Supplementary Materials.

Appendix A. Data collection and processing using semantic “mashup” knowledge base

RDF KN, used by RIIG formalism, is induced from a Semantic “Mashup” Knowledgebase (SMKB). SMKB is an integrated knowledge base that combines information from a number of publicly available biomedical knowledge bases (see [37], [41] for data integration studies). RIIG utilizes RDF, OWL and SPARQL [38] to integrate data from heterogeneous sources and to retrieve relevant molecular mechanisms of different types (e.g. metabolic, signal transduction pathways and protein-protein interaction). The overall schema of SMKB is presented in Figure 1.

![Figure 1. Semantic Mashup Knowledgebase.](image)

The SMKB was constructed by importing the instance RDF data from EntrezGene [71] (~330,000 RDF triples), Reactome [20], [42] (~700,000 RDF triples), HPRD [18], [35] (~40,000 RDF triples), [20], [54] UniprotKB (3,300,000 RDF triples), and NCI Nature Pathways [18], [33] (~240,000 RDF triples), BioCarta [31] (~900,000 RDF triples) into the Oracle’s RDF store [78] totaling to more than 4,700,000 RDF triples of information. RDF data from GO [79] and BioPAX [47] ontologies were also entered into the SMKB. The data that were not available in RDF format were converted into RDF representation using n-ary relation modeling and custom ontologies. For instance, in context of our Systems Pathology Informatics Toolbox (SPIRIT) framework, SMKB is intended to answer therapeutic related queries. To be able to run such queries, we have utilized drug information from COSMIC database [80]. To do so we have converted COSMIC data into RDF format and developed a supporting ontology.

The structure of an RDF knowledge base formalized by RIIG method is largely based on RDF specifications [81]. An RDF knowledge base is composed of ontologies, RDF resources and RDF
properties. Ontologies consist of concept classes $C$ and relations $R$. There are two types of RDF resources: RDF object resources and RDF literals. RDF object resources are specific objects of ontological concept classes and are defined by Uniform Resource Identifiers (URIs). For instance, an RDF object resource “http://pid.nci.nih.gov/biopax#pid_p_100189_deathpathway” is an object of an ontological concept class “pathway”. Other examples of concept classes include such biological entities as “gene”, “protein”, “protein complex”, “interaction”, and “biochemical reaction”. RDF literals are represented by textual and numerical constants, which provide additional information to RDF object resources. For instance, RDF object resource “http://pid.nci.nih.gov/biopax#pid_p_100189_deathpathway” can have a description “induction of apoptosis through dr3 and dr4/5 death receptors”, which is an RDF literal. RDF properties are instantiations of ontological relations $R$ and specify inter-relationships among RDF resources. Examples of RDF properties are “COFACTOR” and “PATHWAY-COMPONENTS”. The formal definitions of the above concepts can be found in SM (see Definitions 1–5).

One of the challenges of creating RDF “mashup” was that some data were not available in RDF N3 format. We used n-ary relations modeling to convert the data into RDF triples. An example of this methodology for transforming tabular drug-protein data into RDF is shown in Figure 2 below.

Given drug-protein interactions’ p-values in tabular format, each interaction will be modeled as a set of three RDF triples: (1) [Drug ID] [participates] [Drug-Protein Interaction ID], (2) [Protein ID] [participates] [Drug-Protein Interaction ID], and (3) [Drug-Protein Interaction ID] [hasPValue] [p-value] (p-value is shown as “0.05”^^xsd:float” in the rightmost box in Fig.17). Higher dimensionality N-ary relations will be represented similarly with “inter-entity” RDF class (e.g. “Drug-Protein Interaction” in the example above) having additional RDF type or object properties.
Appendix B. RIIG formalism

**Definition 1 (ontology).** An ontology \(o = (C, R)\) is a set \(C\) of concept classes along with a set \(R\) of relations among those concept classes. In this work, concept classes \(C\) include such biological entities as “gene”, “protein”, “protein complex”, “interaction”, “biochemical reaction”, and “signal transduction pathway”. Examples of relations \(R\) among concept classes \(C\) are “COFACTOR” and “PATHWAY-COMPONENTS”.

**Definition 2 (RDF object resource).** An RDF object resource \(or_o\) is an object of some concept class \(c \in C\) of the corresponding ontology \(o\). For example, for a concept class “protein” of ontology \(o\) there can be an RDF object resource \(AKT1\).

**Definition 3 (RDF literal).** An RDF literal \(l\) is either a textual or a numerical constant.

**Definition 4 (RDF object property).** An RDF object property \(op_o\) is an instantiation of some relation \(r \in R\) from the corresponding ontology \(o\). For example, there could be several RDF object properties “COFACTOR” in an RDF knowledgebase, which are instantiations of relation “COFACTOR” of ontology \(o\).

**Definition 5 (RDF data property).** An RDF data property \(dp_o\) is an instantiation of some relation \(r \in R\) from the corresponding ontology \(o\). For example, there could be several RDF data properties “NAME” in an RDF knowledgebase, which are instantiations of relation “NAME” of ontology \(O\).

**Definition 6 (RDF triple).** An RDF triple \(t\) is a three-place relation of the form \(<s, p, o>\), where \(s\), \(p\) and \(o\) are called subject, predicate and object respectively. In any RDF knowledge base \(KB_{RDF}\) there can be only two types of RDF triples, namely \(<or_o, op_o, or_o>\) and \(<or_o, dp_o, l>\), where \(or_o \in OR_o\), with \(OR_o\) being the set of all RDF object resources of \(KB_{RDF}\), \(op_o \in OP_o\), with \(OP_o\) being the set of all RDF object properties of \(KB_{RDF}\) and \(dp_o \in DP_o\) with \(DP_o\) being the set of all RDF data properties of \(KB_{RDF}\).

**Definition 7 (RDF knowledge base).** Given a set of RDF resources \(S_{RS}=(OR_o, L)\), consisting of a set of RDF object resources \(OR_o\) and a set of literals \(L\), a set of RDF properties \(S_P=(OP_o, DP_o)\), consisting of a set of RDF object properties \(OP_o\) and a set of RDF data properties \(DP_o\), with sets \(OR_o\), \(OP_o\), \(DP_o\) instantiated from the set of ontologies \(S_o\), an RDF knowledgebase \(KB_{RDF}\) is defined as follows: \(KB_{RDF}=\{t=<s, p, o> : t \in T; s, o \in S_{RS}; p \in S_P\}\), where \(T\) is the set of all RDF triples of \(KB_{RDF}\).

**Definition 8 (RDF knowledge network).** Given an RDF knowledge base \(KB_{RDF}\) from Definition 7 and isomorphic mapping functions \(fn(\cdot)\) and \(fe(\cdot)\), an RDF knowledge network is a graph \(G_{RDF}=(N_{RDF}, E_{RDF})\) with a set of nodes \(N_{RDF}\) and a set of edges \(E_{RDF}\), for which the following holds:

\[
\forall t = <or_o^{Subj}, op_o^{Pred}, or_o^{Obj}> \in T, \text{where } or_o^{Subj}, or_o^{Obj} \in OR_o, op_o^{Pred} \in OP_o: \\
\exists n_i, n_j \in N_{RDF}, e(n_i, n_j) \in E_{RDF}: n_i = f_n(.or_o^{Subj}), n_j = f_n(or_o^{Obj}), e = f_e(op_o^{Pred}),
\]

where \(e(x, y)\) denotes an edge from node \(x\) to node \(y\).
**Definition 9 (RDF protein).** Given the RDF knowledge base $KB_{RDF}$ from Definition 7, an RDF protein, or simply protein, $pr$ is an RDF object resource that an object of the concept class “protein” of some ontology $o \in S_o$.

**Definition 10 (RDF protein complex).** Given the RDF knowledge base $KB_{RDF}$ from Definition 7, an RDF protein complex $prc$, or simply protein complex, is an RDF object resource that an object of the concept class “protein complex” of some ontology $o \in S_o$ such that $prc=\{pr_i\}$, $i=1...n$ for some $n$ RDF proteins $pr_i$.

**Definition 11 (RDF pathway).** Given the RDF knowledge base $KB_{RDF}$ from Definition 7, an RDF pathway $pw$, or simply pathway, is an RDF object resource that an object of either concept class “signal transduction pathway” or the concept class “metabolic pathway” of some ontology $o \in S_o$ such that $pw = \{pr_{i}, pr_{c_{j}}, pw_{k}\}$, $i=1...m$, $j=1...n$, $k=1...l$ for some $m$ proteins $pr_i$, some $n$ protein complexes $pr_{c_{j}}$ and some $l$ pathways $pw_{k}$, which are also called sub-pathways of pathway $pw$.

**Definition 12 (RDF interaction).** Given the RDF knowledge base $KB_{RDF}$ from Definition 7, an RDF interaction, or simply interaction, $ia$ is an RDF object resource that an object of the concept class of some ontology $o \in S_o$ or its subclasses that reflects biological interaction.

**Definition 13 (active state).** Let $a(\cdot)$ denote a state function for RDF proteins, protein complexes, interactions and pathways, $q(\cdot)$ denote a function that returns the quantity of a protein, and $T_f$ be the minimal quantity for a protein sufficient to carry out its biological functions. Then, for an RDF protein $pr$, we have:

$$a(pr) = \begin{cases} 1 & \text{if } q(pr) \geq T_f \\ 0 & \text{otherwise} \end{cases}$$

(4)

When $a(pr)=1$, we say that protein $pr$ is in active state. Then, for every protein complex $prc$ we have:

$$a(prc) = \begin{cases} 1 & \text{if } \forall pr_i \in prc : a(pr) = 1 \\ 0 & \text{otherwise} \end{cases}$$

(5)

and for all pathways $pw$ we have:

$$a(pw) = \begin{cases} 1 & \text{if } \forall pr_i, pr_{c_j}, pw_{k} \in pw : \prod a(x_m) = 1, \text{where } x_m \text{ is either } pr_i, pr_{c_j} \text{ or } pw_{k} \\ 0 & \text{otherwise} \end{cases}$$

(6)

For an RDF interaction $ia$, the state function $a(\cdot)$ returns 1, and $ia$ is considered to be is in active state, when associated with $ia$ biological event (e.g. phosphorylation, ubiquitination) takes place. Otherwise, $a(\cdot)$ returns 0.

**Definition 14 (i-directional RDF predicates).** Given an RDF knowledge network $G_{RDF}$ from Definition 8, corresponding $KB_{RDF}$ from Definition 7, with a set of object properties $OP_o=\{op_o\}$ for each ontology $o \in S_o$, an isomorphic mapping function $fe(\cdot)$, and state function $a(\cdot)$, $op_o$ is a i-directional RDF predicate, or simply i–directional predicate, if the following holds:

$$\forall e(n_i,n_j) = fe(op_o), e \in E_{RDF}, op_o \in OP_o, n_i,n_j \in N_{RDF}, i \neq j:
\text{if } a(n_i) = 1 \text{ then } a(n_j) = 1$$

(7)
where \( e(x, y) \) is a directed edge from node \( x \) to node \( y \). All other object properties \( OP_\circ \) of \( S_\circ \) for which the above conditions do not hold are considered to be non-i-directional RDF predicates. \( OP_\circ \) is denoted by symbol “\( \rightarrow \)”. The head of the arrow indicates direction of a i-directional predicate \( OP_\circ \). Non-directional predicates are denoted by symbol “—”. Examples of i-directional predicates are object properties “CONTROLLER” and “CONTROLLED” of the BioPax 2 ontology [47]. Examples of non-i-directional predicates are “PATHWAY-COMPONENTS” and “PHYSICAL-ENTITY.”

**Definition 15 (RDF path).** Given an RDF knowledge network \( G_{\text{RDF}} \) from Definition 8, with a set of object properties \( OP_\circ = \{ p \} \) for each ontology \( o \in S_\circ \), \( path_{\text{RDF}}(n_k, n_m) \) is an RDF path from RDF node \( n_k \in N_{\text{RDF}} \) to and RDF node \( n_m \in N_{\text{RDF}} \) and is defined as follows:

\[
\text{path}_{\text{RDF}} (n_k, n_m) = \{ e(n_i, n_{i+1}); e(n_i, n_{i+1}) = f e(p_i), n_i, n_{i+1} \in N_{\text{RDF}}, e \in E_{\text{RDF}}, p_i \in OP_\circ, i = k \ldots m - 1, k < m \}
\]  

(8)

where all \( p_i \) are i-directional or non-i-directional predicates.

**Definition 16 (i-directional RDF path).** Given an RDF knowledge network \( G_{\text{RDF}} \) from Definition 8, \( path_{\text{RDF}}(n_k, n_m) \) is an i-directional RDF path from RDF node \( n_k \in N_{\text{RDF}} \) to and RDF node \( n_m \in N_{\text{RDF}} \) and defined as follows:

\[
\text{path}_{\text{RDF}} (n_k, n_m) = \{ e(n_i, n_{i+1}); e(n_i, n_{i+1}) = f e(p_i), n_i, n_{i+1} \in N_{\text{RDF}}, e \in E_{\text{RDF}}, p_i \in OP_\circ, i = k \ldots m - 1, k < m \}
\]  

(9)

where at least one \( p_i \) is i-directional predicate and all i-directional predicates in \( path_{\text{RDF}}(n_k, n_m) \) point in the direction to the node \( n_m \).

**Definition 17 (RDF parent, child, ancestor and descendant nodes).** Given an RDF knowledge network \( G_{\text{RDF}} \) from Definition 8 and corresponding \( KB_{\text{RDF}} \) from Definition 7, for any two RDF nodes \( n_a, n_b \in N_{\text{RDF}} \), if there exists an RDF edge \( e(n_a, n_b) \), such that the corresponding RDF predicate is i-directional and points at \( n_b \), \( n_a \) is called RDF parent node, or simply parent, of \( n_b \) and \( n_b \) is called RDF child node, or simply child, of \( n_a \). Furthermore, if there exists an i-directional RDF path \( path_{\text{RDF}}(n_a, n_b) \), \( n_a \) is called RDF ancestor node, or simply ancestor, of \( n_b \) and \( n_b \) is called RDF descendant node, or simply descendant, of \( n_a \). If every RDF path from \( n_b \) to any of its ancestors includes the path \( path_{\text{RDF}}(n_a, n_b) \), coincides or is a part of the path \( path_{\text{RDF}}(n_a, n_b) \), then \( n_b \) is called mono-descendant of \( n_a \) and \( n_a \) is called mono-ancestor of \( n_b \). Figure 3 shows examples of mono and non-mono RDF descendant nodes.
If we consider relationship of RDF nodes $n_a, n_b, n_c, n_d, n_e, n_f$, the node $n_b$ is a mono-descendant node only for nodes $n_a$ and $n_f$. The node $n_b$ is not a mono-descendant for the node $n_c$ because there are nodes $n_d$ and $n_e$, which are also ancestors of $n_b$, but the paths to these ancestors neither include, coincide nor a part of the path $path_{RDF}(n_c, n_b)$. The notion of mono-descendants is used to define RDF active path (See Definition 19).

**Definition 18 (RDF collider and non-collider nodes and paths).** For any three RDF nodes $n_i, n_j$ and $n_k$, node $n_i$ is called an RDF collider node, or simply collider, if there exist i-directional RDF paths $path_{RDF}(n_i, n_j)$ and $path_{RDF}(n_j, n_k)$, otherwise $n_i$ is called an RDF non-collider node, or simply non-collider. An RDF path $rp$ that has at least one collider node is called a collider RDF path, or simply a collider path, otherwise $rp$ is called a non-collider path.

**Definition 19 (active RDF path).** Given an RDF knowledge network $G_{RDF}$ from Definition 8 and a set of RDF nodes $S \subseteq N_{RDF}$, an RDF path $rp = path_{RDF}(n_k, n_m)$ from RDF node $n_k \in N_{RDF}$ to and RDF node $n_m \in N_{RDF}$ is considered active in relation to $S$ if any of the following two conditions hold: (i) $rp$ is a non-collider path and $\forall n_i, n_d, n_l \in path_{RDF}(n_k, n_m): n_i \not\in S \land n_d \not\in S$, where $n_d$ is any mono-descendant of $n_i$; (ii) $rp$ is a collider path and $\forall n_i, n_d, n_l \in path_{RDF}(n_k, n_m): n_i \in S \lor n_l \in S$, where $n_i$ is a collider node and $n_l$ is some mono-descendant of $n_i$.

**Definition 20 (RDF association).** Given an RDF knowledge network $G_{RDF}$ from Definition 8, an RDF association of a set of RDF nodes $N_{RDF} = \{n_1, n_2, \ldots, n_m\}$ is defined as follows:

\[
association_{RDF}(N_{RDF}) = (N_{RDF}, \{path_{RDF}(n_i, n_j): n_i, n_j \in N_{RDF}, i \neq j\})
\]  

(10)

An RDF association for a set of RDF nodes $N_{RDF} = \{n_1, n_2, \ldots, n_m\}$ is considered partially realized if $\{path_{RDF}(n_i, n_j)\} \neq \emptyset$ and fully realized if $\forall (n_i, n_j), n_i, n_j \in N_{RDF}, i \neq j: \exists path_{RDF}(n_k, n_m), n_k, n_m \in N_{RDF}, k \neq m$. 

![Figure 3. Mono and non-mono descendant RDF nodes.](image-url)
**Definition 21 (RDF relevance and irrelevance).** Given an RDF knowledge network $G_{RDF}$ from Definition 8 and a set of RDF nodes $S \subset N_{RDF}$, any two RDF nodes, such that $n_a, n_b \in N_{RDF}$ and $n_a, n_b \not\in S$, are said to be conditionally RDF relevant in relation to $S$, expressed as $R(n_a, n_b|S)$, if and only if there exists an active path between these two nodes in relation to $S$. Otherwise they are considered to be conditionally RDF irrelevant, expressed as $\neg R(n_a, n_b|S)$. Nodes $n_a$ and $n_b$ are RDF relevant or RDF irrelevant, expressed as $R(n_a, n_b)$ or $R(n_a, n_b)$ respectively, if the relevance or irrelevance holds for $S=\emptyset$, i.e., unconditionally. When RDF relevance holds conditioned for all RDF nodes in $G_{RDF}$ except for a certain set of nodes $T$, it is expressed as $R(n_a, n_b|\neg T)$. The nodes excluded from the condition list can also be just listed after “\” symbol, i.e., $R(n_a, n_b|\neg n_d, n_e, n_f)$ means that $n_a$ and $n_b$ are RDF relevant in relation to all RDF nodes in $G_{RDF}$ except nodes $n_d, n_e$ and $n_f$. The same applies for conditioning node exceptions in RDF irrelevance statements.

**Definition 22 (Negative RDF Relevance Controller).** Given an RDF knowledge network $G_{RDF}$ from Definition 8, an RDF node $n_c$ is called a Negative RDF Relevance Controller (NRRC) for any two RDF nodes $n_a, n_b \in N_{RDF}$, in relation to $n_c$, if node $n_c$ or any of its mono-ancestors lie on a non-collider path between nodes $n_a$ and $n_b$.

**Definition 23 (Positive RDF Relevance Controller).** Given an RDF knowledge network $G_{RDF}$ from Definition 8, an RDF node $n_c$ is called a Positive RDF Relevance Controller (PRRC) for any two RDF nodes $n_a, n_b \in N_{RDF}$, in relation to $n_c$, if node $n_c$ or any of its mono-ancestors is a collider on a path between nodes $n_a$ and $n_b$.

**Definition 24 (relevance free RDF association).** Given an RDF graph $G_{RDF}$ from Definition 8 and a set of RDF nodes $S \subset N_{RDF}$, an RDF association $association_{RDF}(N_{RDF})$ of a set of RDF nodes $N_{RDF}=(n_1, n_2, \ldots, n_m)$ is called conditionally relevance free RDF association in relation to $S$ if $\forall path_{RDF}(n_k, n_l) \in association_{RDF}(N_{RDF})$, $n_k, n_l \in N_{RDF}$, $k \neq l$, $n_k$ and $n_l$ are conditionally RDF irrelevant in relation to $S$. When $S=\emptyset$, we say that $association_{RDF}(N_{RDF})$ is relevance free RDF association.

**Conjecture 1 (modeling of conditional independencies by RDF relevance).**
Conditional independencies $I(\cdot, \cdot|\cdot)$ of a probabilistic model of the expression of proteins, protein complexes and biological pathways can be modeled by RDF irrelevance statements $\neg R(\cdot, \cdot|\cdot)$ of a RIIG model.

**Conjecture 2 (modeling causal links by RDF-induced influence links).**
Probabilistic causal links can be modeled by the links obtained by the application of probabilistic constraint-based structure learning algorithms where conditional independence statements $I(\cdot, \cdot|\cdot)$ are replaced by RDF irrelevance statements $\neg R(\cdot, \cdot|\cdot)$. Such links are called RDF-induced influence links.

**Definition 25 (RDF-induced influence diagram).** Given an RDF knowledge network $G_{RDF}=(N_{RDF}, E_{RDF})$ from Definition 8 with a set of nodes $N_{RDF}$, a set of edges $E_{RDF}$, and isomorphic mapping function $f_{RDF\rightarrow RIIG}$, a semi-directed graph $D_{RDF}=(N_{RID}, E_{RID}, NRRC, PPRC)$ is an RDF-induced InfluenceGram (RIIG) with a set of nodes $N_{RID}$, a set NRRC of negative and a set PPRC of positive RDF relevance controllers respectively, and a set of edges $E_{RID}$, such that $\forall n_{RID} \subseteq N_{RID} : \exists n_{RID} \subseteq N_{RID} : n_{RID} = f_{RDF\rightarrow RIIG}(n_{RDF})$, and $E_{RID}=(E_U, E_D)$, where $E_U$ is a set if undirected edges and $E_D$ is a set of RDF-induced influence links, with sets $E_U$ and $E_D$ generated by a probabilistic constraint-based structure learning algorithm where conditional independence statements $I(\cdot, \cdot|\cdot)$ are replaced by RDF irrelevance statements $\neg R(\cdot, \cdot|\cdot)$ from $G_{RDF}$ (Conjecture 2), and such
that \( \forall nr \in NRRC : nr = (c_{RID}, a_{RID}, b_{RID}), c_{RID}, a_{RID}, b_{RID} \in N_{RID} \), with \( c_{RID} \) being a negative RDF relevancy controller of \( a_{RID} \) and \( b_{RID} \), and \( \forall pr \in PRRC : pr = (c_{RID}, a_{RID}, b_{RID}), c_{RID}, a_{RID}, b_{RID} \in N_{RID} \), with \( c_{RID} \) being a positive RDF relevancy controller of \( a_{RID} \) and \( b_{RID} \).

**Biological interpretation of RDF graph patterns**

Let us now examine a domain specific interpretation of the *Conjecture 1* as it applies to proteins, protein complexes and biological pathways. We need to take a closer look at how i-directional and non-i-directional predicate edges affect RDF relevance in both types of RDF paths with and without colliders.

There could be several elementary node-edge-node-edge-node patterns, which we call edge junctions, in an RDF path without colliders. First, we will review some of these patterns from Figure 4. All others have either equivalent explanation or can be viewed as a combination of the elementary patterns. For each pattern, we ask if the outer nodes are RDF irrelevant in relation to the middle node.

**Figure 4. An RDF path without colliders.**

**Non-collider path. Pattern 1.** (Figure 4 (a)) In the RDF path from node \( B_1 \) to node \( B_3 \), the middle node \( B_2 \) has adjacent edges that are i-directional predicates (according to some ontology) with their heads pointing away from the node \( B_2 \). The proteins/pathways \( B_1 \) and \( B_3 \) are active when protein/pathway \( B_2 \) is active according to the *Definition 13*. We can therefore conclude that an expression of \( B_1 \) indirectly indicates an expression of \( B_3 \) if we do not consider the state of node \( B_2 \). In other words, we have \( R(B_1, B_3|\emptyset) \), which is equivalent to saying that the path from \( B_1 \) to \( B_3 \) is active because separating set \( S \) is empty and \( B_2 \) is not in \( S \) (*Definition 17*). However, if we know the expression status of \( B_2 \), the knowledge of the status of \( B_1 \) becomes irrelevant to the expression of \( B_3 \). In other words, we have that \( \neg R(B_1, B_3|B_2) \) which is the same as to say that path from \( B_1 \) to \( B_3 \) is inactive because node \( B_2 \) is in the separating set \( S \) (*Definitions 17, 19*). In a probabilistic causal model Pattern1 corresponds to the common cause structure [45], [54].

**Non-collider path. Pattern 2.** (Figure 4 (a)) The RDF irrelevance rules for Pattern 1 will hold for the RDF path from pathway/protein node \( B_3 \) to node \( B_1 \) because of the symmetry.

**Non-collider path. Pattern 3.** (Figure 4b) The RDF path from node \( B_2 \) to node \( B_4 \) has all directed predicates with a head-to-tail edge junction. Node \( B_2 \) is an indirect cause of the expression of \( B_4 \), given that we do not have information about \( B_3 \). Therefore \( B_2 \) and \( B_4 \) are RDF relevant, i.e., \( R(B_2, B_4|\emptyset) \), and the path is active (*Definitions 17, 19*). As soon as we have knowledge of the expression state of \( B_3 \), we will have that \( B_2 \) and \( B_4 \) are conditionally RDF irrelevant in relation to \( B_3 \), i.e., \( \neg R(B_2, B_4|B_3) \) and that the path is inactive (*Definitions 17, 19*), because now \( B_3 \) serves as evidence of expression of \( B_3 \) and there is no need to know the active status of protein/pathway \( B_2 \). This pattern corresponds to the causal trail pattern in a probabilistic causal model.
Non-collider path. Pattern 4. (Figure 4 (b)) The RDF path from protein/pathway node B₄ to node B₂ has all directed predicates with a head-to-tail edge junction. Expression of node B₄ serves as an indirect evidence of expression of B₃, if we have no information of the active status of the protein/pathway B₃. Therefore B₄ and B₂ are RDF relevant, i.e. R(B₄,B₂|⊤). Once knowledge of the active status of B₃ is available, nodes B₄ and B₂ will be considered to be conditionally RDF irrelevant in relation to B₃, i.e., ¬R(B₄,B₂|B₃), since expression of B₃ can now determine the expression of B₂.

Non-collider path. Pattern 5. (Figure 4 (c)) The RDF path from the protein/pathway node B₃ to node B₂ has two types of predicates: i-directional and non- i-directional. Non-directional predicates represent associations between biological entities that do not have causal underpinning implied in Eq. 7 of Definition 13, i.e. active state of one entity do not lead to the active state of another. However, they still indicate other types of relationships between connected entities. Let us consider a case when expression of the protein/pathway node B₄ is correlated with the expression of protein/pathway B₅. That means status of the node B₅ still provides us with some information for our understanding of what happens to the node B₄. We can therefore say that expression of the protein/pathway node B₅ is an indirect correlative evidence of the expression status of the node B₃, if we do not have any information about active state of B₃. We can then state that R(B₃,B₅|⊤) holds and consider the path to be active. With the knowledge of the active status of B₄, we will have that B₃ and B₅ are RDF irrelevant in relation to B₄, i.e., ¬R(B₃,B₅|B₄), and the path will become inactive. If nodes B₄ and B₅ have whole-part relationship, i.e., either protein/pathway B₃ is a part of B₄ or vice versa, node B₅ can be considered to be an indirect container evidence for the node B₃ leading to the same RDF relevance properties of the path as in the case of expression correlation. It can be readily shown that other types of non-i-directional predicate connections will result in the same outcome as for correlation and whole-part relationships.

All other non-collider paths in RDF graphs of protein/pathways KB can either be composed of Patterns 1-5 or be proven in a similar way to possess the same property as Patterns 1-5. For instance if we reverse the direction of predicate between nodes B₃ and B₄ in Pattern 5, the node B₅ becomes an indirect correlative or container cause of the node B₃.

Let us now look at an RDF path with a collider as shown in Figure 5, where an example of such type is an RDF path from protein/pathway node B₁ to node B₄ with a collider node B₂.

![Figure 5. An RDF path with a collider node B₂.](image)

The expression of protein/pathway B₂ can be influenced by both B₁ and B₃. If, for instance, the node B₂ represents an apoptotic pathway and the B₁ and B₃ nodes are apoptotic or anti-apoptotic agents, the active status of B₂ can be a result of either active status B₁ or/and B₃. The expression of agents B₁ and B₃ are RDF irrelevant, i.e., R(B₁,B₃|⊤), if we have no information about the status of B₂. However, giving the status of B₂ enables us to draw some conclusions about the status of B₁ depending on the status of B₃. For example if B₂ is activated, and we know that B₃ is expressed too, there are fewer chances that B₁ is expressed than in the case with B₃ not expressed. Since the expression of either B₁ or B₃ nodes can lead to the expression of B₂, knowing that B₃ is expressed makes expression of B₁ less likely since we have already “explained
away” the expression of $B_2$. In such a case we state that nodes $B_1$ and $B_3$ are RDF relevant in relation to $B_2$, i.e., $R(B_1,B_3|B_2)$.

We have to add that the existence of a non-i-directional predicate between the $B_3$ and $B_4$ nodes does not change the conditional independence of expression of $B_1$ and $B_4$ proteins/pathways given the status of $B_2$. As we have already mentioned, non-i-directional predicates do indicate relevance of expression of one protein/pathway on another (just not causal). Therefore, the status of $B_4$ will “propagate” to $B_3$ and influence the expression of $B_1$ the same way as expression $B_3$ does. In a similar way, since the expression of the $B_5$ protein/pathway is a direct evidence of the expression of $B_2$; hence, we can “substitute” node $B_2$ with the node $B_5$ (or any its direct or indirect evidences) and state our RDF relevancy of $B_1$ and $B_3$ (or $B_4$) given $B_5$ and not $B_2$.

We can therefore conclude that for all non-collider and collider paths RDF relevance and irrelevance statements can be logically used to model conditional independencies among biological entities, which makes Conjecture 1 valid.
Appendix C. RIIG assessment protocol

To assess the ability of RIIG framework to recover a set of predefined influence links from a “synthetic” RDF KB we defined a set of RDF object resources corresponding to EOI, including BioPAX proteins, BioPAX protein complexes and BioPAX pathways. Proteins and protein complexes (biomarker proteins) represent clinical findings that are found to be in active state (see RIIG Definition 13) in the hypothetical clinical case. The pathways represent biological mechanisms against which the clinical findings are run and with which the relationships are to be elucidated.

In the next step, a set of hypothetical RDF-induced influence links is generated uniformly randomly among the EOI. We encode these RDF-induced influence links into the synthetic RDF KB through random generation of RDF colliders (RIIG Definition 18), each of which is composed of two i-directional RDF paths (RIIG Definition 16). The colliders encompass probabilistic “explaining away” mechanism that is at heart of causal links inference. In our case, the colliders result in RDF relevancies (RIIG Definition 21) from which RDF-induced influence links are expected to be inferred by the RIIG implementation. To do that, we generate two i-directional paths uniformly randomly for a pair of predefined influence links with common EOI, so that the paths form a collider. The i-directional paths are modeled using BioPAX 2 ontology. There is a number of different ways in which BioPAX can encode such paths. In the current RIIG assessment protocol we utilize a limited set of encoding patterns for each type of paths, some of which are presented in Figure 6. We model four types of i-directional paths: (i) protein-protein, (ii) protein-pathway, (iii) pathway-pathway, and (iv) pathway-protein. For example, Figure 6 (a) shows how an i-directional path from a protein to a pathway can be encoded using BioPAX RDF entities. There are two RDF triples utilized to encode this relationship: <Control, Controller, Protein> and <Control, Controlled, Pathway>, with Control, Protein and Pathway being BioPAX RDF object resources and Controller and Controlled being BioPAX object properties. Controller and Controlled are i-directional RDF predicates (see Definition 14), with the former having direction toward the subject of the triple and the later having direction toward the object of the triple. Parts of the BioPAX RDF object resources hierarchy and BioPAX object properties hierarchy are shown in Figure 7 and Figure 8, respectively.
Figure 6. BioPAX RDF patterns encoding influence relations.

Figure 7. BioPAX RDF object resources from BioPAX ontology used in RIIG assessment.
Another example is a BioPAX RDF pattern of an $i$-directional path from a pathway to a protein shown in Figure 6 (j). In this case there are 3 RDF triples: <$Control$, $Controller$, $Pathway$>, <$Control$, $Controlled$, $Biochemical$ Reaction$>$, and <$Biochemical$ Reaction$, $Left$, $Protein$>. The RDF predicate “Left” is not $i$-directional. However, the entire path from the $Pathway$ to the $Protein$ is resolved by RIIG as $i$-directional (see RIIG Definition 16), propagating active state from the $Pathway$ to the $Protein$.

After construction of the synthetic RDF KB with the encoded RDF-induced influence links, the RIIG algorithm is run and the number of recovered influence links is noted. We count two types of RDF-induced influence links inferred by RIIG: (i) those that match predefined links and (ii) new RDF-induced influence links that were not predefined.

To compare the performance of the RIIG implementation with a naïve Monte Carlo process we used the following protocol:

1. For a given set of EOIs run RIIG algorithm and save the set of inferred RIIG influence links as RIDILS.
2. For each pair of EOIs from Step 1 compute frequency of their joint appearance in PubMed abstracts using Entrez eUtils [71]. Create a set PMILS of those pairs of EOIs whose PubMed frequency is equal or above a predefined frequency threshold.
3. Count number of matches and mismatches of influence links from RIDILS and PMILS.
4. Sample uniformly randomly pairs of the EOIs from Step1 up to the number of RIIG influence links and save them as a set NMCILS. Count matches and mismatches of influence links from NMCILS and PMILS.
5. Repeat Step 4 100 times and save average counts matches and mismatches for NMCILS and PMILS.
6. Compare corresponding values of matches and mismatches from Step 3 and Step 5 to determine which set of influence links (RIDILS or PMILS) has more matches with PubMed-derived links.
7. Repeat steps 1 through 6 50 times and conduct statistical test for difference of means of matches of RIDILS or PMILS.
Appendix D. PubMed queries in the study of discovering alternative therapeutic targets

We found that STAT1 and STAT3 do indeed interact. The following query returned 191 articles that support activation of STAT1 by STAT3, with [53], [55] being the representative papers.


STAT3 over-expression is an important factor in SCC acting as an oncogenic transcription factor, which regulates a number of pathways important in neoplastic transformation including angiogenesis, cell cycle regulation and apoptosis [16], [27], [56]-[61]. The following query returned 87 articles supporting the involvement of STST3 in SCC.


STAT3 may be down regulated by a wide range of compounds including clinically utilized small molecule inhibitors, numerous experimental drugs, small interfering RNAs, and nutrigenic substances including the green tea extract, EGCG [55], [62]-[66]. The systematic literature review using the following query returned 2059 supporting the inhibition of STAT3.

<("inhibition (psychology)"[MeSH Terms] OR ("inhibition"[All Fields] AND "(psychology)"[All Fields]) OR "inhibition (psychology)"[All Fields] OR "inhibition"[All Fields]) AND ("stat3 transcription factor"[MeSH Terms] OR ("stat3"[All Fields] AND "transcription"[All Fields] AND "factor"[All Fields]) OR "stat3 transcription factor"[All Fields] OR "stat3"[All Fields])>

And finally, native and activated (phosphorylated) STAT3 proteins may be identified with routine immunohistochemical staining with localization in the cytoplasm and nucleus respectively. We executed the following query which returned 1067 supporting papers, with [56], [58], [60], [67]-[70] being representative.

<("stat3 transcription factor"[MeSH Terms] OR ("stat3"[All Fields] AND "transcription"[All Fields] AND "factor"[All Fields]) OR "stat3 transcription factor"[All Fields] OR "stat3"[All Fields]) AND ("immunohistochemistry"[MeSH Terms] OR "immunohistochemistry"[All Fields])>