A peer-reviewed version of this preprint was published in PeerJ on 19 January 2016.

View the peer-reviewed version (peerj.com/articles/1558), which is the preferred citable publication unless you specifically need to cite this preprint.

https://doi.org/10.7717/peerj.1558
Mining integrated semantic networks for drug repositioning opportunities

Joseph Mullen, Simon J Cockell, Hannah Tipney, Peter Woolland, Anil Wipat

Current research and development approaches to drug discovery have become less fruitful and more costly. One alternative paradigm is that of drug repositioning. Many marketed examples of repositioned drugs have been identified through serendipitous or rational observations, highlighting the need for more systematic methodologies to tackle the problem. Systems level approaches have the potential to enable the development of novel methods to understand the action of therapeutic compounds, but requires an integrative approach to biological data. Integrated networks can facilitate systems level analyses by combining multiple sources of evidence to provide a rich description of drugs, their targets and their interactions. Classically, such networks can be mined manually where a skilled person is able to identify portions of the graph (semantic subgraphs) that are indicative of relationships between drugs and highlight possible repositioning opportunities. However, this approach is not scalable. Automated approaches are required to systematically mine integrated networks for these subgraphs and bring them to the attention of the user. We introduce a formal framework for the definition of integrated networks and their associated semantic subgraphs for drug interaction analysis and describe DReSMin, an algorithm for mining semantically-rich networks for occurrences of a given semantic subgraph. This algorithm allows instances of complex semantic subgraphs that contain data about putative drug repositioning opportunities to be identified in a computationally tractable fashion, scaling close to linearly with network data. We demonstrate the utility of our approach by mining an integrated drug interaction network built from 11 sources. This work identified and ranked 9,643,061 putative drug-target interactions, showing a strong correlation between highly scored associations and those supported by literature. We discuss the 20 top ranked associations in more detail, of which 14 are novel and 6 are supported by the literature. We also show that our approach better prioritizes known drug-target interactions, than other state-of-the-art approaches for predicting such interactions.
Mining integrated semantic networks for drug repositioning opportunities

Joseph Mullen¹, Simon J. Cockell², Hannah Tipney³, Peter Woollard³, and Anil Wipat¹,*

¹Interdisciplinary Computing and Complex BioSystems Research Group, School of Computing Science, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK
²Bioinformatics Support Unit, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK
³Computational Biology Department, Quantitative Sciences, GlaxoSmithKline Research & Development Ltd, Gunnels Wood Road, Stevenage, Hertfordshire, UK.
*E-mail: anil.wipat@ncl.ac.uk
*Phone: +44-191-222-8213

ABSTRACT

Current research and development approaches to drug discovery have become less fruitful and more costly. One alternative paradigm is that of drug repositioning. Many marketed examples of repositioned drugs have been identified through serendipitous or rational observations, highlighting the need for more systematic methodologies to tackle the problem. Systems level approaches have the potential to enable the development of novel methods to understand the action of therapeutic compounds, but requires an integrative approach to biological data. Integrated networks can facilitate systems level analyses by combining multiple sources of evidence to provide a rich description of drugs, their targets and their interactions. Classically, such networks can be mined manually where a skilled person is able to identify portions of the graph (semantic subgraphs) that are indicative of relationships between drugs and highlight possible repositioning opportunities. However, this approach is not scalable. Automated approaches are required to systematically mine integrated networks for these subgraphs and bring them to the attention of the user. We introduce a formal framework for the definition of integrated networks and their associated semantic subgraphs for drug interaction analysis and describe DReSMin, an algorithm for mining semantically-rich networks for occurrences of a given semantic subgraph. This algorithm allows instances of complex semantic subgraphs that contain data about putative drug repositioning opportunities to be identified in a computationally tractable fashion, scaling close to linearly with network data. We demonstrate the utility of our approach by mining an integrated drug interaction network built from 11 sources. This work identified and ranked 9,643,061 putative drug-target interactions, showing a strong correlation between highly scored associations and those supported by literature. We discuss the 20 top ranked associations in more detail, of which 14 are novel and 6 are supported by the literature. We also show that our approach better prioritizes known drug-target interactions, than other state-of-the art approaches for predicting such interactions.

Keywords: drug repositioning, data integration, data mining, semantic networks, semantic subgraphs, bioinformatics, systems approaches

INTRODUCTION

Drug repositioning is the application of established, approved compounds to a novel therapeutic application. This process is a rapidly-evolving issue in the area of drug development, having the potential to reduce both drug development costs and the time taken for a drug to reach the market. Many repositioned drugs currently on the market have been discovered through either serendipitous or rational observations. However, these manual approaches are not efficient given the potentially huge search space of drug-target (D-T) interactions. Systematic approaches to searching for repositioning opportunities are required to provide an efficient and scalable alternative to manual investigations.

A large number of studies have detailed computational approaches to aid in the systematic identification of drug repositioning opportunities, including methodologies based on: chemical structure (Keiser et al., 2009), protein structure and molecular docking (Moriaud et al., 2011), phenotype similarity (such as
side-effect similarity (Yang and Agarwal, 2011) and gene expression similarity (Lamb et al., 2006)) or genetic variation (Sanseau et al., 2012).

More recent approaches to drug repositioning focus on the creation of integrated networks which combine data from multiple analyses, to give a systems level view of cellular and molecular processes (Chen et al., 2012; Cockell et al., 2010; He et al., 2011). This approach provides a complementary path to reductionist science in understanding complex phenomena. Semantically-rich integrated networks, which utilise a graph-based representation, are a convenient method of representing the types of integrated data necessary for finding drug repositioning opportunities (Betzler et al., 2011). In graph-based data, entities, such as proteins or drugs, are represented as vertices. Interactions between these entities, such as protein-protein interactions or a drug binding to a protein are captured in edges. In semantic graphs, each vertex and edge in the graph is assigned a type from a predefined set. Vertices and edges are also annotated with attributes. Graph representations of complex systems are widely used in computer science, social and technological network analysis science due to their ability to structure and semi-structured data (Riaz and Ali, 2011). Within bioinformatics graph-based representations are also widely adopted, particularly as a means of representing data produced during an exercise in data integration and in protein-protein interactions networks.

In the context of these integrated networks, subgraphs are connected components of the parent network (Gallagher, 2006). These subgraphs formally capture local relationships between the elements represented in the graph. Often, the relationships in a given subgraph are indicative of a particular biological phenomenon. In the case of drug repositioning networks, the types of relationships include amongst others: interactions between drugs and their targets, interactions between targets, and the diseases associated with particular targets. Therefore, within the integrative graph are subgraphs that describe repositioning opportunities as a result of their semantic and topological properties. Once appropriate subgraphs have been observed and defined they can be used as templates to find instances of these subgraphs, and related subgraphs, within a given graph to highlight similar drug repositioning opportunities.

For example, chlorpromazine is an anti-psychotic drug that is also approved as an antihistamine (Mitchell, 1993). The interactions of chlorpromazine can be captured in an integrated network (Fig. 1). Data from DrugBank version 2.5 (DBv2.5) (Wishart, 2006) provides three interactions between chlorpromazine and single protein targets; none of these interactions explain the antihistaminic affects of the drug. Structurally, chlorpromazine is very similar to the antiemetic trimipramine. DBv2.5 captures an interaction between trimipramine and the Histamine H1 receptor, a known target for antihistamines. Through guilt-by-association, we can therefore predict the Histamine H1 receptor as a target for chlorpromazine, an interaction captured in the latest editions of the DrugBank database. The topological and semantic properties of the subgraph depicted in Fig. 1B describe a repositioning relationship that could be generically applicable to any two drugs and their target. Fig. 1B describes a situation whereby a compound, structurally similar to a compound with a known target, may also bind to the same target (the inference is represented as the dashed line). This real example can therefore be used to derive a template semantic subgraph that can be used for searching for similar, but novel, drug-target associations relationships involving different drugs and targets. This template semantic subgraph therefore describes a pattern indicative of a drug interaction with a target, highlighting potential new indications for the drug. Although Fig. 1 shows a simple triad, semantic subgraphs capturing data relevant to repositioning opportunities are likely to be more complex. In the context of drug repositioning, manual identification of potential repositioning opportunities from large target networks is possible, though not efficient for the systematic analysis of such large networks. The definition of semantic subgraphs for known repositioning opportunities, in combination with an algorithm for the mining of integrated complex networks for these subgraphs, allows us to highlight potential repositioning in a more systematic and exhaustive fashion.

In this paper we introduce a formal framework for the definition of a semantic subgraph for integrated networks. We also present DReSMin (Drug RePositioning Semantic Mining), an algorithm for searching integrated networks for occurrences of a given semantic subgraph using semantic distance thresholds. DReSMin optimises the search time for larger subgraphs by including a semantic graph pruning step and applying a method for splitting large subgraphs prior to searching. We demonstrate the utility of our approach by searching an integrated drug dataset for semantic subgraphs that are indicative of drug repositioning opportunities, particularly focusing on inferring D-T interactions. As part of this work we updated an existing integrated dataset used for in silico drug discovery (Cockell et al., 2010). Finally we
demonstrate that our approach can be successfully used to predict putative D-T interactions that were not explicitly represented in the integrated network.

Graphs

Definition of our graph model  A graph $G$ is defined as a ordered pair $(V,E)$, where $V$ is a set of vertices (or nodes), and $E \subseteq V \times V$ is a set of edges (or relations). Each $e \in E$ is a pair $(v_i, v_j)$ where $v_i, v_j \in V$. Edges represent relations between vertices. Edges may be directed or undirected. Both vertices and edges may be labelled, typed and attributed.

DReSMin, requires a directed (edges have a direction associated with them) graph where vertices and edges are labelled with types $T_v$ and $T_e$ respectively, where $T_v$ and $T_e$ are drawn from a finite hierarchy of types $H$, and are annotated with attributes. The algorithm allows for multigraphs (vertices $v_i$ and $v_j$ are permitted to have multiple edges between them) and for vertices to contain self-loops ($v_i$ may have an edge directed toward itself). For the remainder of this paper $|V(G)|$ will be used to represent the number of vertices contained in graph $G$.

Classical subgraph definition  Subgraph isomorphism is a task in which two graphs, $G$ & $Q$ are given as input and one must determine whether $G$ contains a subgraph that is isomorphic to $Q$: is there a subgraph $G'(V',E') : V' \subseteq V, E' \subseteq E$? During the search of a query graph, a mapping $(M)$ is expressed as the set of ordered pairs $(v,m)$ (with $v \in G$ and $m \in Q$) and so $M = \{(v,m) \in V_G \times V_Q | v$ is mapped onto $m\}$; that is $M : G' \rightarrow Q$.

Semantic subgraph definition  A semantic subgraph is defined as $Q = (V,E,T_v,t_v,T_e,t_e)$, where $V$ is a set of vertices, $E$ is a set of edges, $T_v$ is a set of node types and $T_e$ is a set of edge types. $t_v : V \rightarrow T_v$ and $t_e : E \rightarrow T_e$ are surjective functions; each node is assigned a node type and each edge an edge type from $T_v$ or $T_e$ respectively. A semantic subgraph aims to infer a relation between vertices of a particular $t_v$, where a relation does not exist; for example between a compound and a target (Fig. 1B).

Graph matching

Several approaches have been described for combining semantic information with network motif topology including the list coloured motif problem (Betzler et al., 2011; Lacroix et al., 2006). In this case a motif $(M)$ is defined as a multiset of colours, or types. An occurrence of $M$ is a subset of vertices that forms a connected subgraph whose multiset of colours, or types, matches that of $M$ exactly (Lacroix et al., 2006). Although this approach demonstrates how network motifs may be extended to incorporate semantic information, it does not allow for topological exacts to be identified. The ability to identify sub-components of a target network that match a defined topology is a necessity in situations where the topology of a subgraph is believed to aid in describing the functionality of the sub-component. The task of identifying mappings of a predefined subgraph with similar topology from a larger graph is known as the graph matching problem (Gallagher, 2006).

There are different variations of the graph-matching problem. For example, exact matching occurs when the mapping between the vertices of the two graphs is edge-preserving; a mapping contains all edges defined by the query. One of the most stringent forms of exact matching is subgraph isomorphism (Conte et al., 2004) which aims to find all occurrences of a query graph and is an NP-complete problem (Washio and Motoda, 2003). Currently, algorithms addressing this problem are exponential in performance relative to the size of the input graphs (Gallagher, 2006). Many algorithms which have been developed to address the subgraph isomorphism problem are based on the exhaustive algorithm developed by Ullmann (1976). Applying an exhaustive method to the identification of drug repositioning opportunities is important to ensure all possible novel applications for a drug are investigated. Using a backtracking approach, Ullmann’s algorithm finds solutions by incrementing partial solutions or abandoning them when determining they cannot be completed (Ullmann, 1976). An extension of the Ullman approach, incorporating the semantics of a graph, has been implemented using inexact (Djoko et al., 1997), as well as exact approaches (Cordella et al., 2004; Giugno and Shasha, 2002). However, as yet, none of these approaches have been applied to the automated identification of drug repositioning opportunities.

Whilst searching for semantic subgraphs it is important to consider the similarity between the query subgraph and the target, both in terms of graph topology and the meaning of the annotations on vertices and edges. A measurement of semantic similarity between elements of a mapping and the equivalent element in a query must be introduced to the search and the degree of similarity can be expressed as a
We have developed DReSMin, an algorithm for the detection of semantic subgraphs. This algorithm returns all mappings of a semantic subgraph that match, semantically, at a level equal to, or above our threshold, \( ST \). In this case our application for the algorithm is the identification of a semantic subgraph \( Q \) which may be indicative of drug repositioning opportunities within a target graph \( G \). Examples of semantic subgraphs may be initially drawn from a set of templates, that is the graph representation of known repositioned drugs, such as chlorpromazine, shown in Fig. 1A. The algorithm is made up of four main components which are described in Fig. 2. These components comprise: (i) Semantic graph pruning (ii) Topological search (iii) Semantic subgraph distance exclusion (iv) Semantic subgraph splitting.

Semantic graph pruning We are concerned only with identifying semantic subgraphs that match, semantically, at a level equal to, or above our threshold. (Note: In this work, the semantic distance between two graph entities is calculated using the semantic distance calculator described in the ‘Semantic subgraph distance exclusion’ section.) In this graph pruning component of the algorithm, any vertices (and their associated edges) in \( G \) which are above a certain semantic distance from those in \( Q \) are removed from \( G \). This step allows any vertices that are semantically distant from our query to be removed prior to a search, cutting down the search space. Taking \( G, Q \) and a semantic threshold \( ST \) each \( v \in T_r(Q) \) are sent to the semantic subgraph distance calculator (termed SDC and described later in the manuscript), and scored against every \( v \in T_r(G) \). If \( SDC(t_i(Q), t_j(G)) < ST \) then all \( v \in V(G) \) of type \( t_i \) are removed from \( G \) as well as any \( e \in E \) where \( v = v_i \) or \( v = v_j \). Finally after all semantically insignificant elements are removed from \( G \), all isolated \( v \in V(G) \) that may have resulted from the edge pruning step are also removed.

Topological matching Many algorithms addressing the problem of subgraph isomorphism build on Ullman’s work. These applications include: GraphQL (He and Singh, 2008), GADDI (Zhang et al., 2009) and, one of the most efficient, the VF algorithm (Cordella et al., 1999). Performance is increased in these algorithms by exploiting different join orders, pruning rules and auxiliary information to prune out negative candidate subgraphs as early as possible. We carry out topological matching using a variation of the VF algorithm (Cordella et al., 1999). The VF algorithm is exhaustive and suitable for working with ‘large’ graphs (up to \( 3 \times 10^4 \) vertices) and employs a depth-first strategy implemented in a recursive fashion (Cordella et al., 1999). During a search using the VF algorithm, the search space is minimised via the introduction of topological pruning rules (Cordella et al., 1999). Integrated networks typically surpass the aforementioned ‘large’ graphs in size, particularly true within the biological and pharmaceutical settings. As data volumes continue to grow (e.g. omics technologies continue to mature) it is important to develop exhaustive algorithms capable of scaling with the data.

Our initial implementation of the VF algorithm showed poor scalability and so, as an enhancement to the VF algorithm, we introduce three steps to improve the efficiency of searching for topological subgraphs. These three steps include: a set of rules used to determine the appropriate vertices at which an instance of the search is started (initial candidate set), as described in (1) below; a topological pruning rule, based on a closed world assumption, as described in (2) below; and a semantic thresholding step (described in the next section of the manuscript). Focussing on the identification of new indications for existing compounds it is vital that mappings of semantic subgraphs, in this work, contain a compound.

1. When considering an initial candidate set of nodes from the target graph \( G \) at which to initiate
the search, it is desirable to try to ensure that the set is made up of nodes of a type, $X$, such as

\textbf{Compound} to ensure the relevance of the portion of graph being searched. Therefore, an initial

candidate set for the search is chosen based on: all $v \in V(Q)$ whose $t_v \in T_v(G) = X$ are considered

with $v > \text{deg}_v(v)$ (where \text{deg} represents the degree of a node) selected as $v$. $m$ is made up of all

$v \in V(G)$ whose $\text{deg}_v(v) \geq \text{deg}_v(v)$ and $t_v \in T_v(G) = X$.

2. When mining with a given semantic subgraph that describes a potential repositioning situation we

must assume that the lack of a relationship between nodes indicates the absence of a relationship

between the two nodes (a closed world assumption). As a result, when searching for a given

semantic subgraph, $Q$, we only consider a match if there exists no additional edges between the

vertices in a mapping $M$ from the target graph $G$, and their equivalent vertices in $Q$. Therefore,

a mapping $M$ is expressed as a set of ordered pairs and the closed world assumption requires

$(M = \text{match}) \lor (\text{deg}(v) \in (G) \equiv \text{deg}(m) \in V(Q))$.

\textbf{Semantic subgraph distance exclusion} Semantic thresholding is used to exclude matches found in $G$

that are below a given semantic distance from $Q$. This process is achieved through a semantic subgraph

distance calculator (SDC). An SDC comprises of two distance matrices, one for $t_v \in T_v(G)$ and one for

$t_v \in T_v(G)$. We have $n = 18(t_v)$ and $n = 42(t_e) =>$ each matrix is represented as matrix $P' = [p_{ij}]$, the

$n \times n$ matrix defined by;

\[ p_{ij} = \begin{cases} 
1 & \text{if } p_i \text{ is semantically identical to } p_j; \\
0 & \text{if } p_i \text{ is semantically unrelated to } p_j; \\
-1 & \text{if } p_i \text{ is semantically opposite to } p_j. 
\end{cases} \tag{1} \]

During the matching process each element of $M = (V_m, E_m)$ is scored against its equivalent in

$Q = (V_s, E_s)$. The resulting \textit{semantic score (SS)} of $M$ is;

\[ \sum_{m_1, q_1} SDC(m_1, q_1), SDC(m_2, q_2) \ldots SDC(m_n, q_n) \tag{2} \]

A semantic threshold (ST) is defined by the user prior to a search; a value ranging from 0 to 1. During

the search, vertices and edges pass or fail the semantic threshold. Thus we identify topological exacts and

semantic closeness.

\textbf{Semantic subgraph splitting} This component takes a semantic subgraph, $Q$, and returns a set of

semantic subgraphs, $D$, whose $|V| < 4$. In Fig 2. we see how this step interacts with the other components

of DReSMi. $\forall d \in D$ produced during this step of DReSMi the target network, $G$, is pruned using the

semantic graph prune component and $d$, before $d$ is searched for in $G$. The graph splitting component

allows smaller subgraphs to be searched and mappings joined based on sharing a common overlapping

node (ON). In order for this approach to be successful a semantic subgraph is first converted to an

undirected graph. The most connected node, $v_{\text{max}}(Q)$, is then identified and used as ON. Of all the

remaining $v \in V(Q)$, the two most distant vertices $(v_1, v_2)$ from $Q$ are selected. Two new graphs ($D_1 & D_2$)

are then created and populated with nodes as such: $V(D_1) \cup v \in \delta(v_1, ON), V(D_2) \cup v \in \delta(v_2, ON)$, that

is every node in the shortest path from $v_1$ to $ON$ is included in $D_1$ and every node in the shortest path from

$v_2$ to $ON$ is included in $D_2$. Remaining vertices are then allocated depending on which graph they share a

connecting edge with (Fig. 3). Edges are then allocated as such: $\forall e \in E(Q)$ if either $V(D_1)$ or $V(D_2)$

contains both $(v_j, v_i)$ of $e$; $e$ is allocated to that graph. Any edges whose nodes are are not found in the

same graph are not allocated to the split subgraphs. As a result of this process during a search we have $D_1$

and $D_2$ as well as our original semantic subgraph, $Q$. A search is then started with $D_1$ or $D_2$, depending

on which has the smallest $|V|$. The search is started using ON, maintaining the edge set it possessed in

$Q$, reducing the initial candidate set. All starting vertices that lead to an embedding being identified are

then passed to the second search; reducing the initial candidate set once more. All matches from the two

searches who share a common ON are then mapped and a final check for any $e \in E(Q)$ that were not

allocated to either $D_1$ or $D_2$ is made. This splitting may be called iteratively if either $D_1$ or $D_2$ still possess a

$|V| > 3$ after the first round of splitting, as demonstrated in Fig. 3. Subsequent searching will result in the

same set of mappings that would be identified by a non-split search (for algorithm pseudo-code and
discussion please see Article S1).
**Ranking Inferred Interactions**  Scoring of a semantic subgraph, $Q$, is achieved by determining the number of known D-T interactions in the predicted total set of D-T interactions inferred by $Q$. We refer to the complete set of inferred interactions as $Q(I)$. A score $R_q$ is calculated based on the ability of $Q$ to identify D-T interactions captured in DBv3, but not present in our Dat integrated data set (see next section). The set of interactions that are captured in DBv3, but not captured in Dat is known as $DBv3Rel$ (Equation 5.).

$$R_q(Q) = \frac{|Q(I) \cap DBv3Rel|}{|Q(I)|}$$  (3)

Once $R_q$ is calculated for each semantic subgraph we then score individual D-T interactions, $i$, based on the cumulative score of all semantic subgraphs that predicted $i$.

$$R_i(i) = \sum_{i \in Q'(I)} R_q(Q')$$  (4)

DReSMin is an exhaustive algorithm, as such, scoring inferred interactions allows for ranking, with those ranked higher inferred with greater confidence than others.

**Characterisation & Application**  An integrated dataset for in silico drug discovery has been described previously by Cockell and co-workers (Cockell et al., 2010). This dataset satisfies the requirements described for our algorithm (see 'Definition of our graph model' section) and so was used to test the algorithm performance and mined for D-T interactions using a Java based implementation of DReSMin.

The dataset was developed in Ondex (Köhler et al., 2006) and includes compounds and targets from DrugBank1 (Wishart, 2006), Proteins from UniProt2 (UniProt Consortium, 2013) as well as information from eleven other databases and analysis methods (Cockell et al., 2010). An updated version of this dataset was used as a test bed for this work, however the approach we describe is valid for most integrated networks that adopt a semantically rigorous approach to edge and vertex type definition.

Utilising a graph-based data representation and providing a framework for visualisation, both vertices and edges within an Ondex graph are annotated with semantically enriched metadata. Each vertex (or concept) is assigned a $c \in C$, where $C$ is a finite set of conceptClasses, while each edge or relation is assigned a $r \in R$ where $R$ is a finite set of relationTypes (Köhler et al., 2006). As part of this work we developed plug-ins (parsers and mappers) for the Ondex platform to extend the original dataset. These plug-ins allowed us to add disease conceptClass, taken from the National Drug File Reference Terminology (NDF-RT)3. Four relationTypes showing interactions between Disease-Disease (has_parent and has_child) and Compound-Disease (may_treat and may_prevent) originally defined in NDF-RT were also integrated. A final relationType between Target-Disease (involved_in) was integrated from DisGeNET4 (Bauer-Mehren et al., 2010).

The updated dataset, which we refer to as Dat from here on in has an additional 4,463 vertices (155,316) made up of 19 conceptClasses (see Table S1) in comparison to the original, together with an additional 28,736 edges (816,096), representing 42 relationTypes (see Table S2). The metagraph of the dataset described is shown in Fig. S1, with a subsection shown in Fig. 4.

This graph shows a high degree of connectivity with a $d_S(G)$ (average node degree) of 10.42, whereby degrees of vertices range from $\delta(G)$ (minimum degree) of 1 and $\Delta(G)$ (highest degree) of 15,004. Average connectivity differs between conceptClasses, with Proteins displaying the highest $d_S(G)$ of any conceptClass at $\sim$ 45. Other notable connectivity averages include Target $\sim$ 13, Compound $\sim$ 7 and Disease $\sim$ 4. All searches presented here were carried out using a semantic threshold (ST) of 0.8 (see Article S2). We only include vertices of type Compound in our initial candidate set.

**Drug-Target interaction prediction evaluation**  We compared our ranked set of predicted D-T interactions to those produced by another state-of-the-art method for drug target interaction prediction - a ligand-based method. Many approaches to drug-target prediction use ligand-based methods. These

---

1http://www.drugbank.ca
2http://www.uniprot.org
4Gene-disease association data were retrieved from the DisGeNET Database, GRIB/IMIM/UPF Integrative Biomedical Informatics Group, Barcelona. <http://www.disgenet.org/> [September, 2013].
methods use only ligand information to create models that learn which sub-structural features of a ligand correlate with activity against a particular target (Alvarsson et al., 2014). One implementation of such an approach is provided by ChEMBL. ChEMBL provide two models for target prediction, using bioactivity data with a cut-off of 1µM and 10µM respectively. These models allow for n predicted interactions to be made for a given drug. Inferred interactions are also scored and can be ranked, meaning a direct comparison to our approach can be achieved. Predictions using the ChEMBL models can be found in compound report cards, accessed via their website.

Mappings between DrugBank and ChEMBL compounds were retrieved from UniChem (Chambers et al., 2013) via whole source mapping. This mapping provides a set of 3,765 drugs that are contained in both datasets, of which 57 of the ChEMBL ids mapped to >1 DrugBank ID (one to four, five to three, and 51 to two). DReSMIn inferred drug-target associations for 2,223 of drugs common to both databases.

In the comparison presented below we only consider drug-target interaction inferences involving this set of 2,223 drugs involved in these associations.

For each of the 2,223 drugs, we identified associations with single proteins. The top 100 of these associations were identified using the ChEMBL Web resource client. Any interactions already captured in Dat, or interactions from organisms other than humans, were excluded from the analysis. This process was repeated for both the 1µM and the 10µM ChEMBL models, giving us two sets of predicted drug-target associations. In order for a fair comparison to be made for each of the 2,223 drugs the top 108 inferred single protein targets were collated and ranked. This process resulted in three sets of 216,251 ranked drug-target interactions; DReS, Chem1 and Chem10.

RESULTS

Characterisation and performance of DReSMIn

We evaluated the effectiveness of each step of our algorithm by adding each step (initial candidate set selection, topological pruning and semantic distance thresholding) sequentially to the basic topological search algorithm and then comparing the efficiency of each modified version to the VF2 topological search. The algorithm was implemented on a 20 node Ivy-Bridge bioinformatics cluster. Performance was measured as the time taken for a complete search for a semantic subgraph (\(Q\)) within a given target graph (\(G\)). Random semantic target graphs (\(\text{Ran}\)) as well as random semantic subgraphs were produced in order to evaluate the performance of the semantic subgraph search strategy. These random graphs were formulated using an approach that attempted to replicate the semantic and topological properties of Dat. In these random target graphs \(\forall v \in V(\text{Ran})\) of type \(t_v\), the average \(\deg^{-}(t_v)\) and the average \(\deg^{+}(t_v)\) were maintained \(\forall t_v \in T_v(\text{Dat})\). Experiments were repeated 10 times.

The SDC and graph-pruning step display linear running times of \(O(n)\); with the former capable of scoring \(8 \times 10^3\) concept pairs per second and the latter taking < 1 second to prune a graph G, with \(|V(G)|\) of \(1 \times 10^6\). During the performance measures we focused on semantic subgraphs with between 3-6 vertices. The effect on search time when altering semantic subgraph edgset size was also examined (Fig. S2) showing an improvement in performance as the edgset size increases. This performance increase is due to the fact that fewer nodes satisfy the more stringent topological rules. With more stringent pruning during a run of the algorithm the search space at each state is reduced; ultimately meaning that when searching for semantic subgraphs who share the same \(|V|\) but have differing \(|E|\), the semantic subgraph with the \(>|E|\) will be more efficient to search for.

Once semantic subgraphs reach a \(|V(Q)|\) of 4 then restricting the initial candidate set to include only Compounds improves performance. It is at this point the benefits of reducing the initial candidate set successfully reduce the search space, concomitantly increasing performance (Fig. 5). A similar phenomenon is observed with the introduction of the closed world check, whereby the real performance benefits are apparent when semantic subgraphs reach a \(|V(Q)|\) of 4 (Fig. 5). By restricting the initial candidate set as well as using the closed world assumption a two fold increase in performance in comparison to a purely topological approach was observed. Performance is further enhanced when utilising the semantic distance calculator demonstrating an almost 10 fold performance boost when comparing to the purely topological approach.

---

1https://www.ebi.ac.uk/chembl/
2https://github.com/chembl/chembl_webresource_client
3x = 100 or, if DReSMIn inferred < 100 targets for this drug, x = number of DReSMIn inferred targets
The semantic graph prune step introduces a small but noticeable increase in performance to DReSMIn. Despite an overall increase in performance the graph prune step also brings a subtle cost; any potential matches containing an element that scores < ST when passed to the SDC will not be returned. It is for this reason that the graph pruning step is an optional add-on to the DReSMIn algorithm. It is most useful when one wishes to return matches that are semantically exact to the semantic subgraph being used as a query \( Q \). The graph split step can potentially reduce the search time for \( Q \) from that of a \(|V(Q)|\) of 6 to one closer to the sum of a search for a subgraph with a \(|V(Q)|\) of 3 and a subgraph with a \(|V(Q)|\) of 4. It is this step that produces the greatest improvement to performance. For example, when using the SDC to search for \( Q \), where \(|V(Q)| = 6 \) in \( G \) when \(|V(G)| = 1 \times 10^5 \), takes 60 seconds, using the graph split method reduces this search time to just over 8 seconds, a 7 fold increase in performance.

Overall, when using all three of the algorithmic steps in DReSMIn, the performance of DReSMIn showed performance characteristics approximating a linear scale closer to \( O(n) \). This is in contrast to the exponential scaling characteristics observed for the purely topological search algorithm, VF2. These DReSMIn performance characteristics were observed for semantic subgraphs of size \(|V(Q)| \leq 6 \) (Fig. 5). Using DReSMIn with the hardware described above it is possible to complete an exhaustive, exact search for a 6 node semantic subgraph in a target graph containing \( >1.5 \times 10^5 \) vertices in under 10 seconds. The accuracy of the algorithm does not decrease as the target graph connectivity, or \(|E|\), increases (Fig. S3) or as the target graph \(|V|\) increases (Fig. S4).

**Application to search for drug-target interactions**

Semantic subgraphs were identified in \( Dat \) and used to infer novel potential D-T interactions in \( Dat \) using the DReSMIn algorithm. To aid in this process we utilised more recent D-T versions of the DrugBank datasets that were not used to build \( Dat \). This approach allows us to determine if D-T interactions inferred from \( Dat \) using DReSMIn are likely to be supported as more knowledge is obtained. We can thus understand if inferences made have any potential value to drug repositioning now, as opposed to in the future.

To carry out this process the D-T interactions from DBv2.5 that were integrated into \( Dat \) were retrieved and captured in the set \( DatRel \). We used DBv2.5 to construct \( Dat \) in this exercise even though later releases of DrugBank are available; v3.0 (DBv3) and v4.2 (DBv4.2) Knox et al. (2011). DBv3 contains additional drugs, targets and their interactions to those already contained in \( Dat \) (Table 1) and 8,768 additional D-T interactions to those found in \( Dat \). Of these interactions, 2,919 involve drugs and targets that are present in \( Dat \), but the interaction relationship had not yet been defined (i.e. the D-T interaction had not been annotated in DBv2.5). In this work, we refer to these 2,919 interactions from DBv3 as being 'relevant'. These relevant interactions are represented in the set \( DBv3Rel \) (see Equation S) and were used to derive a query set of semantic subgraphs that were in turn used to mine \( Dat \). DBv4.2 was then used as a reference to validate the new repositioning opportunities identified through the mining of \( Dat \).

**Semantic subgraphs inferring drug target interactions** Semantic subgraphs can be derived through manual exploration of the graph and by reference to known repositioning examples. However, in this work, in order to exhaustively test the DReSMIn algorithm, we derived an automated method for producing a set of semantic subgraphs that would be appropriate for systematically mining for new D-T interactions.
In order to produce such a set, we extracted the portions of the network in Dat that contained drugs and targets from the 2,919 D-T interactions whose interaction was annotated later in DBv3Rel. To extract the subnetworks, each drug and target pair was identified in Dat and the subnetwork represented by the shortest path between them was extracted as a semantic subgraph (Fig. 6). To identify the shortest semantic subpaths, Dat was converted to an undirected graph and a Java implementation of Dijkstra’s shortest path algorithm Dijkstra (1959), from the JGraphT library used. On carrying out this semantic subgraph identification exercise 194 different subgraphs with $|V| < 10$ were found to cumulatively identify more than 95% of the D-T interactions in DBv3Rel and were used as a reference set for D-T inference using DReSMin as described below.

**Inference of novel drug-target interactions** The 194 semantic subgraphs were used as queries to search Dat using DReSMin to test the ability of the algorithm to identify D-T interactions in Dat that had not yet been annotated in DBv2.5 (but are present in DBv3). DReSMin was used to identify subgraphs in Dat that were similar to the query set of semantic subgraphs and therefore with the potential to be indicative of novel D-T interactions and ultimately aid in the identification of drug repositioning opportunities. After an exhaustive search of Dat with the 194 semantic subgraphs a set of mappings (or instances) of each subgraph was identified. Semantic subgraphs were scored on their ability to identify D-T interactions captured in DBv3Rel (using Equation 3), with these scores ranging from 0.0 to 0.06589 (Table S4). A single D-T interaction can be inferred by mappings of more than one query semantic subgraph, thus adding confidence to the prediction that a D-T interaction exists. Therefore, in order to rank the D-T interactions in terms of confidence, the scores assigned by all query semantic subgraphs that produced a mapping containing a potential D-T interaction were summed (using Equation 4). The $\Sigma R_q$ of the scores of all 194 query semantic subgraphs was 0.9499 (Fig. S5) and so inferred D-T interaction scores contained within mappings could potentially, range from 0.0 to 0.9459. The top ten performing subgraphs, and a larger illustrative subgraph, are shown in Fig. 7.

A search of Dat with the set of 194 semantic subgraphs described above resulted in 906,152,721 mappings. These mappings now captured the potential drug target interactions in the structure of the mapping subgraph. The 906,152,721 mappings predicted 9,643,061 D-T interactions that were ranked as described above. Unsurprisingly, we identify the interactions from DBv3Rel that were used to create the semantic subgraphs. Importantly, however, these interactions score highly, which indicates that a single interaction was identified by multiple semantic subgraphs. The D-T interactions from DBv3Rel consistently scored better and ranked higher than the unsupported inferred associations (Fig. 8A and Fig. 8B). We also observe that the D-T interactions subsequently annotated and captured in DBv3Rel are identified by two fold the number that infer D-T associations not annotated and present in DBv3Rel (Fig. 8C).

However, in order to quantify the predictive power of DReSMin we examined how many of the high scoring D-T predictions were subsequently annotated in DBv4.2. DBv4.2 contains 333 interactions not captured in DBv3 or Dat. In this work, these interactions are represented in the set DBv4Rel (see Equation 6). These 333 new interactions had not been used to construct the semantic subgraphs used for searching Dat. Of the 333 D-T interactions captured in DBv4Rel, 309 were successfully identified (94%). We also observed high ranking and scoring of 309 D-T interactions from DBv4Rel that were successfully identified by DReSMin (Fig. 8D and Fig. 8E). The average number of semantic subgraphs that have mappings inferring the 309 annotated D-T associations captured in DBv4Rel is increased >4 fold in comparison to the number of semantic subgraphs that produce mappings that infer interactions not captured in DBv4Rel (Fig. 8F).

$$DBv4Rel = \{(DatRel \cup \text{Unique(DBv4.2)}) \cap DBv3Rel \mid d \in DatRel(d) \land t \in DatRel(t)\} \quad (6)$$

Looking in more detail at the top 20 inferred D-T interactions (Table 2) we see 12 different drugs and eight targets. Drugs include: three antihypertensive agents (Verapamil, Mibebradil, and Bepridil); three phenothiazine antipsychotic agents (Promazine, Perphenazine and Thioridazine); three atypical antipsychotic agents (Propiomazine, Clozapine and Quetiapine); two anticonvulsants (Zonisamide and Levetiracetam) and one antiarrhythmic adrenergic beta-antagonist (Propranolol). Of the 12 drugs captured in the top ranked inferred D-T interactions, the average number of D-T interactions captured in

\[\text{http://jgrapht.org}\]
Indication in 11). Of the remaining 16 involved 11). Of the remaining 16

Within Dat 427 with CAC1C. Although Verapamil is already used to treat hypertension, and the inferred D-T interaction

Dat is ~13, with the average number for all compounds being closer to three. The compounds present

in the top 20 inferred D-T interactions are well studied and annotated and are thus highly connected

in Dat. Targets include four voltage-dependant calcium channels (VDCC) and four G-Protein coupled

receptors (GPCR). VDCCs display selective permeability to calcium ions which enter a cell, and alter

a channel’s properties, through the pore which is formed by the α 1 subunit. We can see that three

sub-types of VDCC are represented in Table 2, being: L-type (CAC1C and CAC1D); P/Q Type (CAC1A)

and N-type (CAC1B). Members of the GPCR superfamily in Table 2 include receptors activated by the

neurotransmitters: serotonin (5HT7R and 5HT2B); epinephrine (ADA1A) and dopamine (DRD1).

The ability of DReSMin to predict novel D-T interactions was compared to the state-of-the-art ligand-

based method from ChEMBL. We first examined how many D-T interactions were predicted by both

methods (co-prediction) using interactions captured in the sets DReS, Chem1 and Chem10. Unsurprisingly,

due to the fact that the methods use different approaches, only 10% of the top x D-T interactions inferred

by DReSMin are found in the top x D-T interactions predicted by ChEMBL models (Fig.9A and Fig.9B).

More interestingly is the fact that DReSMin successfully infers >20 % more of the knowns from DBv4Rel

than ChEMBL, for both models (Fig.9C and Fig.9F). We found that DReSMin is able to rank the known

D-T interactions more effectively than ChEMBL, with a mean ranking position of known D-T interactions

from DBv4rel of ~25,000, as opposed to the 50,000 achieved by ChEMBL. We must recognise the fact

that the semantic subgraphs used during this work were derived using DrugBank data and the ChEMBL

models trained on ChEMBL data.

Completing the drug-target-disease pathway The highest ranked D-T interaction identified by DReSMin

receiving a score of 0.49211, is supported by the literature and therefore known to the scientific commu-
nity. This D-T interaction is between one of the antiarrythmic calcium blockers, Verapamil, and CAC1C.

Within Dat eight indications are associated to Veparamil and 12 diseases associated to CAC1C. One

indication, hypertension, shares a has_Indication association with Verapamil and a involved_in association

with CAC1C. Although Verapamil is already used to treat hypertension, and the inferred D-T interaction

already known, we see how DReSMin may be used to help understand the molecular mechanism of a

drug and thus complete the ’drug-target-disease’ pathway. Understanding the molecular mechanisms

of drugs can only aide the identification of repositioning opportunities. In Fig. 10 we see examples of

unsupported, and therefore novel, DReSMin inferred D-T interactions that also increase understanding of

the molecular mechanisms involved in a drugs ability to treat a disease. Like Verapamil, Bepridil is also a

calcium channel blocker with known antiarrhythmia activities. Used as a treatment of hypertension, we

can see in Table 2 an inferred D-T association involving Bepridil and CAC1C. Bepridil is one of the two

drugs from Table 2 that have been withdrawn from the market due to safety concerns. For this reason it

is not a strong candidate to be repositioned, however, via the inferred association we are able to better

understand the molecular mechanism of the drugs ability to treat hypertension (Fig. 10A).

In Dat we see three indications for Quetiapine (Psychotic Disorders, Bipolar Disorders and Autistic

Disorders) and three involved_in associations involving 5HT7R (Schizophrenia, Pain and Muscular

Diseases). Although not captured in Dat, Quetiapine is approved for the treatment of Schizophrenia. By

integrating this knowledge with Dat and our inferred associations we can complete another drug-target-
disease pathway (Fig.10C). Although Schizophrenia, along with many other diseases, is classified as a

psychotic disorder, we see how inferred knowledge can enable better understanding of drug-target-disease

pathways in more specific, as opposed to broader, disease areas.

Propranolol One inferred D-T interaction in Table 2 involves the antiarrythmic adrenergic beta-

antagonist, Propranolol, and the G protein-coupled receptor DRD1. In Dat we capture 12 indications

for Propranolol and 17 disease associations for DRD1, with one indication, Hypertension, in both (Fig.

11). Of the remaining 16 involved_in associations involving DRD1 three of the diseases represent known

off-label indications for Propranolol being: Bipolar disorders; Schizophrenia, Alcoholism and as a

non-stimulant treatment for ADHD Gobbo and Louzã (2014). The remaining 12 diseases present and

support some interesting repositioning opportunities/studies of potential repositioning opportunities for

Propranolol.

Looking at potential indications of Propranolol that are currently being investigated by the scientific

community we see three that are supported by our work. Dat contains an association between DRD1 and
cocaine related disorders, with multiple clinical trials being undertaken to analyse the use of Propranolol

Table 2. DReSMin was executed using the 194 semantic subgraphs that represented the shortest
path between the drugs and targets captured in the relevant associations

<table>
<thead>
<tr>
<th>Drug (DrugBank ID)</th>
<th>Type, Category</th>
<th>Inferred Target (Uniprot ID)</th>
<th>Evidence</th>
<th># Subs</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verapamil (DB00661)</td>
<td>SM, AP</td>
<td>CAC1C (Q13936)</td>
<td>Y</td>
<td>85</td>
<td>0.49211</td>
</tr>
<tr>
<td>Mibefradil (DB01388)</td>
<td>SM, WI</td>
<td>CAC1A (O00555)</td>
<td></td>
<td>74</td>
<td>0.44378</td>
</tr>
<tr>
<td>Mibefradil (DB01388)</td>
<td>SM, WI</td>
<td>CAC1B (Q00975)</td>
<td></td>
<td>59</td>
<td>0.43097</td>
</tr>
<tr>
<td>Promazine (DB00420)</td>
<td>SM, AP</td>
<td>ADA1A (P35348)</td>
<td>Y</td>
<td>117</td>
<td>0.39090</td>
</tr>
<tr>
<td>Quetiapine (DB01224)</td>
<td>SM, AP</td>
<td>5HT7R (P34969)</td>
<td></td>
<td>61</td>
<td>0.38779</td>
</tr>
<tr>
<td>Propiomazine (DB00777)</td>
<td>SM, AP</td>
<td>5HT7R (P34969)</td>
<td></td>
<td>69</td>
<td>0.38774</td>
</tr>
<tr>
<td>Verapamil (DB00661)</td>
<td>SM, AP</td>
<td>CAC1A (O00555)</td>
<td>Y</td>
<td>78</td>
<td>0.38436</td>
</tr>
<tr>
<td>Verapamil (DB00661)</td>
<td>SM, AP</td>
<td>CAC1B (Q00975)</td>
<td>Y</td>
<td>64</td>
<td>0.38180</td>
</tr>
<tr>
<td>Mibefradil (DB01388)</td>
<td>SM, WI</td>
<td>CAC1D (Q5QSC4)</td>
<td></td>
<td>52</td>
<td>0.37525</td>
</tr>
<tr>
<td>Perphenazine (DB00850)</td>
<td>SM, AP</td>
<td>5HT7R (P34969)</td>
<td></td>
<td>86</td>
<td>0.37383</td>
</tr>
<tr>
<td>Thioridazine (DB00679)</td>
<td>SM, AP</td>
<td>5HT7R (P34969)</td>
<td></td>
<td>75</td>
<td>0.36830</td>
</tr>
<tr>
<td>Promazine (DB00420)</td>
<td>SM, AP</td>
<td>5HT7R (P34969)</td>
<td></td>
<td>75</td>
<td>0.36824</td>
</tr>
<tr>
<td>Propranolol (DB00571)</td>
<td>SM, AP, IN</td>
<td>DRD1 (P21728)</td>
<td></td>
<td>96</td>
<td>0.36084</td>
</tr>
<tr>
<td>Zonisamide (DB00909)</td>
<td>SM, AP, IN</td>
<td>CAC1B (Q00975)</td>
<td></td>
<td>50</td>
<td>0.35478</td>
</tr>
<tr>
<td>Levetiracetam (DB01202)</td>
<td>SM, AP, IN</td>
<td>CAC1B (Q00975)</td>
<td>Y</td>
<td>50</td>
<td>0.35478</td>
</tr>
<tr>
<td>Thioridazine (DB00679)</td>
<td>SM, AP</td>
<td>5HT2B (P41595)</td>
<td></td>
<td>107</td>
<td>0.35036</td>
</tr>
<tr>
<td>Clozapine (DB00363)</td>
<td>SM, AP</td>
<td>5HT7R (P34969)</td>
<td>Y</td>
<td>64</td>
<td>0.34799</td>
</tr>
<tr>
<td>Propranolol (DB00571)</td>
<td>SM, AP, IN</td>
<td>ADA1A (P35348)</td>
<td></td>
<td>84</td>
<td>0.34663</td>
</tr>
<tr>
<td>Bepridil (DB01244)</td>
<td>SM, AP, WI</td>
<td>CAC1C (Q13936)</td>
<td></td>
<td>77</td>
<td>0.34610</td>
</tr>
<tr>
<td>Levetiracetam (DB01202)</td>
<td>SM, AP, IN</td>
<td>CAC1A (O00555)</td>
<td></td>
<td>63</td>
<td>0.34605</td>
</tr>
</tbody>
</table>

Notes: See Table S4 for a detailed scoring of the 194 semantic subgraphs. Scores are used as a ranking method for inferred interactions. Of the 20 interactions ranked highest by DReSMin, six were found in DBv3; having literature supporting their existence. For drug Type and category: SM = small molecule, AP = approved, IN = investigational and WI = withdrawn. Scores are to 5 decimal places. # Subs refers to the number of semantic subgraphs that inferred the D-T interaction, with the maximum being 194.

A trial looking at the use of Propranolol as a treatment for Autism is also, at the time of writing, recruiting of Missouri-Columbia (2015). Finally, a clinical trial has also been undertaken to analyse the effects of using Propranolol as a treatment for drug-Induced movement disorders Corp. (2014). We can see that our inferred D-T interactions allow us to predict repositioning opportunities that agree with the community.

DISCUSSION

In this paper, we explore the concept of using semantic subgraphs as a way of inferring novel D-T interactions with the aim of using them to identify drug repositioning leads. We present and formalise semantic subgraphs, showing how they may present patterns indicative of drug repositioning opportunities. By employing a novel approach to reducing the target graph size prior to a search, and by breaking larger semantic subgraphs to a set of smaller subgraphs, DReSMin significantly improves on the performance of a purely topological approach to pattern matching. We also show how the approach can be used to automate the identification of novel D-T interactions in an integrated semantic network, with the aid of historical data. This real-world problem often requires searching for semantic subgraphs where \(|V(S)| > 4\).

The optimisations we have presented here makes searching for instances of these complicated subgraphs computationally tractable and scalable. We have shown an example of the application of DReSMin which highlights the potential of the approach.

When comparing DReSMin to other state-of-the-art drug-target prediction methods we observe an average co-prediction of 10%. The likely reason for the low co-prediction value is due to the fact that they are two different approaches to the task. Despite the differing approaches we were able to directly compare and contrast the results and found DReSMin to outperform the ChEMBL models at inferring annotated DrugBank D-T interactions. Considering DReSMin is a general algorithm, not specifically developed for
the inference of D-T interactions, this highlights its potential. Although the semantic subgraphs used to search Dat were derived from the shortest paths between a drug and target from D-T interactions in DBv3, these interactions were inferred, on average, by around 40 different semantic subgraphs. This is in contrast to the 15 semantic subgraphs that inferred D-T interactions not captured in DBv3. Again this validates the approach we employed during this work. Annotated D-T interactions were not only captured by the semantic subgraph derived from the semantic shortest path between their drug and target but also by many more.

Although DReSMin at present scores semantics based purely on the most abstract form of types, it could be beneficial to include scoring metrics based on node and edge attributes, and the data-sources from which they are retrieved. For example, during the process of data integration it would be useful to consider dataset quality during the construction of the integrated graph and apply annotations that indicate a measure of confidence in a given interaction. To this end we are currently developing a new integrated dataset that will allow provenance and data to reliability to be scored during a search. This modification will allow the scoring of semantic subgraphs to be not only topological and semantic but also based on the reliability of the source of each element.

In the approach described here semantic subgraphs are derived from only the node types and edge types that fall directly on the semantic shortest path between a drug and a target. In order for a semantic subgraph capture even more functional detail it may be beneficial to expand the view that the subgraph takes of its immediate neighbourhood. To this regard we are currently considering extending semantic subgraphs to include nodes that interact with those in the semantic shortest path at a particular depth.

Although we present an exhaustive automated approach it is also worth noting that semantic subgraphs can be drawn from real life repositioning examples via manual curation. Manually developing semantic subgraphs is time consuming, however they may allow for more the creation of more accurate representations of a functional module representative of a drug repositioning opportunity. We hope to create a library of semantic subgraphs curated from real world examples of repositioned drugs and compare the accuracy and efficiency to the semantic subgraphs developed during this work.

With regard to the mining algorithm, as new graph mining frameworks emerge with efficient graph searching algorithms (e.g. Neo4J), it may be possible to exploit these built in algorithms to implement sections of the approach we describe here. However, necessarily, the nature of these implementations will depend on the specific graph database.

We have demonstrated that our algorithm may be used to infer D-T interactions, however, like all in silico approaches to analysing in vivo and in vitro systems the accuracy is limited; overly simplified settings innately struggle to reflect real-life problems. Our approach, unlike many other computational approaches to drug repositioning, is not limited to the inference of D-T interactions. Semantic subgraphs may be designed to infer relations between any conceptClasses in a dataset and can be used to infer a drugs indication, mode of action, side effect and more. We believe that the systems biology approach that we describe here will allow for a more accurate, holistic, systematic approach to drug repositioning.

SUPPLEMENTAL INFORMATION

Uploaded separately.

ADDITIONAL INFORMATION AND DECLARATIONS

Dr. Hannah Tipney and Peter Woollard are employed by GlaxoSmithKline.

ACKNOWLEDGEMENTS

We would like to thank Dr Philipe Sanseau of GlaxoSmithKline Research & Development Ltd for his guidance during this project. We also extend our gratitude to the ICOS Writing Group and acknowledge Bioinformatics Support Unit, both of Newcastle University for their discussions and manuscript input.

REFERENCES


Figure 1. An example of a simple semantic subgraph (B) is derived from the repositioning of Chlorpromazine (A). Chlorpromazine is marketed as a non-sedating tranquilliser, but is also known to be effective as an antihistamine Rukhadze et al. (2001) and so in A a relation is inferred between Chlorpromazine and the Histamine H1 receptor (dashed line).
Figure 2. Overview of the DReSMin algorithm developed for the detection of semantic subgraphs indicative of repositioning opportunities
Figure 3. Subgraph split procedure takes an initial semantic subgraph (Q) and produces two smaller semantic subgraphs (D₁ and D₂) using all vertices (v) from (Q). The overlapping node (ON) is identified in Q (v₃) and used as the overlapping node in both D₁ and D₂. The two most distant vertices in Q are then identified (v₁ and v₆) and vertices in the path between these and ON added to the corresponding graphs (D₁ and D₂). We also see that |V(D₂)| > 3 and so a second call is made to graph split giving us D₂₁ and D₂₂.
Figure 4. A subsection of the Ondex in silico drug discovery dataset metagraph shows how different concept Classes (e.g., Compound & Target) interact via relation Types (binds_to).
Figure 5. Overview of algorithm performance with semantic subgraph (Q) queries nodeset |V(Q)| ranging from 3-6. Graphs show each step of the algorithm reducing search time. Abbreviations: $V_{max}G =$ initial candidate set is chosen using the $v_t$ of the node displaying the highest connectivity in $Q$, COMP = compound makes up the initial candidate set, CW = closed world check implemented, SDC = semantic distance calculator used during search, GP = semantic graph prune step implemented, GS = graph split used. $5^a$ (all steps plotted for |V(Q)| = 5) and $5^b$ (focuses on GP and GS of |V(Q)| = 5). Algorithm Performance (bottom right) shows the best approach for each semantic subgraph size.
Figure 6. Semantic subgraphs were derived from the semantic shortest paths between a drug and a target pair captured in DrugBank v3. A) shows a drug-target interaction captured in DBv3 made up of a drug (D1) and a target (T2) captured in our network, Dat. In order to create semantic subgraphs D1 and T2 are identified in Dat (highlighted in green in B) and the semantic shortest paths between the two nodes calculated (highlighted in red in B). Finally all semantic node types and edge types that fall on the semantic shortest path are used to create a query graph (C). Note: Dashed red line represents the inferred binds_to relations, squares represent compounds, circles targets, diamonds proteins and octagon diseases. For relation types: bi_to = binds_to, sim = similar_to, h_s_s = has_similar_sequence.
Figure 7. Examples of semantic subgraphs drawn from the semantic shortest paths. Q1-Q10 are drawn from the semantic shortest paths that represented the shortest path between the greatest number of D-T interactions in DBv3Rel and Q108 is an example of a more complex semantic subgraph. Note: Dashed red lines represent the inferred binds_to relations, squares represent compounds, circles targets, diamonds proteins and octagon diseases. For relation types: bi_to = binds_to, sim = similar_to, h_s_s = has_similar_sequence, ma_tr = may_treat, inv_in = involved_in and is_a = is_a.
Figure 8. Validation of inferred D-T associations with known D-T associations from DBv3 and DBv4.2. A, B and C show how DReSMiN identifies and ranks the 2,919 known interactions from DBv3 when searching Dat. D, E and F show how DReSMiN identifies and ranks the 333 known interactions from DBv4.2. For A and D hypergeometric distribution of inferred knowns were calculated using the scores of the validated associations. For B and E hypergeometric distribution of inferred knowns were calculated using the ranked position of the validated interactions. C and F show the number of semantic subgraphs that inferred knowns in comparison to the number of semantic subgraphs that inferred novel interactions. Note: Blue line shows the highest scoring semantic subgraph; all scores above this line are definitely inferred by > one semantic subgraph.
Figure 9. DReSMin inferred D-T associations in comparison to those inferred using the ligand-based similarity models provided by ChEMBL. Top graphs (A, B and C) show comparison to those using the 1uM model from ChEMBL and bottom graphs (D, E and F) show comparison with the 10uM model. A and D show the %age crossover between the top ranked x associations from each method for each drug. B and E show the comparative ranking of the 2,919 known D-T interactions from DBv3Rel. C and F show the comparative ranking of the 333 known D-T interactions from DBv4Rel. In B, C, E and F red diamonds show the mean ranking and numbers in red show the number of knowns captured by each method. Only associations inferred by the 2,223 drugs with a mapping between DrugBank and ChEMBL are included.
Figure 10. Drug-target-disease pathways completed via inferred D-T associations. Data presented is extracted from Dat with one association extracted from literature. Note: Dashed lines represent the inferred binds_to relations, zig-zag lines represent has_Indication relation not captured in Dat and extracted from literature, squares represent compound, circles target and octagon diseases.
Figure 11. Diseases associated with Propranolol and DRD1. Drug-disease has Indication associations involving Propranolol and gene-disease involved in associations were extracted from Dat. Note: Dashed lines represent the inferred binds to relations, squares represent compound, circles target and octagon diseases.