether physical form and partitioning are likely to be the same. Similarity in toxicity assesses available empirical data to identify whether target organs impacted are the same and whether the potencies are comparable or follow a specific trend. Such similarity context assessments are largely qualitative and heavily reliant on expert judgement in concert with empirical data[8]. This does result in challenges in terms of reproducibility, scalability and acceptance for regulatory purposes[9]. Indeed, RAx as a technique has been in use for well over 20 years, but hesitation regarding the adoption of the approach for certain regulatory contexts (e.g. risk assessment) or within specific jurisdictions remains[10]. Thus, progress towards approaches that may increase confidence in and reduce the levels of inherent uncertainty in RAx predictions remain of vital importance in the ongoing adoption of RAx for regulatory purposes. Significant effort has been directed towards the evaluation of confidence in analogue selection in RAx across a wide range of studies[8,11,12,13,14]. Several studies have aimed to define frameworks for characterizing uncertainty[12,15,11,14], others have demonstrated how high-throughput screening data can be helpful in substantiating RAx justifications[16,14,17] by providing some evidence of mechanistic or biological similarity. The European Chemicals Agency (ECHA) have developed a read-across assessment framework to address this need[6] whereas the Organisation of Economic and Co-operative Development (OECD) have been facilitating the development of case studies with the aim of updating

existing grouping technical guidance[5] with one focus being on reducing read-across uncertainties[18].

In our own studies, Generalized Read-Across (GenRA)[9,10] was created with the goals of quantifying performance and uncertainty by establishing consistent performance baselines and quantifying the contribution of different similarity contexts play in identifying of source analogues or making toxicity predictions. Within GenRA (and RAx in general), approaches typically use structural information such as chemical fingerprints to identify candidate source analogues. Many software tools exist that facilitate this type of analogue searching, including the similarity searches within the EPA CompTox Chemicals Dashboard[19], PubChem, as well as the many functionalities within the OECD Toolbox (qsartoolbox.org)[20]. There have been a number of efforts to study additional sources of information to perform RAx, including the use of physiochemical[21], mechanistic[16], bioactivity[9,16], and metabolic data[22,23,24]. Within GenRA, research has continued to systematically evaluate the impact that different types of similarity play in read-across for the prediction of in vivo toxicity outcomes[24,25,21,26,27] along with implementing the insights learned into the GenRA (www.comptox.gov/genra) web application[10,28].

1.2. Source analogue identification

As mentioned in Section 1.1, there are a number of software tools that facilitate the identification of source analogues. Most of these use structural similarity as a basis to return analogues. This can be done in two main ways - either by a descriptor-based similarity calculation or a substructure-based assessment[29]. In practice, this means that a software tool contains a large database (or dataset) of chemical substances which serves as a source analogue inventory. To identify analogues, a search query is then performed using the target substance of interest to return candidate analogues. In a substructure-based approach, a determination of the substructures shared with the target substance are made or matched molecular pairs[30,31] are generated to identify common core structures that are distinguished at a given site. Such substructure-based calculations are binary - either the target and source analogues share a pre-defined substructure or not, therefore no adjustable threshold exists to tune the returned set of candidate analogues. On the otherhand, the hits returned are often chemical intuitive and interpretable.

In a descriptor-based approach, the key considerations are how the substances forming the source inventory are represented numerically and what metric is used to quantify a specific threshold of similarity. Source analogues can be characterized by 1D, 2D or 3D representations of structures or hybrids of these. For the aforementioned tools, 1D binary chemical fingerprints are frequently used for practical efficiency especially when a source inventory contains large numbers of substances e.g. 1 million substances. A target substance will then be converted into the same chemical fingerprint

representation and a query based on pairwise similarities will return a number of candidates based on the similarity threshold set. The similarity threshold is a quantitative measure between 0 and 1 that summarizes the commonality in structure based on the presence and absence of particular chemical fingerprints. By far, the most common similarity index that is used in the Tanimoto (Jaccard) index[32] though there are a number of other similarity indices that can also be used[32,33]. There are several types of chemical fingerprints, one of the most popular is the Morgan fingerprint[34], also known as the extended connectivity fingerprint (ECFP4).This perceives the presence of specific circular fragments around each atom in a molecule. It relies on the fragmentation of a molecule into circular fragments with each atom encoded as atom types. Other fingerprints are termed key or dictionary fingerprints where there is a defined fixed set of substructural features representing molecular characteristics. MACCS (Molecular Access System by Molecular Design Limited)[35] and ChemoType ToxPrints[36] are examples of these. Atom pairs is another example where an algortihm of atom typing is performed such that certain values for each atom of a molecule is computed[37]. In each case, the fingerprint is represented as a bit string or binary vector that can be used to search for source analogues.

Chemical fingerprints have proved useful for fast similarity comparisons as well as inputs into development of QSARs for different activity outcomes in particular toxicity endpoints. Regardless of the fingerprint type, there are always limitations in how a chemical may be represented to capture the aspects important for toxicity prediction. For example, circular fingerprints are typically poor at perceiving global features of a molecule (e.g. size or shape) and may fail to discriminate between subtle changes between 2 small molecules. In any case, chemical fingerprints represent a simplified representation of a chemical that may be insufficient to resolve differences in toxicity outcomes.

Consideration of the other similarity contexts offers a means of addressing this shortcoming although leveraging the inherent representation of a chemical structure as a graph, with atoms as nodes and bonds as edges, offers alternative opportunities to eliminate the need for precomputed features and fingerprints.

1.3. Graph Terminology

For context, a few of the pertinent terms/definitions associated with graphs are described. A graph

$$G(V, E)$$

is a visual representation of a collection of nodes (also called vertices depending on domain) and edges that connect these nodes providing a structure to represent entities and their relationships. There are three main types of edges that are typically found in a graph: undirected edges, directed edges and weighted edges.

Figure 1 provides an example of an undirected graph with 5 nodes and 5 edges and the corresponding directed graph.

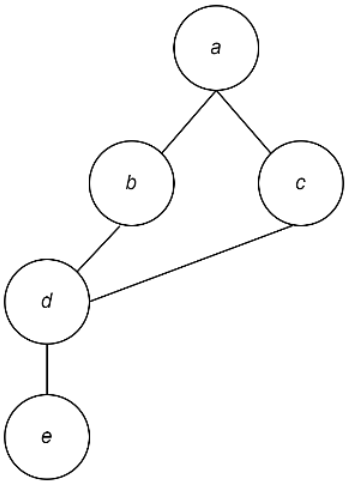

Figure 1: Undirected and directed graph with 5 nodes and 5 edges.

Undirected edges are those which identify a connection between nodes but without a given "flow". Directed edges are those where is a clear direction between nodes e.g. within a metabolic graph where a parent chemical transforms into a metabolite. Weighted edges can occur in both directed or undirected edges to depict some quantitative value e.g. the rate of disappearance of parent chemical to its corresponding metabolite.

The neighborhood of a node N(v) is the set of all nodes adjacent to v. In Figure 1, the neighborhood of node d would be expressed as N(d) = [b,c,e]. A walk comprises a sequence of edges and nodes, whereas a path is a walk with no repeating nodes visited. In Figure 2, the sequence of nodes [a,b,d] is both a walk and a path in graph G.

Two graphs are isomorphic if there is a structure that preserves a one-to-one correspondence between the nodes and edges. In other words, if the two graphs differ only by the names of the edges and nodes but are otherwise structurally equivalent, they are said to be isomorphic.

Graph analytical approaches have been used in a number of different domains including biology (to model biological systems such as protein-protein interaction networks or metabolic networks), social sciences (to study complex social networks), computer science (to model malware, botnet, and anomaly detection), and engineering (to model complex systems such as transportation networks, electrical power grids)[38].

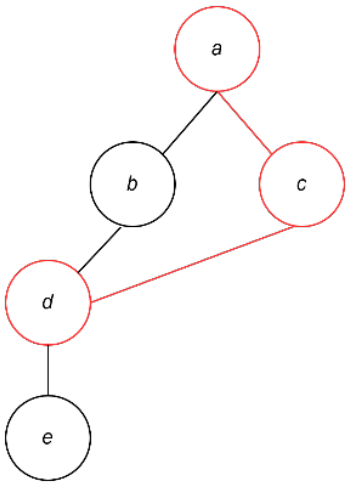

Figure 2: The sequence of nodes [a,c,d] is both a walk and a path on G since there are no repeating nodes in the sequence.

1.4. Topological indices

Considering chemicals as graphs is not a novel concept. In fact, a wide variety of chemical properties and processes have been modelled using information derived from molecular graphs. Topological indices which have been in use for more than 45 years are algebraic invariants of hydrogen depleted molecular graphs which represent the topology of a molecule. There are hundreds of topological indices but the majority can be broadly categorized into 5 main types namely: degree-based indices, distance-based indices, count-based indices, eigenvalue-based indices and information-theoretic indices[39,40]. Degree indices are based on the degree of the nodes in the graph. The Zagreb Index is based on the degrees of the nodes focusing on the sum of squares or products of node degrees whereas the Randic Index is defined by the sum of the inverse of the square roots of the degrees of adjacent nodes. The most common distance based index is the Wiener Index, which represents the sum of the shortest-path distances between all pairs of nodes in the graph. Count based indices include the Hosoya Index which counts the number of matching sets in the graph. Eigenvalue based indices include the Estrada index which is the sum of the exponential of the eigenvalues of the adjacency matrix. Finally information-theoretic indices use concepts from information theory to quantify the distribution of certain properties within the graph. The Shannon entropy measures the diversity in the distribution of node degrees. Topological indices have been widely and successfully applied to the quantitative correlation of many different molecular properties such as boiling point, chemical reactivity as well as biological activity[41]. Topological indices summarize a chemical into a single number or set of numbers. Although the indices have been used in many QSAR studies, one of their main shortcomings was a perceived lack of interpretability[42].

1.5. Graph Similarity

Graph similarity is a well-studied problem with investigations being focused upon direct operations that examine structural properties such as graph edit distance, graph isomorphism and maximum common subgraph matching approaches[43,44,45,46,47,48,49].

1.5.1. Graph edit distances using Reduced graphs

Reduced graphs provide a summarised representation of a chemical structure that are produced by collapsing connected atoms into single nodes and forming edges between the nodes in accordance with bonds in the original structure. Reduced graphs have been used in a variety of applications in chemoinformatics ranging from the representation and search of Markush structures to the identification of structure-activity relationships (SARs). There are a number of different graph reduction schemes though each has been devised to address a different purpose[50,51,52]. Graph reduction schemes have been developed for similarity searching often with the objective of identifying substances with similarity in activity. Various methods have also been developed to quantify the similarity between reduced graphs from fingerprint approaches, graph matching as well as an edit distance method. The edit distance approach quantifies the degree of similarity of 2 reduced graphs based on the number and type of operations needed to convert one graph to the other. One benefit of the edit distance method is the ability to assign different weights to different operations - useful when deriving activity specific weights as evidenced in Birchall et al.[52]. However, graph edit distance are computationally expensive unless approximation algorithms are used particularly for larger graphs. Garcia-Hernandez et al. employed graph edit distances to reduced graph representations to estimate the bioactivity of a chemical on the basis of the bioactivity of similar compounds and found better performance than the array representation-based approaches they compared against[53].

1.5.2. Graph isomorphism

Several foundational questions of chemical similarity analysis have been framed as graph comparison problems; chemical equivalence may be modeled as a graph isomorphism task. Searching for a specific substructure (e.g. a benzene ring) within a chemical has been modeled as a subgraph isomorphism task. The enumeration of possible chemical structures is closely related to graph enumeration[54]. Graph isomorphism is a test of structural equivalence, wherein two graphs are isomorphic if a structure exists that preserves a one-to-one correspondence (a bijection) between the two graphs sets of vertices and edges.

1.5.3. Maximum common subgraph

A common need in cheminformatics to the ability to align pairs of molecules together to make a determination of the degree of structural overlap. This is useful when exploring SARs, predicting

bioactivity of substances or identifying chemical reaction sites. The degree of overlap between a pair of chemicals can be achieved using maximum common subgraph isomorphism algorithms[55,56]. In cheminformatics, maximum common subgraph isomorphism is usually referred to as identifying the maximum common substructure (MCS). As put by A Dalke, "given two structures, the MCS is the largest substructure common to both". Maximum could be interpreted to imply the maximum number of atoms, number of bonds, number of cycles or even some physical property. There are also variations in how atom and bond equivalency might be defined. However the most common MCS is where all atoms are the same if the element numbers are the same and the bonds are of the same type (see http://dalkescientific.com/writings/diary/archive/2012/05/12/mcs_background.html). There are a range of algorithms that can determine the MCS between pairs of chemicals. Some algorithms and perhaps the most prevalent work on the basis of identifing cliques or maximal cliques. Examples include the Bron-Kerbosch algorithm[57] which reports all of the maximal cliques found. A clique is a set of nodes in a graph such that each node is connected to each and every other node, with a maximal clique in a graph being one that is not contained within another large clique. TOPSIM[58] is another algorithm designed to find the Maximum Common Edge Subgraph (MCES) between two graphs. The Maximum Common Edge Subgraph is similar to the Maximum Common Subgraph (MCS) problem but focuses on finding the largest subgraph that has the maximum number of edges in common between the two graphs. In this case, the algorithm converts labeled graph representations of two molecules into a compatibility graph. Then a modified maximal clique algorithm is used to find the maximal clique which represents the largest common substructure (excluding common isolated atoms) for the two molecules. A maximal common substructure is obtained by combining the largest common substructure and the common isolated atoms. The size of a maximal common substructure is then used to define both a molecular similarity index and a topological distance for two molecules. Other types of algorithms include subgraph enumeration algorithms which involve enumerating all connected subgraphs common to the two graphs that are being compared and then returning the largest subgraph. Raymond and Willett (2002) reviewed the main solutions for pairwise MCS[56] including multiple MCS[59].

More recent efforts to evaluate graph similarity include graph kernel methods, graph embedding methods, and deep learning (DL) methods (which can be considered to be a subset of graph embedding approaches in general). Graph kernel methods directly calculate a similarity score between two graphs based on their structural properties. Graph embedding methods transform graphs into numerical representations (vectors) that can be compared using standard distance metrics. Deep learning methods, a subset of graph embedding, learn these numerical representations automatically using neural networks.

1.5.4. Graph Kernels

Graph kernels were first introduced as a way to compare complex structures like graphs based on a concept from Haussler's work on kernels for discrete structures[60]. The term "graph kernels" soon emerged to describe methods specifically for comparing graphs[61,62,63]. The core idea behind graph kernels is to break down a graph into smaller components, called substructures. These substructures are then used to create feature vectors, which characterize the graph. By comparing these feature vectors, it is possible to measure how similar two graphs are. The inner products of the feature vectors can be efficiently computed to produce a similarity score between the graphs. The key to graph kernels lies in how the graph is decomposed. One simple approach is to count how many node labels are shared between graphs and computing the inner products of these label counts to produce a similarity score[64]. Figure 3 provides an example of counting node labels.

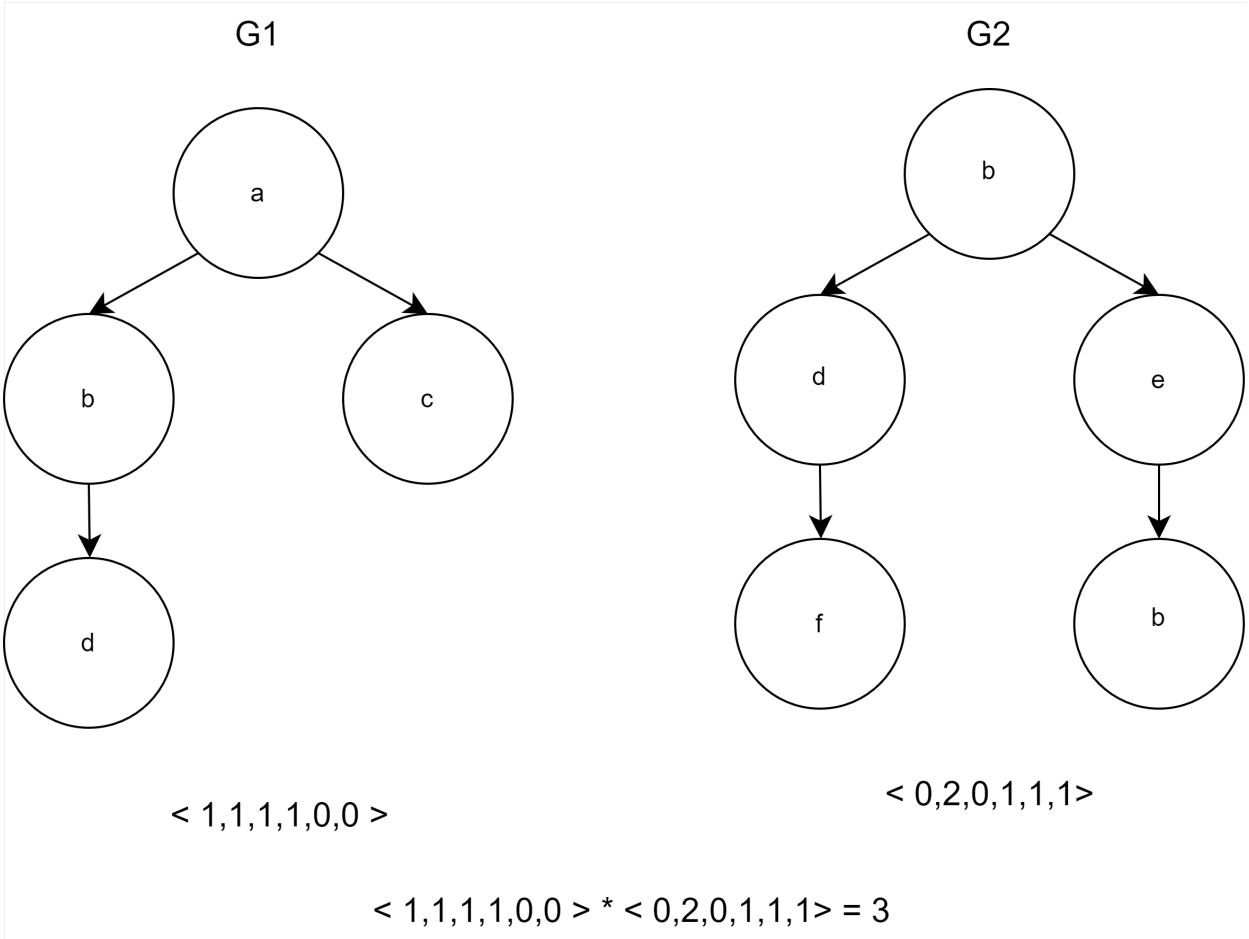

Figure 3: Graph kernel counting node labels.The feature vectors for both graphs are constructed by counting the numbers of node labels in each graph. A similarity score is then obtained by computing the inner products of the feature vectors.

There are many different ways to decompose a graph in order to compare them. One approach

is through random walk kernels. This method involves taking random paths through the graph and counting how often each path occurs in each graph[65]. Shortest path kernels aim to find the shortest paths between labeled nodes (atoms) in each graph and using these to construct feature vectors[66]. A more advanced method builds on the Weisfeiler-Lehman (WL) graph isomorphism heuristic that was introduced by Shervashidze in 2011; known as the WL subtree kernel[67]. The WL isomorphism heuristic works by iteratively updating the labels of each atom based on the labels of its neighboring atoms. Over several iterations, this process captures more detailed substructures within the molecule. If at any point the labels of the nodes in the graphs do not match, the algorithm is terminated as the two graphs can not be isomorphic. The WL subtree kernel then uses these refined labels to compare different molecular graphs. This method is particularly powerful and has been found to be closely related to operations used in graph neural networks. Figure 4 shows an example iteration of the kernel between two graphs.

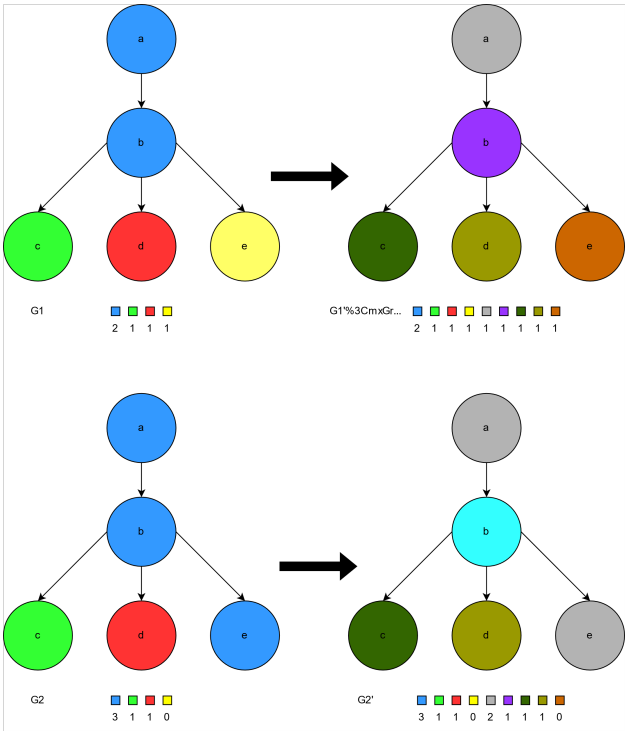

Figure 4: One iteration of the WL kernel. Feature vectors initially consist of counts of original atom labels. At each iteration, new labels (colors) are created for each atoms by considering the labels of its neighbors. Nodes a in both G1' and G2' are labeled gray as they were both adjacent to a single blue label in the previous iteration, whereas nodes e in G1' and G2' are assigned different labels due to the differences in their neighboring node labels. The feature vectors consist of counts of the original and newly created node labels as iteration continues until a defined limit or convergence is reached. The inner products are computed to obtain a similarity score.

1.5.5. Graph Embeddings

While there are numerous advantageous qualities to graph representations, the unstructured, relational nature of the data does not allow it to be directly used as input into many well established ML pipelines that require tightly structured, vectorized data[68]. To overcome this limitation, graph embedding techniques may be employed to create lower dimensional representations of graph data that retain as much topological and label (or feature) information as possible. Graph embeddings allow for a type of similarity measurement between graphs. Embedding methods represent graphs in a multi-dimensional latent space, where strongly similar graphs will lie near each other, and dissimilar graphs will lie further apart. Distance between two graphs embedding in the latent space can be considered a measure of similarity.

A number of different methods exist that are capable of creating graph embeddings which can be broadly divided into two categories: node embeddings, and whole graph embeddings. **Node embeddings** map individual nodes in a graph to numerical vectors, capturing node characteristics and relationships. **Graph embeddings** represent the entire graph as a single vector, often by combining node embeddings or using other methods, enabling comparisons between graphs.There are a variety of different approaches to either task, with well established taxonomies in literature dividing them into three distinct categories; matrix factorization methods, random walk based methods, and neural network methods, with substantial areas of overlap between the three[69,70].

Matrix factorization techniques were the earliest studied, beginning with the multi-dimensional scaling (MDS) that decomposed adjacency matrices[71]. Other factorization methods operate on graph proximity (distance matrices) or graph Laplacian matrices[72,73]. Although factorization methods are the most well-established and theoretically understood, they often scale poorly[74]. Random walk based embeddings[75] later emerged based upon word and document embedding methodologies such as Word2Vec, adopting the skip-gram neural network model used to create word embeddings to the graph context. The skip-gram model is a simple single hidden layer neural network (see Figure 5) that is trained to predict the probabilities for each word in a given vocabulary to appear near in sequence to a given target word. The network is trained, and the weights of the trained network are exploited as vectorized word embeddings, with the underlying intuition being that words that often appear in similar contexts are likely highly similar in some context[76].

DeepWalk adapted the SkipGram approach to a graph setting[75] for node embedding. Words are analogous to nodes in the graph, the sequences of words (a "context") are analogous to random walks across node neighborhoods, and the vocabulary of words is analogous to all nodes in the graph. Node2Vec iterated upon DeepWalk with the introduction of parameters to control the length and freedom of

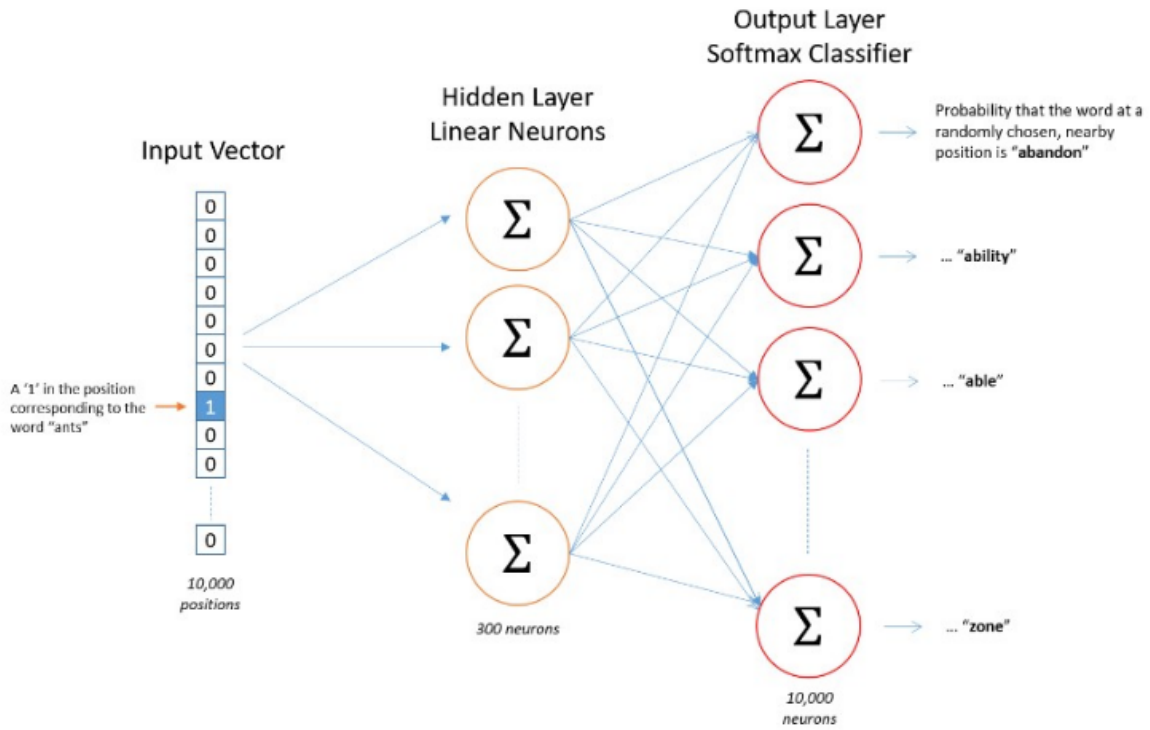

Figure 5: Skip gram model for Word2Vec word embeddings. A one hidden layer neural network is trained to determine the probabilities for each word in a vocabulary of appearing near in sequence to a given target word. The target word is given as a one-hot encoded input, and after training via backpropagation over a number of epochs, the hidden weights of the network are used as embedded vector representations of words.

the random walk operations[77]. Graph2Vec iterated upon Node2Vec to allow for skip-gram based whole graph embeddings based off rooted subgraphs analogous to words in Word2Vec[78]. GL2Vec improved upon Graph2Vec in classification tasks by incorporating information gleaned from a line graph representation, better allowing for the capture of structural information[79]. Research has shown that more complicated approaches to graph embeddings may not necessarily result in better performance. The LDP (Local Degree Profile) embedding method was introduced in 2019 and showed comparable performance to more sophisticated embeddings methods while only considering the degree information of nodes in a graphs without considering any label information whatsoever[80].

1.5.6. Deep Learning Embeddings

Inspired by the widespread success of various deep learning approaches such as convolutional neural networks (CNNs), graph neural networks (GNNs) were introduced in 2009 with the goal of extending existing neural network models for processing graph structured data[81]. Soon after, graph convolutional networks (GCNs) were defined that took inspiration from the concepts of convolutional operations on structured data for image processing tasks.

GCNs take a graph as input and pass it through a number of convolutional layers that aggregate each nodes neighborhood information. At each training epoch, each node in the graph has its hidden state updated by aggregating each of the node's neighbors hidden states together by some function, and combining it with the current hidden state of the node. The output of convolutional layers is a set of node embeddings, vectorized representations of each node in the graph. Whole graph embeddings can be generated from these individual node embeddings by combining them in some way through a "pooling" layer that aggregates the node embeddings together.

In Hagan et al. in prep, this model was used to perform genotoxicity classification on the dataset described in Section 1.5.5 by using GCNs to create whole graph embeddings of metabolic graphs. Metabolic graph representations were created using simulated data generated by metabolism simulation tools TIssue MEtabolism Simulator[82] and BioTransformer[83], where nodes in a graph represented the original chemical and its metabolites, and directed edges between nodes represented a transformation to a metabolite. A number of GCN architectures were employed to classify the metabolic graphs as either genotoxic or non-genotoxic. After training, a held-back validation set of metabolic graphs were embedded by the network. These embeddings were used as input into several classification models and compared against a baseline performance established by the use of Morgan chemical fingerprints into the GenRA package k-nn classifier, with improvements as high as over 7% in AUC-ROC score for genotoxicity classification of the chemicals. Readers are referred to Hagan et al. for further details.

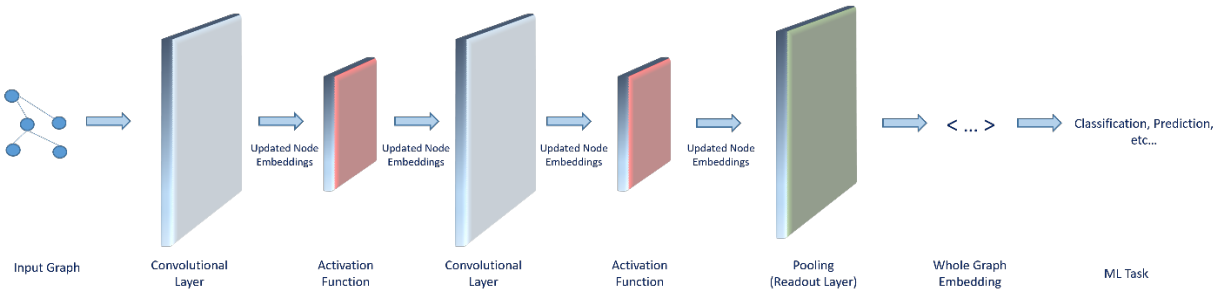

Figure 6: Graph convolutional network conceptual model. A graph is given as input into the model and passed through a series of convolutional layers and activation functions that produce embeddings for each node in the graph. The individual node embeddings are aggregated together by some pooling operation in a readout layer in order to produce a whole graph embedding as output that can be used for structured ML tasks such as classification, regression, or prediction.

Herein, for each family of methods, the approach is described conceptually, and a short, simple example provided to demonstrate how these approach could be practically applied for read-across purposes either for the identification of analogue or to perform an endpoint prediction.

2. Methods

2.1. WL Subtree Kernel Example

A read-across example, comprising target substance 2-Amino-4,6-dinitrotoluene (2-ADNT) (CASRN 35572-78-2) and its structural analogues, was identified from the published EPA Provisional Peer-Reviewed Toxicity Values (PPRTV) assessments. A PPRTV is defined as a toxicity value derived for use in the EPA Superfund Program. PPRTVs are derived after a review of the relevant scientific literature using established EPA Agency guidance on human health toxicity value derivations. The objective is to provide support for the hazard and dose-response assessment pertaining to chronic and subchronic exposures of substances of concern, to present the major conclusions reached in the hazard identification and derivation of the PPRTVs, and to characterize the overall confidence in these conclusions and toxicity values. Current assessments can be accessed on the U.S. Environmental Protection Agency's (EPA's) PPRTV website at https://www.epa.gov/pprtv. In cases where there is a paucity of data to derive a PPRTV for a specific substance, an analogue approach is applied which permits the use of data from related substances to calculate a screening value. The exact procedure is described in more detail in Wang et al[7].

Five structural analogues with relevant oral non cancer toxicity values were identified for the target substance 2-ADNT (see Table 1). Structures were represented as SMILES. WL subtree kernels were derived for the set of candidate analogues to compare their pairwise similarities. Morgan chemical fingerprints (radius = 3, bitvector = 1024) were also derived from which a Jaccard index was calculated. This would provide a similarity metric typically relied upon in analogue searching tools already

described. Molecular graph representations were created using the Python package RDKit[84]. The open source Python package GraKeL[85] was used to implement the WL subtree kernel.

2.2. Node Embedding Example

A dataset of 82 read-across case examples compiled from the literature taken from Patlewicz et al, in preparation was used to explore the uility of Node2Vec embeddings in comparing analogue pairs. The 82 cases comprised 468 substances which were converted to graph objects as described in Section 2.1. The Node2Vec python library was used to learn node embeddings by creating biased random walks of the molecular graphs. The embedding dimensions were set to 64, with a random walk length of 30 and parameters to adjust the balance between structural equivalence and homophily.

2.3. Graph Embedding Example

To demonstrate the applicability of graph embedding methodologies within a RAx context, an example was developed using a dataset of substances with associated genotoxicity outcomes. The dataset was an updated version of that compiled in Pradeep et al[86] drawn from the EPA Toxicity Values database (ToxValDB). The same methodology as described in Pradeep et al[86] was used to create a dataset with a summary genotoxicity outcome for each chemical. Genotoxicity studies, including in vitro and in vivo chromosomal aberration, Ames, micronucleus, mouse lymphoma studies were initially retrieved from ToxValDB. To create a single outcome per chemical, the dataset was first grouped by substance identifier and summarized as follows: if a substance was associated with a positive Ames result, a positive genotoxicity outcome was returned, if a substance was not associated with a positive Ames but did have a reported positive chromosomal or micronucleus outcome, it was tagged as a clastogen. If only inconclusive studies were associated with a substance, an inconclusive tag was assigned, finally if only negative outcomes were associated with the substance, a non-genotoxicity outcome was returned. For the dataset compiled with structural information, there were 5403 chemicals with QSAR-READY SMILES and a genotoxicity outcome.

Genotoxicity is an endpoint of particular interest where many different prediction models have been developed from quantitative structure activity relationships (QSARs) to read-across approaches. Examples of models include the Ames mutagenicity model that exists within the EPA TEST suite (see [https://www.epa.gov/comptox-tools/toxicity-estimation-software-tool-test] as well as a myriad of genotoxicity models available within the VEGA suite of tools (see [https://www.vegahub.eu/portfolio-item/vega-qsar/]. Benigni reviewed the state of the art of modelling genotoxicity[87] discussing the different approaches that have been applied to which test guidelines have been modelled. There has been a renewed interest in building new models for genotoxicity since the International Conference on Harmonization (ICH) M7 guideline permited the use of in silico approaches for predicting Ames

mutagenicity for the initial assessment of impurities in pharmaceuticals. The guideline allows for a knowledge base and statistical model to be used in combination to predict Ames mutagenicity. Two modelling challenges were established recently to crowdsource the development of new models to predict the Ames mutagenicity, first of which was reported in Honma et al[88] with a followup study described in Furuhama et al.[89].

Beyond QSAR approaches, read-across approaches have been also been applied to genotoxicity as discussed by Benigni[90]. One quantitative read-across approach includes the method employed by GenRA[9], wherein structural aspects of chemicals are used in a K nearest-neighbors (k-nn) classification to derive a similarity weighted activity outcome. Morgan chemical fingerprints form one possible set of structural features to identify similar substances to perform a read-across. An alternative approach is to employ graph representations and embedding methods such as Graph2Vec, GL2Vec, and LDP to characterize substances and identify analogues. Herein, the constructed dataset was used to perform a genotoxicity prediction where the three aforementioned methods were used to create molecular graph representations as a basis to identify similar analogues.

Molecular graph representations were generated using the open source Python package RDKit as described in Section 2.1. Graph2Vec, GL2Vec, and LDP embedding models, implemented within the Python package KarateClub[91], were used to generate vectorized embeddings for each substance. The embeddings were projected in 2D using a t-distributed stochastic neighborhood embedding (t-SNE)[92], which was color coded by genotoxicity outcome. The embeddings were used as inputs in 2 classifiers; a k-nn classifier and logistic regression to assess their informative content. As a baseline comparator, Morgan chemical fingerprints were used as feature inputs into the same two classifiers. The 2 classifiers were implemented using the open source Python package scikit-learn[93] with the area under the curve-receiver operating characteristic (AUC-ROC) as a performance metric.

2.4. GCN Embeddings Example

The same dataset as described Section 2.3 was used to demonstrate the applicability of the GCN embedding method within a RAx context. An end-to-end GCN supervised graph classification model was constructed by using three convolutional layers (GATv2Conv convolutional layer, a graph attentional layer from Brody et al.,[94]) with ReLU activation functions, a global mean pooling readout layer, and a single fully connected linear layer to make predictions. For the molecular graphs, one hot encodings of the atom symbol labels were attached as node feature vectors. The graphs were split into a training and validation set of size 4,000 and 1,403 respectively. Using cross entropy loss and an Adam optimizer with a learning rate of 0.001, the model was trained over 50 epochs, with the AUC score of the training and validation graphs reported at each epoch. After training, embeddings for the

validation graphs were generated by inputting the graphs into the trained model and extracting the resultant embedding from the readout layer. These were visualized via t-SNE and labeled by outcome. The embeddings were also used as inputs into k-nn and logistic regression classification models, with performance compared against the use of Morgan chemical fingerprints.

3. Results

3.1. WL Subtree Kernel

Based on an expert-driven evaluation of the structural, physicochemical, available toxicokinetic (TK) data, and toxicity data, 2,4,6-Trinitrotoluene (TNT) was chosen as the 'best analogue' primarily based on its metabolic similarity, structural similarity, and shared metabolites. The similarity of toxicological outcomes across all the source analogues established confidence in the toxicologic read-across for 2-ADNT. TNT was also determined to be the most health-protective analogue because its point of departure (POD) and corresponding reference dose (RfD) value were lower than the other candidate analogues. WL and Jaccard (based on Morgan fingerprints) pairwise similarities across the target and all analogues are shown in Figure 7. TNT had both the highest WL and Jaccard score. The Jaccard similarities based on Morgan fingerprints were notably lower from the WL scores though the ranking in terms of the similarities relative to the target chemical was largely comparable. The difference between nitro group vs. amino group accounted for the slight decrease in WL score from the target 2-ADNT whereas the position of the methyl group appeared not to impact the score. The remaining candidate analogues all had lower WL scores owing to the change of substituent position as well as the substituents themselves. The high WL scores are likely to be as a result of the manner in which the graph was initially constructed using only atom symbols as labels. The absence of other relevant atom properties might better discriminate the differences between the analogues. Substances with similar overall topology but different substituents are likely to yield high WL scores since the WL kernel is sensitive to the global structure of the chemical and may overemphasize this at the expense of distinct local features. To explore this further the manner in which the graphs were constructed (see related code) was refined to incorporate additional atom property information and the WL scores were re-computed.

Table 1 compares the refined WL scores with the original WL based on atom labels alone and the Jaccard metric. Whilst the naive WL scores gave rise to the highest scores, incorporating additional atom property information (including aromaticity, hybridization and atom degree) refines the score (WL-rev) so that the differences between the substituents and their positions are better accounted for. The WL scores offer an effective means of computing a similar score direct from the structural representation bypassing the need to compute separate chemical fingerprints or descriptors. The

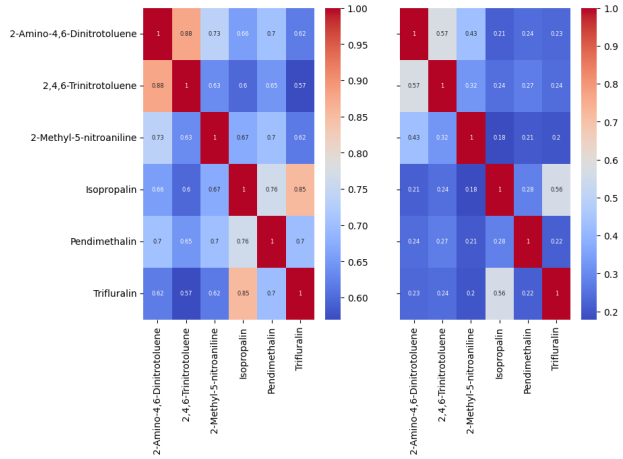

Figure 7: Pairwise similarities based on WL and Jaccard Morgan Fingerprints

approach was sensitive to the manner in which the graphs were constructed and care needs to be placed on incorporating atom and bond information so that small local differences between substances such as substituent differences on an aromatic ring structure are not lost in the manner in which global structure is characterized. Careful consideration of the type of information to attribute through label selection is an important aspect of this graph kernel operation.

Table 1: 2-ADNT is denoted as the target substance based on its role designation. TNT was ultimately selected as the read-across candidate out of the 5 candidate analogues. WL, Jaccard and WL-rev denote the similarity scores computed. WL relies on molecular graphs constructed using only atoms as labels whereas WL-rev scores were derived taking into account other atom property information. The pairwise scores are shown in each case. e.g. TNT was determined to have a Jaccard similarity with 2-ADNT of 0.57 whereas the WL score and revised WL score was 0.88 and 0.73 respectively.

Substance	Role	DTXSID	WL	Jaccard Morgan FP	WL-rev
2-ADNT	Target	DTXSID6044068	1	1	1
TNT	Selected	DTXSID7024372	0.88	0.57	0.73
2-Methyl-5-nitroaniline	Candidate	DTXSID4020959	0.73	0.43	0.52
Isopropalin	Candidate	DTXSID8024157	0.66	0.21	0.45
Pendimethalin	Candidate	DTXSID7024245	0.70	0.24	0.50
Trifluralin	Candidate	DTXSID4021395	0.62	0.23	0.42

3.2. Node Embedding

The cosine distance for selected read-across examples (see Table 2) were calculated on the basis of the node embeddings. On first inspection, the node embeddings appear promising given the low cosine distances - the read-across candidates were found to be similar to their respective targets. However

the maximum cosine distance across the overall dataset was determined to be quite low (0.44). Indeed, the pairwise cosine distances based on the embeddings for two random chemicals extracted from the dataset, chlorobenzene and caffeine was computed to be 0.19 which is not that much higher than for their respective candidate analogues. A permutation test between all the individual read-across cases relative to the overall dataset found that 67% of case examples had maximum pairwise distances that were not significantly different. Overall the node embeddings produced did not appear to be able to discriminate read-across case substances that were considered to be particularly similar in terms of their structure relative to the entire dataset suggesting other embeddings that are able to capture the whole graph might be more promising.

Table 2: Selected targets and their candidate analogues and their corresponding pairwise cosine scores.

Substance	Role	DTXSID	Cosine Distance
Chlorobenzene	Target	DTXSID4020298	0
1,4-Dichlorobenzene	Candidate	DTXSID1020431	0.12
1,2-Dichlorobenzene	Candidate	DTXSID6020430	0.14
Caffeine	Target	DTXSID0020232	0
Theophylline	Candidate	DTXSID5021336	0.14
Theobromine	Candidate	DTXSID9026132	0.11

3.3. Graph Embedding

Figure 8, Figure 9 and Figure 10 show the embeddings for Graph2Vec, GL2Vec and LDP as projected into t-SNE plot and color coded by genotoxicity outcome. The mean AUC-ROC scores from 5-fold cross validation classifiers are shown in Table 3.

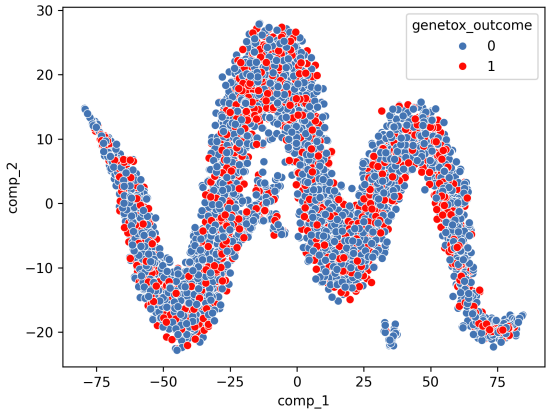

Figure 8: Graph2Vec embeddings labeled by genotoxicity outcome

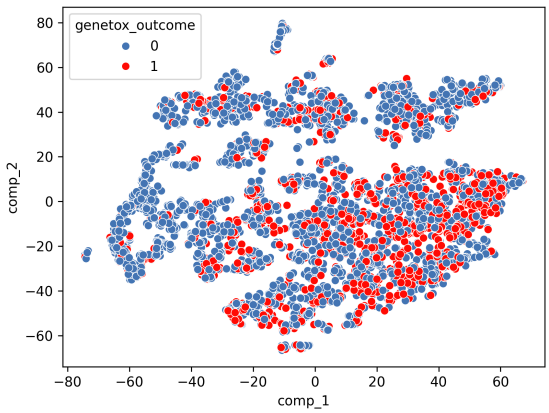

Figure 9: GL2Vec embeddings labeled by genotoxicity outcome

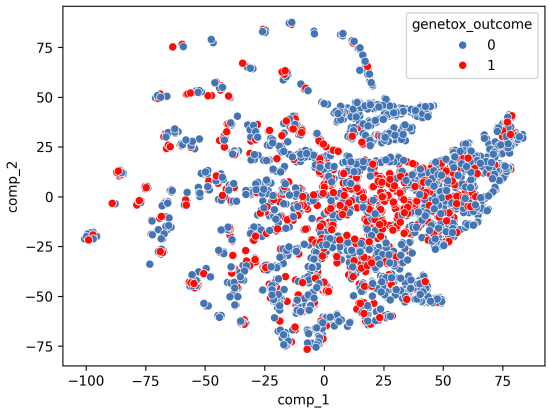

Figure 10: LDP embeddings labeled by genotoxicity outcome

Table 3: 5-fold cross validated k-nn and logistic regression genotoxicity classification results using Morgan fingerprints and the three embeddings methods .

Embedding Method	K-nn	Logistic Regression
Morgan FPs	0.662	0.727
Graph2Vec	0.526	0.500
GL2Vec	0.604	0.598
LDP	0.596	0.549

The quality of the embeddings generated by Graph2Vec, GL2Vec, or LDP failed to capture relevant chemical features effectively to be able to discriminate between genotoxic and non-genotoxic

outcomes. Morgan chemical fingerprints outperformed the graph embeddings using both classifiers. Graph2Vec struggled to separate the data, with almost no discrimination between the two outcomes as shown in Figure 8. GL2Vec and LDP both provided better discrimination, with clearer clustering of genotoxic and non-genotoxic chemicals (Figure 9 and Figure 10). Fine tuning parameters such as embedding length and learning rates may increase performance since all embeddings were generated using the default parameters of the models. Default parameters were also used for the classification models, leaving another area of possible improvement. The graph representations used were also very simple, using only the atom symbol as the node labels. Different types of labels may lead to better performance in embedding methods that rely on node label information alone.

3.4. GCN Embeddings

GCN embeddings were visualized via t-SNE and labeled by outcome as shown in Figure 11. The 5-fold cross validation AUC scores for the K-nn and Logistic regression using the GCN embeddings were found to be 0.66 and 0.78 respectively, a comparable performance to Morgan fingerprints using a K-nn approach but a marked improved with the logistic regression.

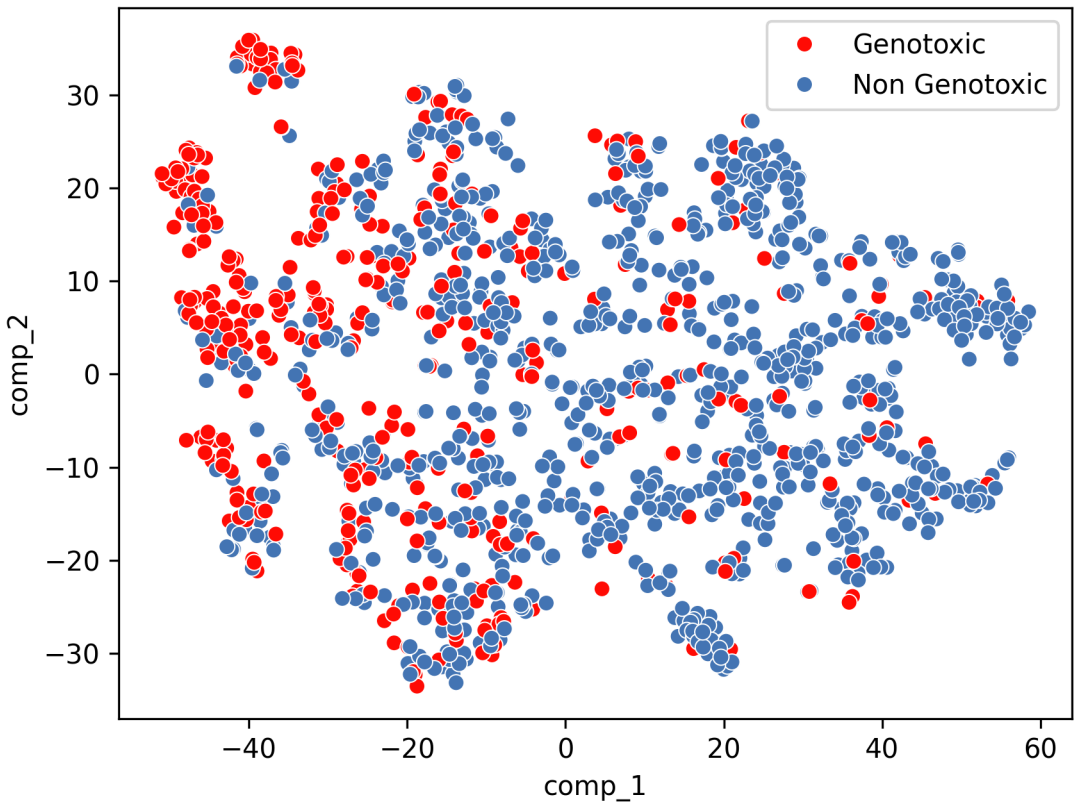

Figure 11: GCN embeddings of validation graph set labeled by genotoxicity outcome

A clearer separation between genotoxic and non-genotoxic chemicals in the embedded space created by the supervised classification GCN model was observed (Figure 11) with an improvement in both the K-nn and logistic regression classification performance relative to using Morgan fingerprints. As with the previous graph embedding discussed in Section 2.3, default parameters were used for both classification models, likely leaving room for improvement in performance through hyperparameter tuning. As with all DL models, there are a large number of options available when constructing a GCN architecture. Layer types, selection of activation functions, pooling methods, choice of loss functions and optimizers, as well as the fine tuning of parameters such as the optimizer's learning rate, the number of training epochs and number of neurons per layer are all of significant importance in a network's performance. Further experimentation with network architecture would likely lead to better performance, but for the purposes of this illustrative example, the application of a generically designed network without any fine tuning was still able to yield reasonable performance.

4. Conclusion

In this tutorial review, a selection of approaches to quantifying graph similarity were described and demonstrated for their role in identifying and evaluating analogues within a read-across approach. A WL graph kernel approach was found to be useful in characterizing potential analogues relative to 2-ADNT, identifying TNT as the most similar analogue. TNT was selected as the source analogue for use in the read-across assessment. The WL scores were found to be sensitive to the way in which the graphs were initially constructed such that if atom and bond characteristics were not sufficiently captured, local differences in structural representations could be underrepresented relative to the whole molecular effects and thereby overinflating the resulting scores. Careful attention is needed to capture node and edge information before their use. Both the WL with atom labels or a revised WL taking into account additional atom properties identified TNT as the most similar analogue to 2-ADNT, their relative ranking of analogues was the same even though the actual WL scores differed. The Jaccard scores using Morgan fingerprints were lower no doubt highlighting that small changes in substituents and their positions are not well discriminated across the analogues relative to the target substance. Topological and label information played a significant role in ascertaining the WL similarities.

In contrast embedding approaches building on the Word2Vec approach were found to be poor at capturing relevant molecular information and were ineffective in discriminating between substances that were read-across candidates (using Node2Vec to learn embeddings) or as in the second example were categorized as genotoxic or not. In the latter example, Morgan fingerprints were found to be superior in predicting the genotoxicity outcomes. Graph2Vec and LDP performed particularly poorly whereas GL2Vec was slightly better at discriminating genotoxicity or not using the 2 classifiers.

A Deep learning GCN model fared better, with a marked improvement in performance compared with Morgan fingerprints. Whereas the embedding approaches applied in Section 1.5.5 were unsupervised in nature, the GCN required labeled training data to create informed embeddings to facilitate genotoxicity classification. This performance increase observed also came at a cost of resources, and complexity. The GCN approach can be computationally expensive, depending on model parameters, scale of datasets, size of graphs and graph features, and more. These examples illustrate the potential that graph similarity approaches could play significant roles in the identification of suitable analogues for RAx However careful attention needs to be paid to the embedding representation applied and how the initial graphs are constructed.

Disclaimer

This manuscript reflects the opinions of the authors and are not reflective or the opinions or policies of the US EPA.

References

[1] NRC, Toxicity Testing: Strategies to Determine Needs and Priorities., National Academies (1984).

[2] E. Commission, Regulation (EC) No 1907/2006 of the European Parliament and of the Council of 18 December 2006 concerning the Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH), establishing a European Chemicals Agency, amending Directive 1999/45/EC and repealing Council Regulation (EEC) No 793/93 and Commission Regulation (EC) No 1488/94 as well as Council Directive 76/769/EEC and Commission Directives 91/155/EEC, 93/67/EEC, 93/105/EC and 2000/21/EC, legislative Body: CONSIL, EP (Dec. 2006).
URL http://data.europa.eu/eli/reg/2006/1907/oj/eng

[3] D. S. Macmillan, A. Bergqvist, E. Burgess-Allen, I. Callan, J. Dawick, B. Carrick, G. Ellis, R. Ferro, K. Goyak, C. Smulders, R. A. Stackhouse, E. Troyano, C. Westmoreland, B. S. Ramón, V. Rocha, X. Zhang, The last resort requirement under REACH: From principle to practice, Regulatory Toxicology and Pharmacology 147 (2024) 105557. doi:10.1016/j.yrtph.2023.105557.
URL https://www.sciencedirect.com/science/article/pii/S0273230023002258

[4] S. j. Enoch, Chemical Category Formation and Read-Across for the Prediction of Toxicity, in: T. Puzyn, J. Leszczynski, M. T. Cronin (Eds.), Recent Advances in QSAR Studies: Methods and Applications, Springer Netherlands, Dordrecht, 2010, pp. 209–219. doi:10.1007/978-1-4020-9783-6_7.
URL https://doi.org/10.1007/978-1-4020-9783-6_7

[5] OECD, Guidance on Grouping of Chemicals, Second Edition | en | OECD (2014).
URL https://www.oecd.org/publications/guidance-on-grouping-of-chemicals-second-edition-9789264274679-en.htm

[6] E. C. Agency, Read-Across Assessment Framework (RAAF)., Publications Office, 2017.
URL https://data.europa.eu/doi/10.2823/619212

[7] N. C. Y. Wang, Q. Jay Zhao, S. C. Wesselkamper, J. C. Lambert, D. Petersen, J. K. Hess-Wilson, Application of computational toxicological approaches in human health risk assessment. I. A tiered surrogate approach, Regulatory Toxicology and Pharmacology 63 (1) (2012) 10–19, number: 1. doi:10.1016/j.yrtph.2012.02.006.
URL https://linkinghub.elsevier.com/retrieve/pii/S0273230012000323

[8] G. Patlewicz, N. Ball, P. J. Boogaard, R. A. Becker, B. Hubesch, Building scientific confidence in the development and evaluation of read-across, Regulatory toxicology and pharmacology: RTP 72 (1) (2015) 117–133, number: 1. doi:10.1016/j.yrtph.2015.03.015.

[9] I. Shah, J. Liu, R. S. Judson, R. S. Thomas, G. Patlewicz, Systematically evaluating read-across prediction and performance using a local validity approach characterized by chemical structure and bioactivity information, Regulatory toxicology and pharmacology: RTP 79 (2016) 12–24. doi:10.1016/j.yrtph.2016.05.008.

[10] G. Patlewicz, I. Shah, Towards systematic read-across using Generalised Read-Across (GenRA), Computational Toxicology 25 (2023) 100258. doi:10.1016/j.comtox.2022.100258.
URL https://www.sciencedirect.com/science/article/pii/S2468111322000469

[11] K. Blackburn, S. B. Stuard, A framework to facilitate consistent characterization of read across uncertainty, Regulatory toxicology and pharmacology: RTP 68 (3) (2014) 353–362, number: 3. doi:10.1016/j.yrtph.2014.01.004.

[12] T. W. Schultz, A.-N. Richarz, M. T. D. Cronin, Assessing uncertainty in read-across: Questions to evaluate toxicity predictions based on knowledge gained from case studies, Computational Toxicology 9 (2019) 1–11. doi:10.1016/j.comtox.

2018.10.003.
URL https://www.sciencedirect.com/science/article/pii/S2468111318300811

[13] S. Wu, K. Blackburn, J. Amburgey, J. Jaworska, T. Federle, A framework for using structural, reactivity, metabolic and physicochemical similarity to evaluate the suitability of analogs for SAR-based toxicological assessments, Regulatory toxicology and pharmacology: RTP 56 (1) (2010) 67–81. doi:10.1016/j.yrtph.2009.09.006.

[14] G. Patlewicz, M. T. Cronin, G. Helman, J. C. Lambert, L. E. Lizarraga, I. Shah, Navigating through the minefield of read-across frameworks: A commentary perspective, Computational Toxicology 6 (2018) 39–54. doi:10.1016/j.comtox.2018.04.002.
URL https://linkinghub.elsevier.com/retrieve/pii/S2468111318300331

[15] T. W. Schultz, P. Amcoff, E. Berggren, F. Gautier, M. Klaric, D. J. Knight, C. Mahony, M. Schwarz, A. White, M. T. D. Cronin, A strategy for structuring and reporting a read-across prediction of toxicity, Regulatory toxicology and pharmacology: RTP 72 (3) (2015) 586–601, number: 3. doi:10.1016/j.yrtph.2015.05.016.

[16] S. E. Escher, H. Kamp, S. H. Bennekou, A. Bitsch, C. Fisher, R. Graepel, J. G. Hengstler, M. Herzler, D. Knight, M. Leist, U. Norinder, G. Ouédraogo, M. Pastor, S. Stuard, A. White, B. Zdrazil, B. van de Water, D. Kroese, Towards grouping concepts based on new approach methodologies in chemical hazard assessment: the read-across approach of the EU-ToxRisk project, Archives of Toxicology 93 (12) (2019) 3643–3667, number: 12. doi:10.1007/s00204-019-02591-7.
URL http://link.springer.com/10.1007/s00204-019-02591-7

[17] C. Rovida, S. E. Escher, M. Herzler, S. H. Bennekou, H. Kamp, D. E. Kroese, L. Maslankiewicz, M. J. Moné, G. Patlewicz, N. Sipes, L. v. Aerts, A. White, T. Yamada, B. v. d. Water, NAM-supported read-across: From case studies to regulatory guidance in safety assessment, ALTEX - Alternatives to animal experimentation 38 (1) (2021) 140–150, number: 1. doi:10.14573/altex.2010062.
URL https://www.altex.org/index.php/altex/article/view/2140

[18] Integrated Approaches to Testing and Assessment (IATA) - OECD.
URL https://www.oecd.org/chemicalsafety/risk-assessment/iata/

[19] A. J. Williams, C. M. Grulke, J. Edwards, A. D. McEachran, K. Mansouri, N. C. Baker, G. Patlewicz, I. Shah, J. F. Wambaugh, R. S. Judson, A. M. Richard, The CompTox Chemistry Dashboard: a community data resource for environmental chemistry, Journal of Cheminformatics 9 (1) (2017) 61. doi:10.1186/s13321-017-0247-6.

[20] T. W. Schultz, R. Diderich, C. D. Kuseva, O. G. Mekenyan, The OECD QSAR Toolbox Starts Its Second Decade, Methods in Molecular Biology (Clifton, N.J.) 1800 (2018) 55–77. doi:10.1007/978-1-4939-7899-1_2.

[21] G. Helman, I. Shah, G. Patlewicz, Extending the Generalised Read-Across approach (GenRA): A systematic analysis of the impact of physicochemical property information on read-across performance, Computational toxicology (Amsterdam, Netherlands) 8 (2018) 34–50. doi:10.1016/j.comtox.2018.07.001.
URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6820193/

[22] C. Lester, E. Byrd, M. Shobair, G. Yan, Quantifying Analogue Suitability for SAR-Based Read-Across Toxicological Assessment, Chemical Research in Toxicology 36 (2) (2023) 230–242. doi:10.1021/acs.chemrestox.2c00311.

[23] D. Gadaleta, A. Golbamaki Bakhtyari, G. J. Lavado, A. Roncaglioni, E. Benfenati, Automated integration of structural, biological and metabolic similarities to improve read-across, ALTEX 37 (3) (2020) 469–481. doi:10.14573/altex.2002281.

[24] G. Patlewicz, P. Karamertzanis, K. Paul Friedman, M. Sannicola, I. Shah, A systematic analysis of read-across within REACH registration dossiers, Computational Toxicology 30 (2024) 100304. doi:10.1016/j.comtox.2024.100304.
URL https://www.sciencedirect.com/science/article/pii/S2468111324000069

[25] T. Tate, J. Wambaugh, G. Patlewicz, I. Shah, Repeat-dose toxicity prediction with Generalized Read-Across (GenRA) using targeted transcriptomic data: A proof-of-concept case study, Computational Toxicology (Amsterdam, Netherlands) 19 (2021) 1–12. doi:10.1016/j.comtox.2021.100171.

[26] M. D. Nelms, C. L. Mellor, S. J. Enoch, R. S. Judson, G. Patlewicz, A. M. Richard, J. M. Madden, M. T. D. Cronin, S. W. Edwards, A Mechanistic Framework for Integrating Chemical Structure and High-Throughput Screening Results to Improve Toxicity Predictions, Computational Toxicology (Amsterdam, Netherlands) 8 (2018) 1–12. doi:10.1016/j.comtox.2018.08.003.

[27] M. Boyce, B. Meyer, C. Grulke, L. Lizarraga, G. Patlewicz, Comparing the performance and coverage of selected in silico (liver) metabolism tools relative to reported studies in the literature to inform analogue selection in read-across: A case study, Computational Toxicology (Amsterdam, Netherlands) 21 (2022) 1–15. doi:10.1016/j.comtox.2021.100208.

[28] I. Shah, G. Patlewicz, GenRA (2024).
URL https://www.comptox.epa.gov/genra

[29] R. Kunimoto, M. Vogt, J. Bajorath, Maximum common substructure-based Tversky index: an asymmetric hybrid similarity measure, Journal of Computer-Aided Molecular Design 30 (7) (2016) 523–531. doi:10.1007/s10822-016-9935-y.

[30] N. M. O'Boyle, J. Boström, R. A. Sayle, A. Gill, Using matched molecular series as a predictive tool to optimize biological activity, Journal of Medicinal Chemistry 57 (6) (2014) 2704–2713, number: 6. doi:10.1021/jm500022q.

[31] A matched molecular pair (MMP) approach for selecting analogs suitable for structure activity relationship (SAR)-based read across - ScienceDirect (Nov. 2021).
URL https://www.sciencedirect.com/science/article/abs/pii/S0273230021001069

[32] D. Bajusz, A. Rácz, K. Héberger, Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations?, Journal of Cheminformatics 7 (1) (2015) 20. doi:10.1186/s13321-015-0069-3.
URL https://doi.org/10.1186/s13321-015-0069-3

[33] M. Floris, A. Manganaro, O. Nicolotti, R. Medda, G. F. Mangiatordi, E. Benfenati, A generalizable definition of chemical similarity for read-across, Journal of Cheminformatics 6 (1) (2014) 39. doi:10.1186/s13321-014-0039-1.
URL https://doi.org/10.1186/s13321-014-0039-1

[34] D. Rogers, M. Hahn, Extended-Connectivity Fingerprints, Journal of Chemical Information and Modeling 50 (5) (2010) 742–754, publisher: American Chemical Society. doi:10.1021/ci100050t.
URL https://doi.org/10.1021/ci100050t

[35] J. L. Durant, B. A. Leland, D. R. Henry, J. G. Nourse, Reoptimization of MDL Keys for Use in Drug Discovery, Journal of Chemical Information and Computer Sciences 42 (6) (2002) 1273–1280, publisher: American Chemical Society. doi:10.1021/ci010132r.
URL https://doi.org/10.1021/ci010132r

[36] C. Yang, A. Tarkhov, J. Marusczyk, B. Bienfait, J. Gasteiger, T. Kleinoeder, T. Magdziarz, O. Sacher, C. H. Schwab, J. Schwoebel, L. Terfloth, K. Arvidson, A. Richard, A. Worth, J. Rathman, New publicly available chemical query language, CSRML, to support chemotype representations for application to data mining and modeling, Journal of Chemical Information and Modeling 55 (3) (2015) 510–528. doi:10.1021/ci500667v.

[37] R. E. Carhart, D. H. Smith, R. Venkataraghavan, Atom pairs as molecular features in structure-activity studies: definition and applications, Journal of Chemical Information and Computer Sciences 25 (2) (1985) 64–73, publisher: American Chemical Society. doi:10.1021/ci00046a002.
URL https://doi.org/10.1021/ci00046a002

[38] T. Hegeman, A. Iosup, Survey of graph analysis applications, ArXiv abs/1807.00382 (2018).

[39] A. T. Balaban, Topological indices based on topological distances in molecular graphs, Pure and Applied Chemistry 55 (2) (1983) 199–206, publisher: De Gruyter. doi:10.1351/pac198855020199.
URL https://www.degruyter.com/document/doi/10.1351/pac198855020199/html?lang=en

[40] A. Alameri, M. Alsharafi, TOPOLOGICAL INDICES TYPES IN GRAPHS AND THEIR APPLICATIONS, 2021.

[41] J. C. Dearden, The Use of Topological Indices in QSAR and QSPR Modeling, Vol. 24, Springer International Publishing, Cham, 2017, pp. 57–88, book Title: Advances in QSAR Modeling Series Title: Challenges and Advances in Computational

Chemistry and Physics. doi:10.1007/978-3-319-56850-8_2.
URL http://link.springer.com/10.1007/978-3-319-56850-8_2

[42] R. Todeschini, R. Cazar, E. Collina, The chemical meaning of topological indices, Chemometrics and Intelligent Laboratory Systems 15 (1) (1992) 51–59. doi:10.1016/0169-7439(92)80026-Z.
URL https://www.sciencedirect.com/science/article/pii/016974399280026Z

[43] J. Ullmann, An Algorithm for Subgraph Isomorphism, Journal of the ACM 23 (1) (1976) 31–42, type: Journal Article. doi:https://doi.org/10.1145/321921.321925.

[44] M. Pelillo, Replicator Equations, Maximal Cliques, and Graph Isomorphism, Neural Computation 11 (8) (1999) 1933–1955. doi:10.1162/089976699300016034.
URL https://direct.mit.edu/neco/article/11/8/1933-1955/6302

[45] S. Melnik, H. Garcia-Molina, E. Rahm, Similarity flooding: a versatile graph matching algorithm and its application to schema matching, in: Proceedings 18th International Conference on Data Engineering, IEEE Comput. Soc, San Jose, CA, USA, 2002, pp. 117–128. doi:10.1109/ICDE.2002.994702.
URL http://ieeexplore.ieee.org/document/994702/

[46] G. Jeh, J. Widom, SimRank: a measure of structural-context similarity, in: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '02, Association for Computing Machinery, New York, NY, USA, 2002, pp. 538–543. doi:10.1145/775047.775126.
URL https://doi.org/10.1145/775047.775126

[47] L. A. Zager, G. C. Verghese, Graph Similarity for scoring and matching, Applied Mathematics Letters 21 (1) (2008) 86–94, type: Journal Article. doi:https://doi.org/10.1016/j.aml.2007.01.006.

[48] D. Koutra, A. Parikh, A. Ramdas, J. Xiang, Algorithms for Graph Similarity and Subgraph Matching, Report, Carnegie Mellon University (2011).

[49] G. Chartrand, G. Kubicki, M. Schultz, Graph similarity and distance in graphs, aequationes mathematicae 55 (1) (1998) 129–145, type: Journal Article. doi:10.1007/s000100050025.
URL https://doi.org/10.1007/s000100050025

[50] V. J. Gillet, P. Willett, J. Bradshaw, Similarity searching using reduced graphs, Journal of Chemical Information and Computer Sciences 43 (2) (2003) 338–345. doi:10.1021/ci025592e.

[51] K. Birchall, V. J. Gillet, Reduced graphs and their applications in chemoinformatics, Methods in Molecular Biology (Clifton, N.J.) 672 (2011) 197–212. doi:10.1007/978-1-60761-839-3_8.

[52] K. Birchall, V. J. Gillet, G. Harper, S. D. Pickett, Training similarity measures for specific activities: application to reduced graphs, Journal of Chemical Information and Modeling 46 (2) (2006) 577–586. doi:10.1021/ci050465e.

[53] C. Garcia-Hernandez, A. Fernández, F. Serratosa, Ligand-based virtual screening using graph edit distance as molecular similarity measure, Journal of Chemical Information and Modeling 59 (4) (2019) 1410–1421. doi:10.1021/acs.jcim.8b00820.
URL https://doi.org/10.1021/acs.jcim.8b00820

[54] T. Akutsu, H. Nagamochi, Comparison and enumeration of chemical graphs, Computational and Structural Biotechnology Journal 5 (6) (2013) e201302004. doi:https://doi.org/10.5936/csbj.201302004.
URL https://www.sciencedirect.com/science/article/pii/S2001037014600325

[55] E. Duesbury, J. D. Holliday, P. Willett, Maximum Common Subgraph Isomorphism Algorithms, MATCH Communications in Mathematical and in Computer Chemistry 77 (2) (2017) 213–232, number: 2 Publisher: Sheffield.
URL http://match.pmf.kg.ac.rs/content77n2.htm

[56] J. W. Raymond, P. Willett, Maximum common subgraph isomorphism algorithms for the matching of chemical structures, Journal of Computer-Aided Molecular Design 16 (7) (2002) 521–533. doi:10.1023/A:1021271615909.

URL https://doi.org/10.1023/A:1021271615909

[57] C. Bron, J. Kerbosch, Algorithm 457: finding all cliques of an undirected graph, Commun. ACM 16 (9) (1973) 575–577. doi:10.1145/362342.362367.
URL https://dl.acm.org/doi/10.1145/362342.362367

[58] P. Durand, R. Pasari, J. Baker, C. Tsai, An Efficient Algorithm for Similarity Analysis of Molecules, Internet J. Chem. 2 (1999) 1–16.
URL https://www.cs.kent.edu/~jbaker/paper/

[59] A. Dalke, J. Hastings, FMCS: a novel algorithm for the multiple MCS problem, Journal of Cheminformatics 5 (Suppl 1) (2013) O6. doi:10.1186/1758-2946-5-S1-O6.
URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3606201/

[60] R. Kondor, J. Lafferty, Diffusion Kernels on Graphs and Other Discrete Input Spaces, ICML Vol. 2, type: Journal Article (2002).

[61] D. Haussler, Convolution kernels on discrete structures ucsc crl.

[62] R. Kondor, J. D. Lafferty, Diffusion kernels on graphs and other discrete input spaces, in: International Conference on Machine Learning.

[63] S. Vishwanathan, K. M. Borgwardt, N. N. Schraudolph, Fast Computation of Graph Kernels, in: B. Schölkopf, J. Platt, T. Hofmann (Eds.), Advances in Neural Information Processing Systems 19, The MIT Press, 2007, pp. 1449–1456. doi:10.7551/mitpress/7503.003.0186.
URL https://direct.mit.edu/books/book/3168/chapter/87579/Fast-Computation-of-Graph-Kernels

[64] N. M. Kriege, F. D. Johansson, C. Morris, A survey on graph kernels, Applied Network Science 5 (1) (2020) 1–42, number: 1 Publisher: SpringerOpen. doi:10.1007/s41109-019-0195-3.
URL https://appliednetsci.springeropen.com/articles/10.1007/s41109-019-0195-3

[65] T. Gärtner, A survey of kernels for structured data, SIGKDD Explor. Newsl. 5 (1) (2003) 49–58. doi:10.1145/959242.959248.
URL https://doi.org/10.1145/959242.959248

[66] K. M. Borgwardt, H. P. Kriegel, Shortest-path kernels on graphs, in: Fifth IEEE International Conference on Data Mining (ICDM'05), p. 8 pp. doi:10.1109/ICDM.2005.132.

[67] N. Shervashidze, P. Scweitzer, E. Jan van Leeuwen, K. Melhorn, Weisfeiler-lehman graph kernels, Journal of Machine Learning Research 12 (77) (2011) 2539–2561.

[68] H. Cai, V. W. Zheng, K. Chang, A Comprehensive Survey of Graph Embedding: Problems, Techniques, and Applications, IEEE Transactions on Knowledge & Data Engineering 30 (09) (2018) 1616–1637, type: Journal Article. doi:10.1109/TKDE.2018.2807452.
URL http://doi.ieeecomputersociety.org/10.1109/TKDE.2018.2807452

[69] M. Xu, Understanding Graph Embedding Methods and Their Applications, SIAM Review 63 (4) (2021) 825–853, type: Journal Article. doi:10.1137/20m1386062.
URL https://epubs.siam.org/doi/abs/10.1137/20M1386062

[70] P. Goyal, E. Ferrara, Graph embedding techniques, applications, and performance: A survey, Knowledge-Based Systems 151 (2018) 78–94, type: Journal Article. doi:https://doi.org/10.1016/j.knosys.2018.03.022.
URL https://www.sciencedirect.com/science/article/pii/S0950705118301540

[71] Kruskal (2024/06/18 1978). doi:10.4135/9781412985130, [link].
URL https://methods.sagepub.com/book/multidimensional-scaling

[72] J. B. Tenenbaum, V. de Silva, J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–23. doi:10.1126/science.290.5500.2319.

[73] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Computation 15 (6) (2003) 1373–1396. doi:10.1162/089976603321780317.

[74] M. Xu, Understanding graph embedding methods and their applications, arXiv:2012.08019 [cs, math] (Dec. 2020). doi:10.48550/arXiv.2012.08019.
URL http://arxiv.org/abs/2012.08019

[75] B. Perozzi, R. Al-Rfou, S. Skiena, DeepWalk: Online Learning of Social Representations, in: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 2014, pp. 701–710, arXiv:1403.6652 [cs]. doi:10.1145/2623330.2623732.
URL http://arxiv.org/abs/1403.6652

[76] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, arXiv:1301.3781 [cs] (Sep. 2013).
URL http://arxiv.org/abs/1301.3781

[77] A. Grover, J. Leskovec, node2vec: Scalable feature learning for networks, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2016).

[78] A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu, S. Jaiswal, graph2vec: Learning Distributed Representations of Graphs, ArXiv abs/1707.05005, type: Journal Article (2017).

[79] H. Chen, H. Koga, Gl2vec: Graph embedding enriched by line graphs with edge features, Neural Information Processing, Springer International Publishing, 2019, pp. 3–14. doi:https://doi.org/10.1007/978-3-030-36718-3_1.

[80] H. Cai, V. W. Zheng, K. C.-C. Chang, A comprehensive survey of graph embedding: Problems, techniques, and applications, IEEE Transactions on Knowledge Camp; Data Engineering 30 (09) (2018) 1616–1637. doi:10.1109/tkde.2018.2807452.
URL https://doi.ieeecomputersociety.org/10.1109/TKDE.2018.2807452

[81] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, G. Monfardini, The Graph Neural Network Model, IEEE Transactions on Neural Networks 20 (1) (2009) 61–80, type: Journal Article. doi:10.1109/TNN.2008.2005605.

[82] A systematic approach to simulating metabolism in computational toxicology. i. the times heuristic modelling framework., Curr Pharm Des. 10 (2004) 1273–1293. doi:10.2174/1381612043452596.

[83] Y. Djoumbou-Feynang, J. Fianmoncini, A. Gil-de-la Fuente, Biotransformer: a comprehensive computational tool for small molecule metabolism prediction and metabolite identification., J Cheminform 11 (2019). doi:https://doi.org/10.1186/s13321-018-0324-5.

[84] G. L. Landrum, RDKit: Open-source cheminformatics;.
URL http://www.rdkit.org

[85] G. Siglidis, G. Nikolentzos, S. Limnios, C. Giatsidis, K. Skianis, M. Vazirgiannis, GraKeL: A Graph Kernel Library in Python, arXiv:1806.02193 [cs, stat] (Mar. 2020). doi:10.48550/arXiv.1806.02193.
URL http://arxiv.org/abs/1806.02193

[86] P. Pradeep, R. Judson, D. M. DeMarini, N. Keshava, T. M. Martin, J. Dean, C. F. Gibbons, A. Simha, S. H. Warren, M. R. Gwinn, G. Patlewicz, Evaluation of Existing QSAR Models and Structural Alerts and Development of New Ensemble Models for Genotoxicity Using a Newly Compiled Experimental Dataset, Computational Toxicology (Amsterdam, Netherlands) 18 (May 2021). doi:10.1016/j.comtox.2021.100167.

[87] R. Benigni, C. Bossa, Data-based review of QSARs for predicting genotoxicity: the state of the art, Mutagenesis 34 (1) (2019) 17–23. doi:10.1093/mutage/gey028.

[88] M. Honma, A. Kitazawa, A. Cayley, R. V. Williams, C. Barber, T. Hanser, R. Saiakhov, S. Chakravarti, G. J. Myatt, K. P. Cross, E. Benfenati, G. Raitano, O. Mekenyan, P. Petkov, C. Bossa, R. Benigni, C. L. Battistelli, A. Giuliani, O. Tcheremenskaia, C. DeMeo, U. Norinder, H. Koga, C. Jose, N. Jeliazkova, N. Kochev, V. Paskaleva, C. Yang, P. R. Daga, R. D. Clark, J. Rathman, Improvement of quantitative structure-activity relationship (QSAR) tools for predicting Ames mutagenicity: outcomes

of the Ames/QSAR International Challenge Project, Mutagenesis 34 (1) (2019) 3–16. doi:10.1093/mutage/gey031.

[89] A. Furuhama, A. Kitazawa, J. Yao, C. Matos Dos Santos, J. Rathman, C. Yang, J. Ribeiro, K. Cross, G. Myatt, G. Raitano, E. Benfenati, N. Jeliazkova, R. Saiakhov, S. Chakravarti, R. Foster, C. Bossa, C. Battistelli, R. Benigni, T. Sawada, H. Wasada, T. Hashimoto, M. Wu, R. Barzilay, P. Daga, R. Clark, J. Mestres, A. Montero, E. Gregori-Puigjané, P. Petkov, H. Ivanova, O. Mekenyan, S. Matthews, D. Guan, J. Spicer, R. Lui, Y. Uesawa, K. Kurosaki, Y. Matsuzaka, S. Sasaki, M. Cronin, S. Belfield, J. Firman, N. Spînu, M. Qiu, J. Keca, G. Gini, T. Li, W. Tong, H. Hong, Z. Liu, Y. Igarashi, H. Yamada, K. Sugiyama, M. Honma, Evaluation of qsar models for predicting mutagenicity: outcome of the second ames/qsar international challenge project, SAR and QSAR in Environmental Research 34 (12) (2023) 983–1001, pMID: 38047445. arXiv:https://doi.org/10.1080/1062936X.2023.2284902, doi:10.1080/1062936X.2023.2284902.
URL https://doi.org/10.1080/1062936X.2023.2284902

[90] R. Benigni, Towards quantitative read across: Prediction of Ames mutagenicity in a large database, Regulatory Toxicology and Pharmacology 108 (2019) 104434. doi:10.1016/j.yrtph.2019.104434.
URL https://www.sciencedirect.com/science/article/pii/S0273230019301989

[91] B. Rozemberczki, O. Kiss, R. Sarkar, Karate Club: An API Oriented Open-source Python Framework for Unsupervised Learning on Graphs, in: Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), ACM, 2020, p. 3125–3132.

[92] L. van er Maaten, G. Hinton, Visualizing Data using t-SNE., Journal of Machine Learning Research 8 (2018) 2579–2605.

[93] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[94] S. Brody, U. Alon, E. Yahav, How attentive are graph attention networks?, ArXiv abs/2105.14491 (2021).