Integration of literature and data for context-aware model curation: a glioblastoma stem cell case study

by

Emilee Holtzapple

B.S., Ohio University, 2016

Submitted to the Graduate Faculty of the

School of Medicine in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2023

UNIVERSITY OF PITTSBURGH

SCHOOL OF MEDICINE

This dissertation was presented

by

Emilee Holtzapple

It was defended on

March 29, 2023

and approved by

Brent Cochran, PhD, Professor, Department of Developmental, Molecular, and Chemical Biology, Tufts University

Wei Wu, PhD, Senior Systems Scientist, Computational Biology Department, Carnegie Mellon University

Amro El-Jaroudi, Associate Professor, Department of Electrical and Computer Engineering

Thesis Advisor: Natasa Miskov-Zivanov, PhD, Assistant Professor, Departments of Electrical and Computer Engineering, Computational & Systems Biology, and Bioengineering

Committee Chair: Jianhua Xing, PhD, Professor, Department of Computational & Systems Biology

Integration of literature and data for context-aware model curation: a glioblastoma stem cell case study

Emilee Holtzapple, PhD

University of Pittsburgh, 2023

Computational modeling serves many purposes in biomedical research. In addition to understanding mechanisms of normal healthy cell function, computational modeling also provides valuable insights into the mechanisms of disease. In recent years, automated tools for curating computational models of cell function have become more accurate and widespread. However, many obstacles remain for automated modeling in a personalized medicine context. First, many models of disease signaling are merely interaction networks, and do not encode information about rules for dynamic signaling behavior. Additionally, many of these models are not comprehensive enough to make widespread conclusions about the effect of disease control interventions. While automated information retrieval speeds up model curation, machine learning approaches for extracting signaling events from literature are not trustworthy enough to use without human intervention. This dissertation will attempt to address several of these obstacles through a glioblastoma multiforme (GBM) stem cell case study. By utilizing discrete modeling techniques, this GBM model is able to capture the progression of disease at multiple levels of specificity. To address the inaccuracies and natural language processing results I also present a tool for using database results to judge machine-reading. Altogether, the GBM case study and methodology presented in this dissertation can serve as a guide for personalized, automated modeling of disease.

Table of Contents

List of Tables

List of Figures

Acknowledgements

This dissertation would not have been possible without the support and mentorship of Dr. Natasa Miskov-Zivanov. I would also like to thank my dissertation committee for their feedback and advice over the course of my dissertation research. Many thanks to past and present MeLoDy lab members and collaborators - Dr. Cheryl Telmer, Dr. Khaled Sayed, Dr. Kara Bocan, Dr. Casey Hansen, Dr. Yasmine Ahmed, Gaoxiang Zhou, Adam Butchy, and Stefan Andjelkovic. Your guidance and encouragement were invaluable, and greatly contributed to my growth as a scientist.

I am deeply grateful to my friends and family who supported me during graduate school! My parents motivated me and made this dissertation possible. To my younger siblings, Grant, Dana, Cathy, and Samuel, thank you for your unwavering love and support, and for always cheering me on. For Emily and Danielle, for all the love and camaraderie at Ohio University (and beyond)! I would not be the scientist I am today without Mr. Devan Lippincott, who inspired my love of biology. Finally, I need to thank my Pittsburgh friends who made me love the city and kept me company throughout my classes, research, and dissertation writing – Amanda, Cathy, Laura, Trevor, Mel, Leon, Gabe, Jenn, and Spencer.

1.0 Introduction

Computational modeling of interaction networks can provide valuable insight into disease progression and potential interventions. However, model curation can be time-consuming, rely on information that is unavailable or difficult to ascertain. Disease models are usually curated based on one specific subtype or presentation, and are not applicable to other patients or subtypes. A guided, data-informed approach to model curation will improve the accuracy and flexibility of disease models. In this dissertation, I will describe novel methods for curating literature, benchmarking the accuracy of machine-read interactions, and informing model parameters from data. I will also present a glioblastoma multiforme stem cell model curated using said methodology.

1.1 Motivation

Computational modeling of biological signaling cascades is an essential method for understanding the mechanisms of disease [1-3]. Trustworthy models are based on up-to-date literature or data and can be experimentally validated. In return, these models can quickly provide testable hypotheses, and reduce the number of experiments needed to elucidate mechanistic details. However, assembly of a believable computational model usually requires a significant amount of time and mostly manual work. Furthermore, many computational models are a generalization of multiple possible genomic profiles, and they ignore *de novo* or rare mutations [4]. For these

reasons, computational models of cellular signaling or disease are often incomplete or overly general.

One such disease that would benefit from a detailed computational model is glioblastoma multiforme (GBM) [5]. GBM is composed of many subpopulations of tumor cells, which are genetically distinct [6]. These tumors are also able to draw from a pool of cancer stem cells [7]. No models of GBM stem cells currently exist that account for every possible cell line-specific difference, which would be necessary for any extrapolation to potential treatments. Using biological data to inform model parameters would help emphasize the real differences in cell signaling between GBM stem cell lines. I will use the model to provide predictions on how GBM stem cells will respond to certain kinase inhibitors, which are commonly proposed treatments [8]. The model, the parameterization approach, and the kinase inhibition predictions will enhance understanding of GBM stem cells and provide testable hypothesis for effective drug treatments that are based on genetic data.

However, it is difficult to quickly and accurately model all the mechanisms of tumor growth and survival that are necessary for an individual tumor. If any computational model, including the GBM stem cell model, is to be applicable for newly discovered cell lines, or possibly for clinical use, there is a need for improved methods for information extraction and model assembly. While manual curation and parameterization of models of GBM stem cell line is possible, this process is time-consuming, and thus, not practical for any clinical use. For highly heterogenous tumors such as GBM, fast and accurate model curation will require addition of signaling events automatically, based on which genes and proteins are differentially expressed. To extend models without human intervention, machine reading can be used to automatically extracts signaling events from biomedical literature [9]. However, this process is error-prone [10], and so

automated model extension is not feasible with these automatically extracted interactions. The methodology laid out in this thesis, as well as the GBM stem cell model case study, show how both literature and database resources can be integrated with data to curate reliable models of cell signaling.

1.2 Scope

Figure 1 shows how novel methodology introduced in this dissertation (FLUTE, described in Section 3.1) and the network verification and curation tool (described in Section 3.3) are integrated with the DySE workflow. In addition, the new algorithms for initialization of discrete models can be applied to the DySE workflow as well. While there exists a number of machine reading engines that extract information from biomedical literature, and a number of databases with information about biochemical reactions in intracellular networks, the methods outlined in this dissertation combine the information from both sources, and therefore, enable automated model assembly with high confidence information. The described methods are implemented as part of an open source tool, which will be the first tool that not only combines the information from the two sources, but also provides a feedback to improve both machine reading methods and databases.

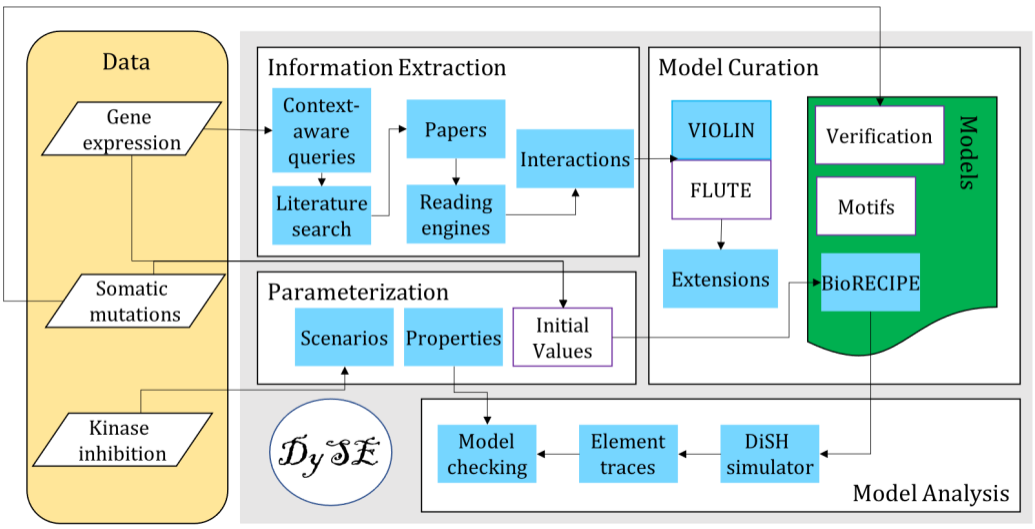


Figure 1. Novel methodology integrated with the DySE workflow.

In this thesis, I propose a method for selecting trustworthy interactions from machine reading output based on manually curated biological data. Interaction databases contain evidence on millions of signaling events [11-13], and comparing machine reading results to these databases can help find trusted interactions. The methodology for selecting only trustworthy interactions from machine reading output will improve both understanding of where machine reading of biomedical literature fails, and how to put biological interaction data to use in the process. Automated extension of new GBM stem cells lines with minimal human intervention will help develop guidelines on how to tailor machine reading results to specific cell lines. It will also greatly reduce the time needed for extension of a model, which is the current protocol for adapting models to new data or information.

To date, the model of GBM stem cell signaling will be the first to account for cell-line specific differences. Furthermore, this novel approach to modeling GBM stem cells will combine the knowledge about the system from the published literature and from experts, with the experimental data, to assemble models that capture the causality in cellular signaling (not only correlations), and that allow for studying dynamic changes of the GBM stem cells in time. Using

gene expression or whole exome sequencing data to parameterize this model can provide predictions for whether a kinase inhibitor will be effective in preventing tumor growth in a specific GBM stem cell line. The results of each parameterized GBM stem cell model will provide novel mechanistic explanations for how GBM stem cells survive certain drug treatments.

2.0 Background

2.1 Why use mechanistic modeling for computational systems biology?

Understanding a disease at a mechanistic level is a complex task, requiring extensive knowledge of how affected genes influence disease progression. Signaling networks are studied to gain more comprehensive understanding of a disease, or to predict potential therapeutic targets [14-16]. In contrast to curating a mechanistic model of disease, training a model on data is only one way to make predictions about disease networks and treatment efficacy [17]. In addition to making predictions, curation of mechanistic signaling models can provide additional benefits [18]. There are still many unknowns about how signaling events are handled within cells, and how these responses differ between individuals [19]. While training classifiers on biomedical data has many applications, it cannot compensate for deficits in knowledge about the underlying system. Curating detailed computational models of signaling networks enhances understanding of cellular signaling cascades.

Existing resources contain a wealth of knowledge that can guide automated technology, benchmark inferred networks, inform understanding of disease mechanisms, and improve patient-specific outcomes. Specifically, there are many literature and data resources that are already curated and accessible by machine or human curators. According to FAIR data principles (Findability, Accessibility, Interoperability, and Reuse of digital assets), data should be computationally accessible whenever possible [20]. In this dissertation, I will present novel methodologies to increase the findability and accessibility of existing literature and data for use in model curation.

2.2 Interaction networks

Interaction networks illustrate the set of biochemical reactions that constitute cellular function. This includes processes such as external signals (for example: stress, nutrient availability, etc.) being conveyed internally to second messengers and eventually, to disrupt or alter gene transcription ([21-23]). These networks can be represented by a graph $G(V, E)$ with a set of nodes V and a set of edges E. Common interaction network components, also referred to as *entities*, include proteins, genes, and biological processes. An interaction between two entities may be *directed*, where one is acting upon the other, or in *undirected* manner, where the effect of the interaction is unknown. In a directed interaction, the *sign* of the interaction may be positive, where the amount or activity of the downstream element is increased, or negative, where there is a corresponding decrease in the amount or activity of the downstream element. One example of a signaling network is shown in Figure 2, which details major signaling events that occur in basal cell carcinoma. The individual components interact through post-translational modifications, complex formation, crosstalk between canonical signaling pathways, etc.

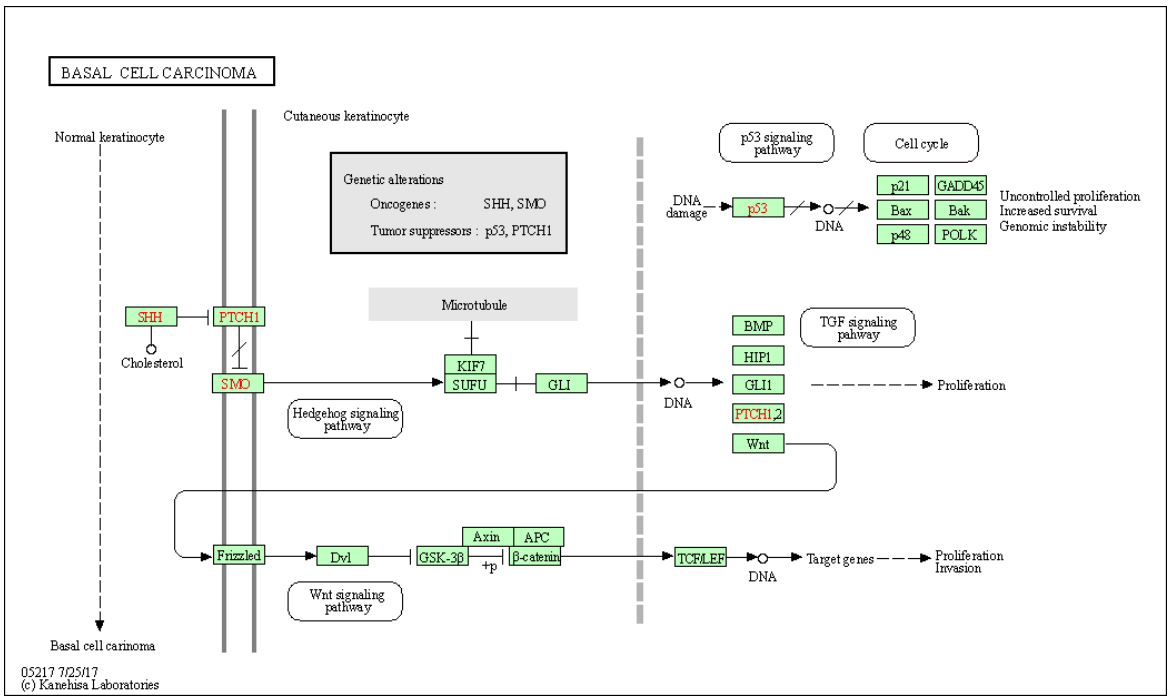


Figure 2. An example of a signaling network involved in basal cell carcinoma curated by KEGG [24].

2.3 Information retrieval

2.3.1 Literature queries

The amount of published work in molecular biology, biotechnology, and biomedical research increases exponentially every year [25]. There is a considerable number of published papers on any one mainstream biomedical research topic, potentially hundreds of thousands of relevant articles. For many areas of study, simply reading every paper is unrealistic, or even physically impossible. When studying biological systems, such as intracellular signaling networks, this problem is apparent – accurate representation of all relevant signaling events requires extensive, expert knowledge acquired over many years of study.

To retrieve relevant papers for given topic or question, a common method is to query databases that contain biomedical literature. One repository for biomedical literature, MEDLINE, contains over 27 million papers [26], and a common method for retrieving papers from MEDLINE is through its associated search engine, PubMed. Querying MEDLINE through PubMed is particularly useful for identifying papers on a specific context such as disease or cell type. It is also used for identification of individual proteins, signaling pathways, and general cell processes in one specific context. One example of a PubMed query that targets a single pathway in a specific context is ‘"hippo pathway" **AND** "stem cells"’. This query returns 272 papers, many of which describe hippo pathway signaling trends in cancerous stem cells [27-32], as well as non-cancerous stem cells. These papers contain a wealth of information about the mechanistic causes of stemness. However, retrieval of these papers requires *a priori* knowledge that the Hippo pathway is important in stem cell maintenance and renewal [31-33]. Additionally, these papers describe one small facet of stem cell signaling, and do not contain all the information needed to understand the system as a whole. To widen the scope, all papers in MEDLINE that concern stem cells can be retrieved by querying PubMed with "stem cells". Here, there are two obstacles – this query returns over 271,000 papers, many of which describe morphological or anatomical details, and not signaling pathways.

State-of-the-art methods for paper retrieval rely on term lists generated by experts or users [34, 35], or automated information retrieval of similar papers [36, 37]. These methods have several disadvantages. First, paper retrieval may depend on the cooperation of one or more experts in the field. Even for automated techniques that locate papers through related citations [36], or semantic analysis [37], some level of prior knowledge is needed.

2.3.2 Machine reading

The sheer number of peer-reviewed publications drives the need for automated methods for extracting information from text. Information extraction for modeling in systems biology can be greatly aided by machine reading. The state-of-the-art automated reading engines are capable of extracting cell signaling events from published papers [38-40]. For example, from the sentence "TNFa reduces BMPR-II expression in vitro and in vivo" [41], the REACH reading engine extracts the interaction "TNFa negatively regulates BMPR-II". By using natural language processing (NLP), machine readers are capable of extracting interactions from hundreds or thousands of papers in a matter of hours, achieving a substantial speedup over manual information extraction [42-44]. For this reason, automated methods for information extraction, such as machine reading, are used to assemble computational models of intracellular signaling networks.

However, current state-of-the-art methods for automation of network assembly are fraught with obstacles that make accurate network assembly time-consuming and labor-intensive [45-47]. NLP enables faster information retrieval, but at the price of reduced accuracy. Even manually extracted information may be inconsistent from one source to the next. Accurate representations of biological interactions are critical for assembling signaling networks, since even one misplaced interaction can have drastic consequences for understanding the true function and behavior of the network. In the same vein, missing interactions in an assembled signaling network can also affect dynamic behavior and lead to inaccurate conclusions.

2.4 Database resources

2.4.1 Entity databases

Entity databases curate information on biological entity types (Table 1). For example, the UniProt database contains information on genes, known transcripts, as well as information on the gene product, if available. UniProt also provides a convenient service for mapping plain text names to standardized IDs – a process also known as *grounding*. Grounding is an important step in the machine reading process, as it assigns a unique ID to each extracted entity. There is currently no resource that aggregates data on all biological entity types - proteins, genes, small molecules, biological processes, and miRNAs. While GILDA [48] is capable of inferring standardized IDs for multiple entity types from text, the accuracy of this tool varies greatly depending on the entity type. Thus, finding standard IDs for these entities is reliant on individual databases.

Entity databases can also provide valuable metadata describing the curation efforts for an entity. For example, each gene in the UniProt database has an assigned annotation score, which is an amalgamation of evidence of the gene and gene product's existence, including cross-references in other databases, known aliases, experimental evidence, and more. The annotation score has an integer value in the interval between 1 and 5, where score of 5 indicates ample evidence of the protein in existing literature and databases, and score of 1 indicates little to no available information about the protein. For example, the *TP53* gene in humans (UniProt ID P04637), a well-known tumor suppressor, has an annotation score of 5, while the *OATL1* transcript in humans (UniProt ID B4DF03), which has not been observed at the protein level, has an annotation score of 1.

Table 1. Entity databases frequently used for grounding entities.

Name	Entity type	Programmatic access?	Size
UniProt [49-51]	Genes and proteins	Yes (API)	569,213 reviewed / 245,871,679 unreviewed proteins
CHEBI [52]	Chemicals and small molecules	Yes (Web service)	151,344 substances / 139,678 annotations
GeneOntology [12, 53, 54]	Biological Processes	Yes (API)	43,096 GO terms / 7,486,838 annotations /1,503,185 gene products
miRbase [55]	mRNAs	No	38,589 miRNAs

2.4.2 Interaction databases

Interaction databases curate information on known biochemical signaling events (Table 2). This information can be curated manually, or inferred automatically from data. One database, STRING, contains both of these types of curated information about predicted protein-protein interactions (PPIs). STRING curates several different types of data on PPIs such as physical interactions, homologous sequences, and co-mentions in databases. The interactions in STRING are drawn from pre-existing databases, or manually extracted from either whole manuscripts or abstracts. STRING also scores the confidence in an interaction as a numeric value from 0 (low confidence) to 1000 (high confidence). Furthermore, there is detailed information on association type available for a subset of interactions. Experimental evidence that shows physical binding

increases the experimental score (escore). The database score (dscore) is derived from curated data from other sources. The textmining score (tscore) measures the co-occurrence of the two proteins in abstracts. These fields are present for all protein-protein interactions in the STRING database. The other score types include co-expression, homology, co-occurrence, fusion, phylogeny, and neighborhood scores.

Two other databases, Reactome [56] and BioGrid [57] also contain a sizeable number of PPIs. While both of these databases contain other interaction types, the bulk of the interaction data is mostly PPIs. BioGRID contains over 2 million protein-protein or protein-gene interactions, while containing <30,000 chemical interactions. While these databases store the same type of interactions as STRING, they are smaller and do not have scoring metrics. STITCH is a sister database to STRING and can be used in the same manner for protein-chemical interactions (PCIs). The escore, tscore, and dscore from STITCH are computed similarly to STRING.

A Gene Ontology (GO) term is a functional association between a gene and a biological process, and the GO annotations are based upon several different evidence types and are subject to multiple quality control measures [54]. Biological processes are frequently included in interaction networks, and the GO database standardizes these entities (Table 1). However, these annotations can also be used to judge the quality of protein-biological process interactions.

Table 2. List of popular interaction and pathway databases.

Name	Curation	API access?	Size
SIGNOR [58]	Manual (staff curators)	No	29,245 interactions
Pathway Commons [59]	Manual (from data providers)	No	5,772 pathways /2,424,055 interactions/ 22 databases
WikiPathways [60]	Manual (registered users)	No	>1,100 pathways
Reactome [56, 61]	Manual (staff curators)	Yes	13,827 interactions / 2536 pathways
STRING [11, 62, 63]	Manual and automated	Yes	>20 billion interactions
BioGRID [57]	Manual and automated	No	>3 million interactions
STITCH [13]	Manual and automated	Yes	1.6 billion interactions
KEGG [24]	Manual (staff curators *and* data providers)	No	59 pathways
HPRD [64]	Manual (staff curators)	No	>40,000 PPI, 36 pathways

2.4.3 Metadatabases

Due to the overwhelming number of interaction databases, metadatabases are gaining in size and popularity (Table 3). These metadatabases contain interactions from multiple sources and often have additional functionality such as visualization or sharing plugins. Metadatabases such as IntAct and OmniPath [65] aim to curate all possible interaction and pathway data in one repository. Other metadatabases, such as PCnet and the INDRA database use the aggregated information to provide a measure of confidence in the individuals interactions.

The Parsimonious Composite Network (PCnet) [66] is a high-confidence network of protein-protein interactions. PCnet uses 21 different human interaction databases to inform the network, where each interaction must be found in at least two of the 21 networks. This composite

network excludes interactions that are not reproducible, and therefore, it contains only high-confidence interactions. While PCnet interactions are highly supported, they are undirected, and independent of context.

The Integrated Network and Dynamical Reasoning Assembler (INDRA) is a system that draws on natural language processing tools and structured databases to collect statements about mechanistic and causal entity interactions [45]. INDRA relies on a number of machine readers to extract these interactions from literature. The INDRA database stores these statements that have already been processed, and provides a belief score for each interaction in its database.

Table 3. Common interaction metadatabases.

Name	Curation	API?	Representation format	Size
NDEx	Manual (registered users)	Yes	CX	>5,000 networks
PCnet	Manual (staff curators)	No	SIF	21 networks/databases
INDRA	Manual and automated	Yes	PySB, SBML, BEL, JSON	N/A
IntAct [67]	Manual (staff curators)	No	PSI-MITAB	5,565,271 interactions
OmniPath	Manual (staff curators)	Yes	SIF	100+ networks/databases

2.5 Network curation

2.5.1 Network standardization

To increase network accessibility and usability, many efforts have been made to standardize representation of signaling networks. Biological Expression Language (BEL) [68], Systems Biology Graphical Notation (SBGN) [69], Biological Pathway Exchange (BioPAX) [70], and Biological system Representation for Evaluation, Curation, Interoperability, Preserving, and Execution (BioRECIPE) [9, 71] are a few examples of network representation formats. For curation of directed interactions or networks, the BioRECIPE representation format is both human- and machine-interpretable [9]. Any network in BioRECIPE is also executable, and is compatible with a number of tools and other representation formats. For example, Systems Biology Markup Language (SBML) [72], a machine-interpretable representation format, is compatible with BioRECIPE. Translation between these two formats is automated and allows for increased reusability with SBML and modeling of dynamic behavior with BioRECIPE.

2.5.2 Network verification

There is a deficit in current methods for using existing interaction databases for verification of signaling networks. While there are many tools that can compute network similarity, such as MIMO [73] or SAGA [74], they can only do so between exactly two existing networks. VIOLIN (Verifying Interactions of Likely Importance to the Network) [75] evaluates the similarity between networks as measured by shared interactions (described in greater detail in Section 2.7.1).

Other tools allow for hosting and collaborative annotation of interaction networks. With platforms like BioKC [76] and MINERVA [77] curators can upload networks and provide feedback on networks curated by other users. However, their verification process is entirely internal - MINERVA has quality control settings to ensure all fields for uploaded annotations are complete. Other tools, such as CompNet [78] and Cytoscape [79], have features for visualizing overlapping signaling networks. There are multiple tools for comparing networks inferred from co-expression data ([74]). However, these tools do not incorporate *a priori* knowledge from interaction networks or databases. Furthermore, these tools are unable to compare more than two networks, and often have size limits- one exception is CoDINA [80], however, this tool assumes that networks have been inferred from expression data. To curate comprehensive, reliable, and context-aware networks, there is a need for verification methods integrated with existing knowledgebases, that are capable of comparing three or more large networks.

2.5.3 Network sharing and accessibility

Platforms for sharing and annotating model repositories influence information availability. With platforms like BioKC [76] and MINERVA [77] curators can upload networks and provide feedback on networks curated by other users. Other hosts, such as BioModels [81] or CellCollective [82], often do not support automated methods for network curation.

Table 4. Model repositories.

Name	Programmatic access?	Compatible model formats	Number of models/networks
NDEx [83]	Yes (API)	CX	>5,000 networks
BioModels [81]	No	SBML (preferred), CellML, matlab	2,647 models
CellCollective [82]	No	SBML, Boolean expressions	229 models
MINERVA [77]	Yes (API)	SBML	9 networks
BioKC [76]	Yes (API)	SBML	No public networks
Path2Models [84]	No	SBML	~140,00 models
MINT [85]	Yes (API)	MITAB	>90 hosted models

2.6 Element-based models

Understanding the complex feedback between genes, protein, chemicals, and larger cell processes requires modeling methods capable of representing different scales, both in terms of size and time. In contrast to causal models inferred from expression or other genomic or epigenomic data, mechanistic models incorporate a priori knowledge about signaling events. Executable

mechanistic models are used to study dynamic behavior, and they rely on either ODEs [23], reaction rules [86], or element update rules [2, 87, 88]. However, mechanistic modeling of cellular processes and biochemical interactions may also rely on reaction rates and other kinetic parameters that are not readily available.

Element-based modeling [2, 89-91] is capable of representing biochemical reactions without strictly relying on most of these parameters. In this approach, each model element represents a biological entity (described in Section 2.4.1), and the element state over time is determined by its update rules. The update function for an element may be based on Michaelis-Menten reaction kinetics, or a discrete function. Discrete models [14, 92, 93] are an example of element-based models, as are Bayesian Networks [94].

2.7 DySE

The Dynamic System Explanation (DySE) [89] framework is a collection of methods and tools for information extraction, as well as curation and analysis of element-based models. DySE uses interactions obtained by machine reading to automatically assemble executable models at different levels of abstraction. The DySE framework includes methods for automated model analysis to predict system behavior or guide interventions. One of these methods is the Discrete Stochastic Heterogeneous Simulator (DiSH) [87], which is capable of reproducing dynamic cell signaling behavior. DiSH takes an executable model written in the BioRECIPE format [71] (Table 5), along with simulation parameters, and outputs trajectories (i.e., state changes in time) for all model elements. The simulation can be parametrized to reproduce in silico any in vitro or in vivo scenarios, which include information about the starting state of the system and treatments such as

inhibitors, knockouts, added cytokines, etc. DiSH can provide insights into all reachable steady states under given scenario, as well as transient state changes over a studied time interval.

Table 5. BioRECIPE model format. Each element

Element Name	Positive Regulator	Negative Regulator	Motif
Vav2	**Gab2**		**Post-translational modification (activating)**
Cdc42		**Shp2**	**Post-translational modification (inhibiting)**
RhoA		**(Gab2,Shp2)**	**Post-translational modifications (AND)**
Gab2_gene	**E2F1**		**Transcription -gene**
Gab2_rna	**Gab2_gene**		**Transcription - RNA**
Gab2	**{Gab2_rna}[HER2]**		**Translation**

2.7.1 VIOLIN

Beyond finding shared interactions (corroborations), VIOLIN also indicates interactions that would extend the model (extensions), as well as interactions that contradict the model (contradictions). VIOLIN will flag interactions that cannot be judged automatically- specifically, interactions in a potential feed-forward, feedback loops, or self-regulations. These interactions must be manually reviewed to be established as corroborations, contradictions, or extensions.

2.8 The genomic profile of GBM

GBM, the most common and deadly brain tumor in adults, is remarkably resistant to current treatments [5, 95]. Clinicians utilize radiation, surgical resection, and temozolomide (TMZ) to treat GBM patients. Unfortunately, patient response to TMZ is highly variable – specifically, the methylation of the MGMT gene determines treatment efficacy [95]. Tumor heterogeneity complicates the development of effective therapeutic strategies. Furthermore, these tumors are reliant upon a subpopulation of cancer stem cells, which can resist therapy and are thought to reinitiate tumor growth following chemotherapy and radiation [96]. While non-stem tumor cells may be susceptible to certain drug treatments, cancerous stem cells have been shown to evade treatment and restore tumor bulk post-treatment.

Chemoresistance is a hallmark of GBM tumors. This is due to the large number of genetically distinct clones present within one individual tumor [97]. With virtually any chemical treatment, there is likely at least one subpopulation that has a mutation which grants resistance. From this subpopulation, the tumor will continue to grow. In addition to the diverse landscape of potential somatic mutations within a GBM cell, epigenetic changes can affect tumor growth and treatment response [98]. Tumors from different patients are also genetically diverse, decreasing the chance that one single treatment will be effective for all patients. Given the rapid progression of this cancer and likelihood for chemoresistance, patients are in need of individualized solutions.

3.0 Automated biocuration tools

This chapter describes automated approaches for biocuration - FLUTE, MINUET, and data-driven literature queries.

3.1 FiLter for Understanding True Events (FLUTE)

This section describes the FiLter for Understanding True Events (FLUTE), a database, tool, and methodology to select interactions with high confidence from the set of events extracted by machine reading. The main contributions of the proposed work are a fast automated tool to reduce the vast number of cellular events extracted by machine reading and facilitate rapid model building, and a filtration methodology to select interactions for addition to an existing model, and to increase confidence in the interactions added to the model. The results highlight the influence of query categories and topics on the number of papers found, and on the percentage of selected interactions output by FLUTE, and findings are discussed in Section 3.1.4.

3.1.1 FLUTE workflow

Figure 3 outlines a typical FLUTE workflow. The selection of inputs for FLUTE is guided by user queries and can be compiled, through manual or machine reading of literature, into a set of extracted interactions. While FLUTE is best utilized when filtering machine reading output, it can be applied to manually extracted interactions as well. FLUTE outputs selected interactions, a

subset of extracted interactions, which can then be used to curate models and help answer the queries.

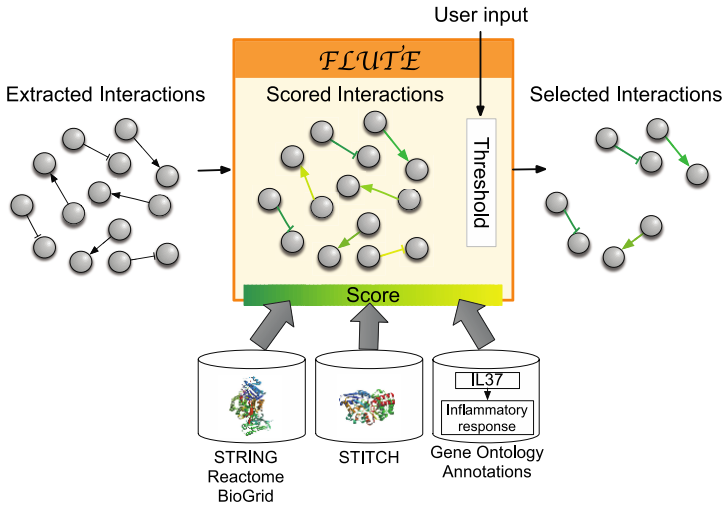


Figure 3. Filtration process with FLUTE: Inputs to FLUTE include extracted interactions, scores of these interactions that are found in databases, and the user's selection of thresholds for the scores. Outputs from FLUTE include selected interactions determined by their scores and thresholds.

FLUTE can process any set of interactions, from any source, as long as the interactions are represented in either a list of edges or the BioRECIPE input format. The IDs for both entities must be known, and the entity types (protein, biological process, etc.) can be inferred from the ID. While FLUTE does not explicitly check the effect of the interaction, or the reference listed, this information can be used by human curators or by downstream model assembly tools. Each extracted interaction also has an associated evidence statement with the text from which the event was extracted and could also be used for human judgement.

The reading engine REACH [99] was used to extract relevant information from selected literature. Through manual curation of machine reading output, four major types of errors in the extracted interactions were identified:

1. ambiguous or misconstrued sentences (Omission error)
2. interactions where one or both elements are incorrectly grounded (Grounding error)
3., and interactions that have opposite directionality (Direction error)
4. interactions that have opposite effect (Sign error)

In the case of Omission error, the reader denotes a relationship between two elements that does not exist in the evidence statement, while in the Grounding error, the reader was unable to match the elements in the interaction to the correct IDs. As an example, for the evidence statement "Although Tcf3 binds GSK3, it does not inhibit the activity of GSK3 against axin.", machine reading gives us the following interaction: "Tcf3 inhibits GSK3". Due to the verbosity of the sentence, machines output an incorrect interaction. From the sentence "CtIP (CtBP interacting protein) is also critical for HR mediated DSB repair", machine readers extract an interaction where HR regulates DSB, both classified as proteins. However, DSB stands for double-stranded break, not the protein DSB, thus leading to a grounding error. The distribution of these error types in the context of several queries is described in Section 3.1.6.

3.1.2 The FLUTE database design

To harness the advantages of both query-specific machine reading and more reliable databases, FLUTE uses multiple databases to determine the confidence in the interactions

extracted from literature by machines. The databases used are the GO, STITCH, BioGrid, Reactome, and STRING databases. For ease of use, interaction data is stored in a MySQL database, and stores the interaction information offline. The database schema is shown in Figure 4. The database contains six tables total, which can be classified into four categories: protein-protein interaction data, protein-chemical interaction data, protein-biological process data, and ID mapping. The aggregated FLUTE database contains more than 30 million unique interactions. This setup is easily utilized to select multiple interaction types from the reading, based on the level of support found in the literature.

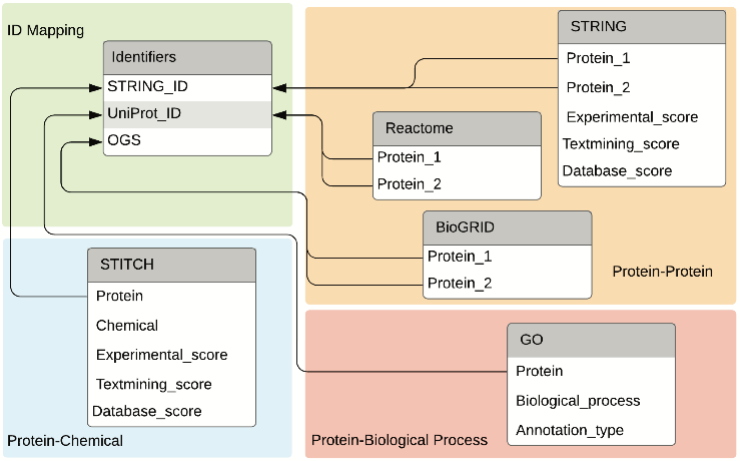


Figure 4. Databases and the connections between databases used by FLUTE.

Protein-protein interaction data was imported from STRING, BioGRID, and Reactome. For STRING, the escore, dscore, and tscore metrics (described in greater detail in Section 2.4.2) were included in the database schema. These fields are present for all protein-protein interactions in the STRING database. The other score types include co-expression, homology, co-occurence, fusion, phylogeny, and neighborhood scores. However, these are less likely to have nonzero values, and are less likely to be indicative of a physical interaction, and as such, were not implemented in the FLUTE database. In contrast to STRING, Reactome and BioGRID protein-protein interaction information does not contain a score, and therefore, it was incorporated in

FLUTE simply as an indication of whether an interaction is known. All protein-chemical interaction data is imported from STITCH. For protein-biological process interactions, the FLUTE database also contains list of all GO annotations. While there is no "score" for the confidence of these annotations, there is an annotation type that describes the curation method (e.g., experimentally, electronically, etc.).

The final table included in the FLUTE database schema is a mapping for all STRING IDs to their UniProt IDs and the HGNC-approved gene symbol. While STRING and STITCH use Ensembl IDs for proteins, Reactome and GO use UniProt IDs. BioGRID uses the HGNC-approved gene symbol, instead of either of the previously mentioned ID types. Figure 4 shows the relationships between the ID table and fields that can be converted. While the ID mapping table contains all known data for each of the three ID types, there is no guarantee that all three fields will be available for every known protein.

Executable models assembled downstream of FLUTE require the information about interaction direction, therefore, it is important to note that STRING does not always include a direction in the interactions it supports. Therefore, FLUTE obtains this information from the machine reading output. If there is a specific interaction, for example, phosphorylation, that is clearly directed in STRING. Similarly, STRING can determine if an interaction is positive or negative depending on whether there is evidence for the sign of interaction. However, this information is not always available, and so it has not been implemented in FLUTE, that is, if an extracted interaction matches the available data, it will be selected, even if the interaction has a Sign error.

3.1.3 FLUTE database thresholds

When a new input from machine readers is provided to FLUTE in response to a user query, FLUTE matches elements within this input to the information in the ID mapping table. Once all IDs have been matched, FLUTE searches the relevant databases for each interaction. For example, the search for PPIs is conducted on the Reactome, BioGRID, and STRING databases. All supporting fields, such as scores, are extracted and reported for all interactions. FLUTE discards any unmatched interactions. Furthermore, besides guiding literature selection with queries, FLUTE allows users to select database thresholds, that is, interaction score thresholds that tailor the number and confidence of the selected interactions. For each interaction type, FLUTE can select only interactions that meet a certain score. For example, FLUTE can return only PPIs and PCIs with escore > 0, which guarantees that all selected interactions have at least one source of experimental data. A higher score threshold will decrease the number of selected interactions, but it will also increase the confidence in the selected interactions.

To complement the selection that relies on database thresholds, FLUTE can also select interactions based on the year of publication and their repeated occurrence in literature. To do so, FLUTE uses two non-database thresholds, one threshold for the earliest allowed publication year, and another threshold for the least required number of papers that mention the same interaction. Following these thresholds, FLUTE can flag interactions from recently published papers as potentially novel interactions, and interactions that appear in multiple papers as between-paper duplicates. Besides the between-paper duplicates, there are also within-paper duplicates, that is, interactions repeating in the same paper. However, if interactions are repeated in one paper only, it was assumed that they are lower confidence interactions when compared to those that appear in multiple papers, and therefore, an optional flag for the within-paper duplicates was not

implemented. By marking interactions as potentially novel or between-paper duplicates, FLUTE allows the user, if desired, to find the interactions that have been recently published, or those that have more support in literature. For example, a user query for a well-known pathway may include a gene or protein with a recently discovered function. In this case, interaction databases may not be up-to-date, and the option to select potentially novel interactions or between-paper duplicates could be beneficial for modeling. Furthermore, since these interactions are flagged, the user can easily find them and conduct a further manual review.

3.1.4 Influence of query choice

To explore the influence of various topics that could be included in queries, FLUTE results were obtained for 28 different queries (Table 6). To ensure results were obtained for a wide range of possible subjects, 16 different query topic categories (e.g., “Disease and Pathway”, Q7) were selected. An example topic for each category (e.g., “Breast Cancer, MAPK/ERK pathway”) was chosen, and finally, the terms for each query topic that are combined into a machine readable query written as a logical expression (e.g., “breast cancer” AND ("Erk pathway" OR "MAPK pathway" OR "Ras pathway")). The queries were written to account for the fact that, in biological literature, different aliases can be used across biological papers to represent the same entity (e.g., "rsk 90" or RPS6KA1 or RSK-1 or S6K, in Q13a-d), and that some aliases include characters that are not accurately recognized by machine reading engines (e.g., ‘-‘ in Q13c).

Table 6. Queries: different topic categories, example query terms for each topic category, and the corresponding query expressions entered in PubMed.

#	Query topic category	Query terms	Query expression
1	Disease	Breast cancer	“breast cancer”
2	Cellular Process	DNA repair	"DNA repair"
3	Signaling Pathway	MAPK/ERK pathway	"erk pathway" or "mapk pathway" or "ras pathway"
4	Protein	BRCA1	BRCA1
5	Chemical	Progesterone	Progesterone
6a	Disease and Process	Breast cancer, DNA repair	“breast cancer” and “dna repair”
6b		Autophagy, cancer	autophagy and cancer
7	Disease and Pathway	Breast cancer, MAPK/ERK pathway	“breast cancer” and ("erk pathway" or "mapk pathway" or "ras pathway")
8	Disease and Protein	Breast cancer, BRCA1	“breast cancer” and brca1
9	Disease and Chemical	Breast cancer , progesterone	“breast cancer” and progesterone
10a	Process and Protein	DNA repair, BRCA1	“dna repair” and brca1
10b		ADAM17, inflammation	ADAM17 and inflammation
11a	Well-Studied	EGFR	EGFR
11b		HER2	her2
12	New Discovery	copb2*	copb2
13a	Multiple Aliases	RSK90	"rsk 90"
13b			RPS6KA1
13c			RSK-1
13d			S6K
14a	Non-Standard Characters	Beta catenin	CTNNB1
14b			“Beta catenin”
14c			Beta-catenin
14d			CTNNB
15a	Different Gene and Protein Name	Estrogen receptor	"Estrogen Receptor 1"
15b			ESR1

15c			ER
16a	Same Gene and	PTEN	Pten
16b	Protein Name	GRB2	GRB2

The results in Figure 5, which were obtained for the list of queries in Table 6, suggest that the selection of a query topic, the choice of terms and characters in the query, and the terms' presence in literature can all affect the number of papers retrieved from PubMed. As Figure 5 shows, the number of papers returned by a PubMed search can vary several orders of magnitude (from tens to hundreds of thousands) for different queries. For example, a well-studied term (e.g., EGFR, Q11a) will return many papers, whereas a recent discovery (e.g., copb2*, Q12) will have fewer PubMed hits. Furthermore, for terms with special characters or multiple aliases, machine reading may have difficulty extracting all relevant interactions, as shown by examples Q13a-d, where searches for different aliases of RSK90 all returned a different number of papers. To get comprehensive results, all well-known aliases may have to be included in a query. These results highlight the importance of the careful choice of query terms.

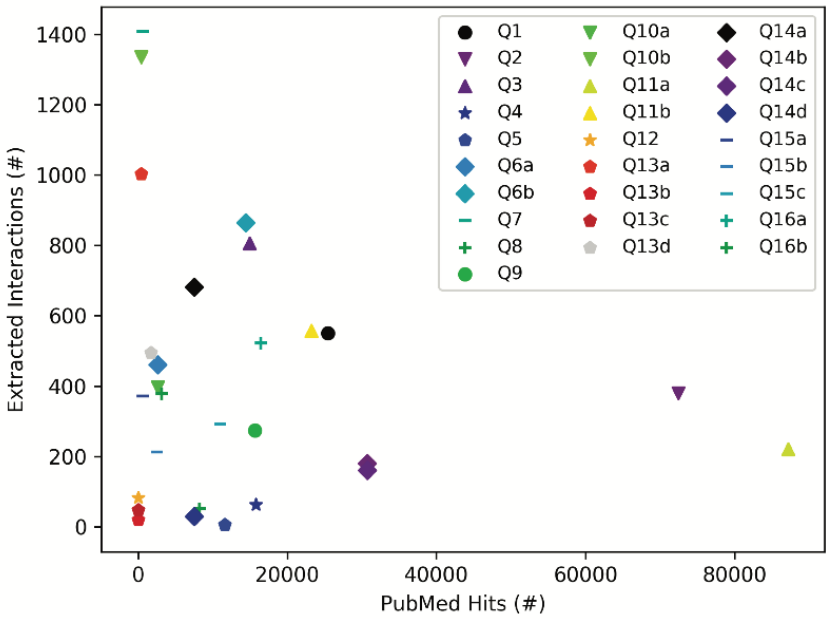


Figure 5. Influence of query category and term choice (the legend corresponds to query numbers in Table 6) on the number of papers found in PubMed and on the number of interactions extracted from the top 200 papers (except Q12, Q13b, and Q13c, where PubMed returned less than 200 hits). Results obtained for the same query topic category, but different term aliases, or different example terms, are grouped together with the same marker shape and similar color.

To explore the influence of query topic on the number of interactions that machine reading can extract, for each query term, either all the found papers were selected, or the top 200 PubMed hits with valid PMC IDs (when the number of PubMed hits is larger than 200). A cutoff of 200 papers was chosen to ensure that every query would return at least a few dozen interactions. The results in Figure 5 suggest that the query topic and the choice of query terms could have a significant impact on the size of the machine reading output, as the number of extracted interactions does not seem to be correlated with the number of papers read. As expected, the query topic influences the selection of papers, while scientific texts can vary in the level at which they describe systems, from high level review papers, to those that focus on precise mechanistic details of a small number of biochemical interactions. The choice of query terms, and the characters that

are used in these terms can also have a strong influence, if machine readers are not trained to recognize most of the aliases of the same entity.

Interestingly, the selection of a query topic and query terms did not have a noticeable influence on the FLUTE output. In other words, while being conservative and selecting only 8.86% (mean computed for the 28 queries in Table 6) of the overall number of instructions provided by machine reading, this percent was relatively consistent across queries (standard deviation of 4.02%). These results suggest that FLUTE can reliably filter interactions for any query category, that is, it provides to model assembly only those interactions that have high confidence.

Finally, compared to manually filtering the interaction sets from the machine reading output, FLUTE achieved a significant speedup. Assuming it would take a human approximately 30 seconds to judge one interaction, the average speedup that FLUTE achieved was 2560.28, with a standard deviation of 482.98. That is, FLUTE can increase the rate at which interactions are selected from hours to seconds, or from days to minutes.

3.1.5 Influence of interaction type

The remainder of the experiments focused on three sets of interactions: the first two sets are obtained as a result of the two queries from Table 6, Q6b (Disease and Process query) and Q10b (Process and Protein query), and the third set (referred to as a Multiple Protein query) is obtained using the REACH reading engine for several individual protein queries (MEK, ERK, AKT, GSK3, P70RSK, S6, CDK4, 4EBP1, YB1, SRC, CHK2, MTOR, and PI3K). For queries Q6b and Q10b, the 200 most relevant papers were selected from PubMed, and REACH extracted 865 and 1336 interactions from these papers, respectively. In the third case, from the papers returned by the REACH Explorer tool, followed by REACH (when necessary to get more papers),

followed by manual selection of ten relevant (all in the context of melanoma) papers for each of the 13 proteins. REACH read these 125 papers and extracted 6305 interactions.

To prepare the reading output for FLUTE, the type for each interaction was determined, in particular, focusing on all the interactions where the interacting elements are either of protein (P), chemical (C), or biological process (BP) type (i.e., interactions of type PPI, PCI, PBPI, CCI, CBPI, and BPBPI), and all the other interactions are assigned to type Other (Figure 6(a)). The interaction type Other includes molecules such as mRNAs, protein families, or unknown types. Protein families are common, as well as complexes, however, these types are excluded from analysis in this work due to difficulty mapping to a standard identifier, and a lack of data on known interactions.

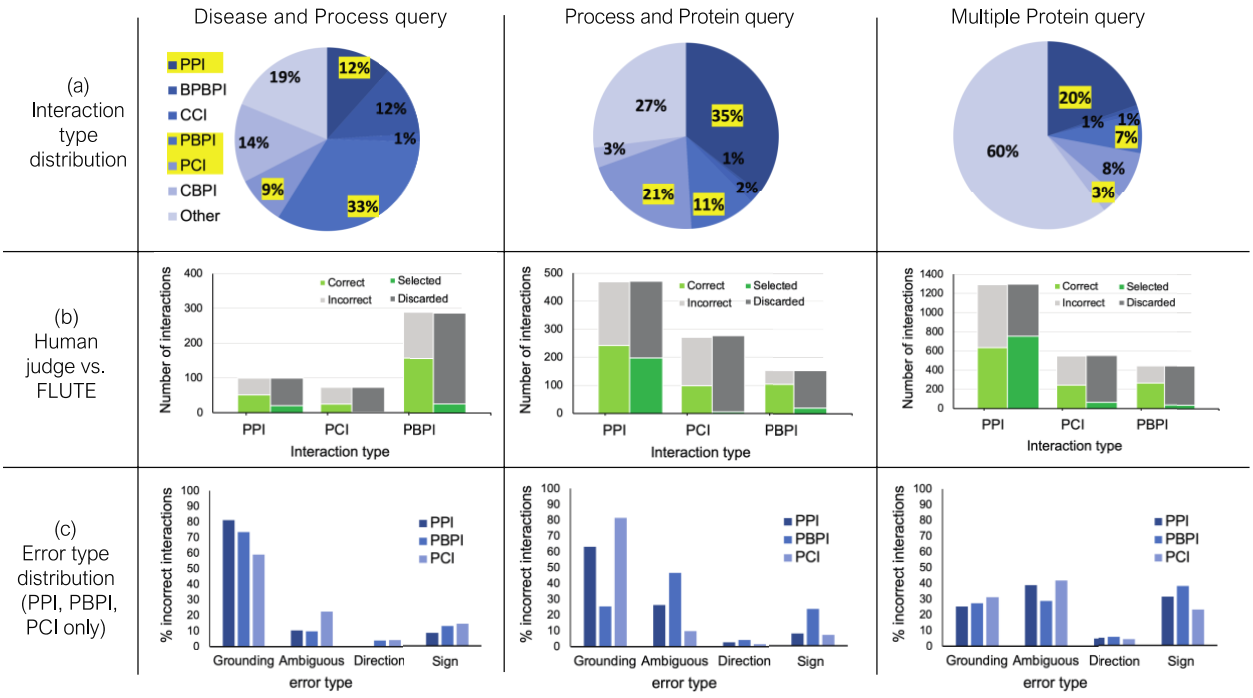


Figure 6. The influence of interaction type and machine reading errors on the number of selected interactions. (a) Overall distribution of interaction types for the three different queries, disease and biological process query, biological process and protein query, and multiple protein query. (b) The comparison between FLUTE and manual selection; human judge decides whether interaction is correct given literature evidence, and FLUTE

selects the interactions that are supported by databases. (c) The distribution of errors types in machine extraction of PPIs, PBPIs, and PCIs for the three different queries.

The three sets of extracted interactions obtained from machine readers were processed both manually and with FLUTE (Figure 6(b)). First, each interaction was manually assigned to one of the two groups, "correct" and "incorrect", based on whether the evidence statement that the machine reader provided agreed with the extracted interaction or not. FLUTE was used to filter the same three sets of extracted interactions, that is, assign each interaction to either "selected" or "discarded" group, based on whether it was supported by the databases that FLUTE uses.

The results shown in Figure 6(b) suggest that the accuracy of machine reading varies with different interaction types. From manual filtration, the PBPIs appear to be correct more often (54-69% correct), while the PCIs are the least likely to be correct (36-45% correct). Approximately half of all PPIs are correct (47-52% correct). Machine reading may erroneously extract PCIs from papers that use a recognized chemical in the methods protocol. Grounding may also be difficult for chemical compounds, due to the prevalence of non-standardized names. On the other end of the spectrum, PBPIs may be correct more frequently since biological process names are almost never abbreviated. Overall, for all three interaction sets, the number of correct interactions is approximately half the size of all the extracted non-Other interactions. On the other hand, across all three sets of interactions, FLUTE selects much higher percent of PPIs, compared to the non-PPI interaction types. The number of interactions selected by FLUTE is also smaller than the number of interactions manually marked as "correct". While the number of selected PPIs is similar to the number of correct PPIs, FLUTE is much less likely to select PCIs and PBPIs. This is due to the fact that the information on both PCIs and PBPIs is found less frequently in the databases used by FLUTE. This results in a much smaller output from FLUTE, compared to manual filtration, as well as a different distribution of interaction types in the final output.

3.1.6 Influence of machine reading errors

To provide further guidance for the use of FLUTE, the types of errors in the reading output was investigated, along with whether FLUTE is sensitive to the difference in machine reading error types, and also how well it can filter out the errors. Figure 6(c) shows the relative abundance of the four error types, Grounding, Omission, Direction, and Sign (see Section 3.1.1 for definitions) in the three reading sets. For the Disease and Process query (Figure 6, left) and the Process and Protein query (Figure 6, middle), the distribution of error types varies slightly across different types of interactions (PPI, PBPI, and PCI), with mostly Grounding and/or Omission errors across all three interaction types, while Sign errors are generally lower. For the Multiple Protein query (Figure 6, right), with the exception of Direction error, the other three error types remain consistent across interaction types. The machine reading output rarely had Direction errors in any query category or interaction type for the selection of manually curated interactions studied.

The results in Figure 6(c) suggest that FLUTE could be especially useful in the case of papers with proteins or genes that have non-standard names, or descriptions of complicated signaling pathways, such as those obtained for the example Disease and Process and Process and Protein queries. This is due to the fact that FLUTE is capable of filtering out a significant portion of interactions with Grounding and Omission errors. However, FLUTE does not address interactions with Direction or Sign errors, as STRING and STITCH do not always contain information about the direction and sign of interactions. Furthermore, GO annotations do not provide cause and effect information, only correlations, and therefore are not suitable for assignment of direction or sign.

Overall, FLUTE performed well on the interaction sets due to the relatively low occurrence of both Direction and Sign errors in these sets, but this may not be the case for other queries and

interaction sets. In general, the information about direction and sign is critical for creating models that are used to study system dynamics, and a number of Direction and Sign errors can often be identified by examining contradictions within the machine reading output.

3.1.7 Interaction scores and thresholds

FLUTE allows the user to choose confidence level for selected interactions, that is, for the three different score types, the user can choose a score threshold value for interactions. Using several threshold values (0, 200, 400, 600, 800) for the three STRING score types, the effect of score types and their values on the FLUTE output size were studied. As Figure 7 shows, the number of selected PPIs decreases with the increase of a threshold. While the number of selected interactions decreases linearly with escore and tscore thresholds, the number of selected interactions is affected only at very low or very high threshold value for the dscore. The escore and dscore metrics are stringent due to the type of evidence required: either evidence of physical binding, or a well-known association present in a pathway database, respectively. The tscore seems to be least selective, allowing more interactions to pass through the filter, while escore causes the largest reduction of output size. Since the tscore is calculated using abstract co-mentions, there is less confidence in the results using a tscore threshold than if an escore threshold had been used.

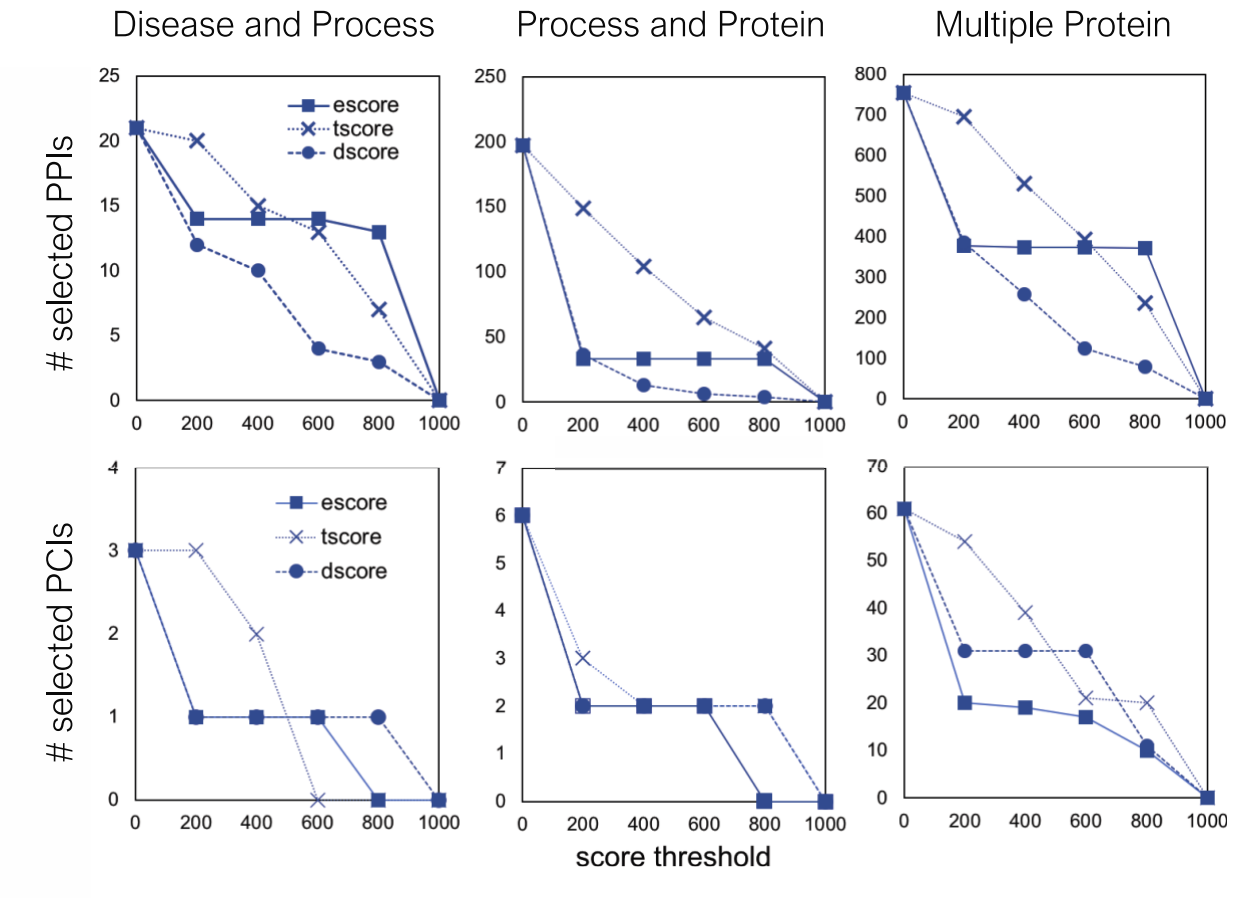


Figure 7. The number of selected interactions, PPIs (top) and PCIs (bottom) as a function of a score threshold for each score type, for the three different queries.

For PCIs, a similar threshold-based approach using the score types from the STITCH database can be implemented. However, due to the scarcity of PCIs in the selected output, only the STITCH escore can be used as a threshold. Similar to the PPIs, the number of selected PCIs decreases with the increase in the score threshold. Any escore threshold larger than 0, for all three queries, decreases the number of selected PCIs by ~66-67%. While the escore metric appears to be the most stringent for all the interaction sets, those interactions that go through the filter using the escore have concrete evidence of physical interaction. Therefore, there is higher confidence in any interactions, either PPIs or PCIs, that are selected using the escore.

3.1.8 FLUTE precision and recall

To validate the correctness of the PPIs selected by FLUTE, the overlap between human judgement and FLUTE output was compared. Precision and recall were calculated for each query, by finding the percent of PPIs selected by FLUTE that were also marked as correct (precision), as well as the total number of interactions manually judged as correct that were also selected by FLUTE (recall). For each score type (escore, tscore, or dscore), the precision and recall were calculated at scores 0-1000, with intervals of 200. Both the effect of using one subscore as a threshold and using a combination of all three subscore types were tested. Figure 8(a) shows the effect of changing one subscore threshold at a time while the other two subscores have no threshold constraints. In Figure 8(b), the average precision and recall was calculated for each of the 125 different score type combinations. To get the average precision and recall, the mean for each 25 precision and recall values were considered, where one score type is kept with a constant value. Figure 8(c) shows precision and recall for PCIs at one threshold, due to the small output size of filtered PCIs, and PBPIs supported by the GO database. Using one subscore threshold at a time favors higher recall, at the cost of precision, while using multiple subscore thresholds together results in high precision but low recall. For the Multiple Protein interaction set, increasing the threshold did not increase precision, however, it did for the other two queries. Recall decreased in response to raising the score threshold, since higher thresholds exclude more interactions. As the threshold is increased, FLUTE inevitably excludes more correct interactions in the selected output.

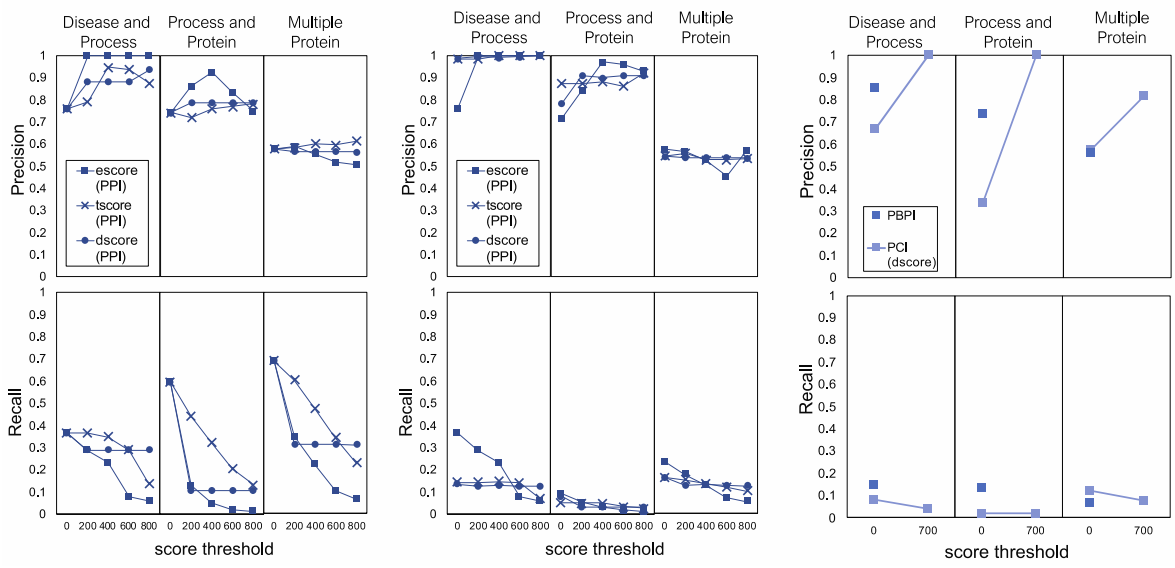


Figure 8. Precision and recall of FLUTE, compared to human judging, and the sensitivity of precision and recall to the scores, for the three different queries: (a) precision and recall when filtering PPIs with only one subscore at a time, (b) average precision and recall when filtering PPIs for all possible subscore combinations, and (c) precision and recall when filtering PBPI and PCIs.

The increase of precision in response to more stringent score thresholds (Figure 8) indicates that higher STRING and STITCH scores are correlated with correct machine reading output. Any of the three STRING score types that were tested, or the STITCH escore, are capable of differentiating between correct and incorrect machine reading output. Using interaction databases to inform interaction selection results in a higher-confidence output. Overall, these suggest that FLUTE can prioritize either quality or quantity of interactions, depending on user-determined thresholds. Selecting a low FLUTE threshold will output a higher quantity of interactions, at the cost of the correctness of the individual interactions. By comparison, a high threshold will output less interactions, however, there will be more confidence in the results.

Tables 7-9 show the updated precision and recall when different combinations of database and non-database thresholds are used, for filtering PPIs, PCIs, and PBPIs. The publication year threshold was chosen based on when the oldest dataset was gathered in the three interaction sets,

Disease and Process, Process and Protein, and Multiple Protein. The Multiple Protein interactions set was obtained from papers published as recently as 2016, while the other two sets were obtained from papers up until 2018. Therefore, for the PPIs (Table 7), interactions published after 2014 were chosen, which would return potentially novel interactions. The duplicate threshold was set to either 2, 4, or 6 duplicates (interactions extracted from 2,4, or 6 papers, respectively) as anything beyond 6+ duplicates is extremely rare in the interaction set. For the Disease and Process PPI dataset, the output is small enough that there are no between-paper duplicates. The upper limit on number of duplicates increases as the size of the interaction set increases, so the optimal non-database thresholds change for each interaction set. These interactions were flagged, and added to the PPIs filtered using the FLUTE database thresholds.

Table 7. The effect of inclusion of between-paper duplicates or potentially novel interactions on precision and recall in PPIs. Red numbers besides recall indicate the number of true interactions added by using non-database filters.

Query		Any STRING score ≥0	Any STRING score ≥0 or published after 2014	Any STRING score ≥0 or 2+duplicates	Any STRING score ≥0 or 4+duplicates	Any STRING score ≥0 or 6+duplicates	Any STRING score ≥0 or 2+duplicates or published after 2014
Process and Protein	Precision	0.74	0.53	0.74	0.73	0.72	0.74
	Recall	0.60 (+0)	0.91 (+74)	0.66 (+15)	0.63 (+7)	0.62 (+6)	0.92 (+76)
Disease and Process	Precision	0.76	0.37	--	--	--	--
	Recall	0.37 (+0)	0.48 (+6)	--	--	--	--
Multiple Protein	Precision	0.58	0.57	0.57	0.58	0.58	0.55
	Recall	0.69 (+0)	0.75 (+38)	0.72 (+18)	0.70 (+9)	0.70 (+3)	0.77 (+50)

Similarly, the PCIs (Table 8) and PBPIs (Table 9) were filtered, using the least stringent interaction database threshold, a 2014 publication year thresholds, or a thresholds of at least two between-paper duplicates. As expected, interactions selected using these non-database filters greatly increase the recall of FLUTE output, however, this comes at a cost to precision. These

results show that using flags may be useful for indicating interactions that could benefit from manual review, but these thresholds are not rigorous enough to warrant automatic inclusion into filtered output. The research topic determines the queries as well as the machine reading output sets, and therefore, the optimal combination of thresholds will be largely context dependent.

Table 8. The effect of inclusion of between-paper duplicates or potentially novel interactions on precision and recall in PCIs. Red numbers besides recall indicate the number of true interactions added by using non-database filters.

Query		Any STITCH score ≥0	Any STITCH score ≥0 or published after 2014	Any STITCH score ≥0 or 2+duplicates
Process and Protein	Precision	0.33	0.36	0.66
	Recall	0.02 (+0)	0.90 (+88)	0.43 (+40)
Disease and Process	Precision	0.67	0.34	0.75
	Recall	0.08 (+0)	0.40 (+8)	0.12 (+1)
Multiple Protein	Precision	0.54	0.43	0.48
	Recall	0.14 (+0)	0.29 (+36)	0.25 (+26)

Table 9. The effect of inclusion of between-paper duplicates or potentially novel interactions on precision and recall in PBPIs. Red numbers besides recall indicate the number of true interactions added by using non-database filters.

Query		Any GO annotation	Any GO annotation or published after 2014	Any GO annotation or 2+duplicates
Process and Protein	Precision	0.74	0.68	0.76
	Recall	0.13 (+0)	0.79 (+69)	0.51 (+40)
Disease and Process	Precision	0.85	0.42	0.84
	Recall	0.18 (+0)	0.51 (+52)	0.39(+33)
Multiple Protein	Precision	0.67	0.65	0.67
	Recall	0.13 (+0)	0.38 (+59)	0.34 (+56)

3.1.9 FLUTE database-based expansion of interaction set

Besides selecting high-confidence interactions using database thresholds, the FLUTE database can also be utilized to find interactions by their citation. In other words, to supplement the results of machine reading, a function in FLUTE searches the FLUTE database for additional interactions that cite the same papers as those read by reading engines and includes this set of interactions in the output. This FLUTE function allows for finding interactions in the selected papers that reading engines have missed.

3.2 Selecting context-aware, targeted literature

This section describes the use of automatically generated targeted queries in information extraction conducted by machine readers, followed by automated reasoning about affected signaling networks and biological processes. This method allows the for identification of differentially expressed genes (DEGs) in the context of a disease, cell line, tissue type, or other condition (e.g., drug treatments), and for using them to form query terms when searching literature.

3.2.1 Identification of differentially expressed genes

As shown in Figure 9, the first step in the query design method is to define a context for literature search. This approach allows a user to automatically design queries for many different contexts, including any biological condition that can be observed long enough to generate gene expression data. The user selects a data source and a relevant dataset from that source. While any

kind of gene expression data can be used (microarray, RNA-seq, or single cell RNA-seq), public databases for expression data most frequently include RNA-seq data. Public databases for RNA-seq data include the Cancer Genome Atlas (TCGA) [100], Gene Expression Omnibus [101], and the Expression Atlas [102], all of which contain sufficient expression data to be used in the proposed query generation method.

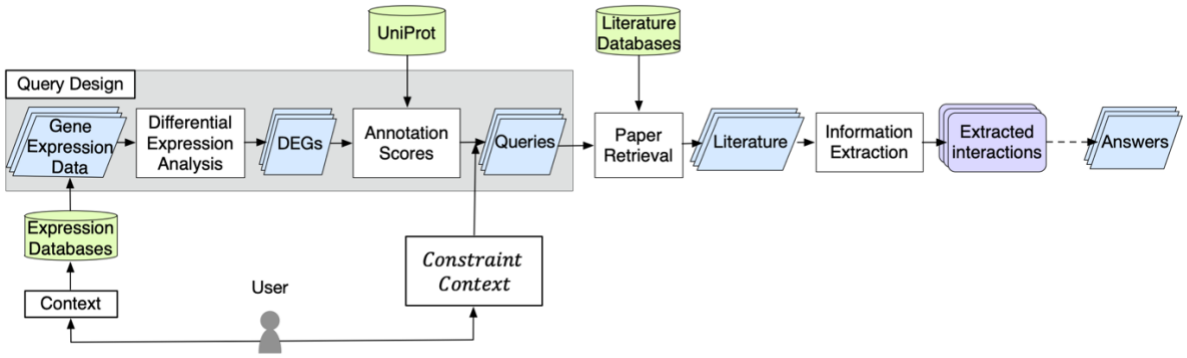


Figure 9. The automated query design process for information retrieval in biomedical research.

Once the dataset file is selected and input by the user, the proposed query design method identifies genes that are differentially expressed in the context of interest (e.g., disease state, cell line, etc.), compared to the control. The RNA-seq technique provides insight into the transcriptional activity of a cell population and reveals the number of gene transcripts present at a single point in time. For any gene *X*, its differential expression was computed as the $\log_2$ fold change between the amount of its transcript ($\boldsymbol{X_{transcript}}$) in two scenarios, control ($\boldsymbol{X^{control}_{transcript}}$) and disease state ($\boldsymbol{X^{disease}_{transcript}}$), a common method for measuring changes in gene expression [103]. Since the magnitude of the change from the control is the relevant measurement, and not the direction of the change (i.e., increase or decrease), the absolute value of the change was used:

$$\boldsymbol{d_X} = \left|\boldsymbol{\log_2 \frac{X^{disease}_{transcript}}{X^{control}_{transcript}}}\right| \quad (1)$$

After determining the $\boldsymbol{d_X}$ value for all transcripts in the selected RNA-seq dataset, the transcripts were sorted in a descending order of their $\boldsymbol{d_X}$ values (i.e., descending magnitude of change). A threshold for the $\boldsymbol{d_X}$ value was set to ensure that all genes used as query terms are relevant to the dataset context. Specifically, 2.0 was set as the threshold, that is, any transcripts that have $\boldsymbol{d_X < 2.0}$ were removed from the sorted list. The standard threshold for $\boldsymbol{d_X}$ is usually 2.0 or 1.5 [103], based on what a cell biologist would consider notable or likely due to the effect of the disease or altered state, and not just noise in gene expression. While $\boldsymbol{d_X \geq 2.0}$ was the chosen threshold for a DEG, the user can adjust this threshold to suit the research context (i.e., diseases or cell types with more or less DEGs than expected). The transcripts remaining in the sorted list were considered DEGs. As probable indicators of a disease state, these DEGs become candidates for query terms. To give an estimate of an expected size of the sorted DEG list, previous work on analyzing many RNA-seq datasets over a wide range of conditions, including disease, tissues, cell types, drug treatments, etc., has shown that the median number of DEGs (with $\boldsymbol{d_X \geq 2.0}$) per dataset is 92 [104]. However, as many as 10,000 DEGs per dataset were also observed, although rarely. Reasonably, dozens to hundreds of DEGs (with $\boldsymbol{d_X \geq 2.0}$) were expected, out of the 20,000+ genes in an RNA-seq dataset.

3.2.2 Selection of query terms

Using all DEGs with $\boldsymbol{d_X > 2.0}$ to formulate a query is still not practical, as there can be tens or hundreds of such DEGs. Instead, to determine the number of DEGs to be used as query terms, this method estimates the number of papers that would be retrieved from a literature database when using the query formed from these terms. For example, in PubMed, the "popularity" of genes varies widely: *TP53* is a well-known oncogene with over 100,000 papers found in

PubMed, and therefore, any query containing "p53" will return more papers than a query using a novel gene.

The UniProt annotation score (described in greater detail in Section 2.4.1) is used to estimate the impact of each DEG, as a possible query term, on the number of papers retrieved. The annotation score is one possible measure of how established a gene is in the literature. To decide which DEGs to include as query terms, both the annotation score and the $\boldsymbol{d_X}$ value are considered. The combination of these two measures allows the design of queries for different objectives or tasks, for example, to search for literature that contains a few well-known (high annotation score) proteins, or many novel or unstudied (low annotation score) proteins. Furthermore, by incorporating the UniProt annotation score to choose terms, this method automates query design that will lead to a selection of a manageable number of papers. Additionally, the number of papers found in a literature database as a result of the query will be different for each user depending on the input dataset, annotation score, and the addition of new publications in the literature database, and so this method allows to tailor the query design process to the user's research goals. The DEGs that are selected to be used in a query are referred to as *query term DEGs* from this point forward.

Different research tasks, paper contexts, and datasets will require a different number of papers to be read. Therefore, this method allows the user to provide an additional input, $\boldsymbol{Constraint}$, which will influence the number of papers selected for reading. The $\boldsymbol{Constraint}$ input can be either categorical or a discrete number greater than 0, and is used in this method to determine the cut-off parameter, $\boldsymbol{C}$. The cut-off $\boldsymbol{C}$ value is in turn used to select those DEGs that will be included in the query. Starting with the DEG that has the largest $\boldsymbol{d_X}$ value, DEGs were added to the query term list, as long as the sum of their annotation scores is smaller than or equal to the cut-off value $\boldsymbol{C}$.

There are three categories to indicate the level of automated reading needed to comprehend all information in the paper set. The first category, "human-readable", results in a selection of a small number of papers, suitable for a human to read in a short time (e.g., hours). The second category, "automation suggested", leads to a medium number of selected papers that is possible for a human to read (e.g., days), but more practical if processed by machine reading. The third category, "automation required", results in a large number of selected papers, only practical for machine reading.

Allowing for two different ways to enter the ***Constraint*** input provides additional flexibility. If the user knows exactly which value they want to use for the cut-off parameter, they can directly enter it. However, in the research process, the users may sometimes be interested in exploring a smaller subset of relevant papers, or doing a more comprehensive exploration of the topic, and the three categories listed above are useful in such cases. The values of the parameter $\boldsymbol{C}$ that correspond to the three categories are listed in Table 10.

Table 10. User-input categories, the corresponding cut-off parameter C for the annotation score sum, as well as the expected maximum and minimum number of query term DEGs. (These values do not account for DEGs with no entry in the UniProt Database).

user-input category	C	expected min # of DEGs	expected max # of DEGs
human-readable	15	3	15
automation suggested	35	12	35
automation required	60	20	60

While these values are set internally in the code, they could be easily changed to better suit different domains or research goals. For example, for a "human-readable" reading output, the cut-off value $\boldsymbol{C}$=15, and following the method for selecting query term DEGs given $\boldsymbol{C}$, this could result in as few as 3 query term DEGs (all with annotation score 5) or as many as 15 query term DEGs (all with annotation score 1).

To this end, it is worth noting that not all DEGs are always found in UniProt, and therefore, the DEGs without a corresponding UniProt entry are assumed to have annotation score value of 0. As this is possible even for DEGs with large $\boldsymbol{d_X}$ value, this could lead, in rare cases, to the actual number of query term DEGs exceeding the cut-off value $\boldsymbol{C}$ (e.g., this would be 15, for the example above). While, in theory, the number of DEGs with $\boldsymbol{d_X \geq 2.0}$ and annotations score of 0 could potentially be very large, this case was not encountered. Moreover, the experiments have shown that allowing for DEGs with annotation score 0 to be added to the query term list does not significantly increase the number of selected papers, while at the same time can lead to the retrieval of papers with very novel disease mechanisms. Table 10 provides the $\boldsymbol{C}$ values that were used for the three user-input categories, and the corresponding typical minimum and maximum number of query term DEGs. It is important to note that the typical minimum and maximum numbers shown in Table 10 are easily determined from $\boldsymbol{C}$ values, as they take into account only those genes with an annotation score greater than 0, and thus the actual maximum number of DEGs could sometimes be even larger.

Once the list of the query term DEGs is determined, their official gene names (e.g., *TP53, BRCA1, EGFR*) are combined with a logical **OR**, thus allowing any paper that includes at least one of the query term DEGs to be selected. The logical **OR** was used to retrieve the maximum number of relevant papers for each query, since a logical **AND** would make the query more specific, and so restrict the number of papers. Other combinations of logical **AND** and **OR** between the terms in the query are possible and could be informed by the user or inferred if relevant information is available.

Furthermore, since queries should be able to focus on a particular context, *Context* was added to this logical expression as a necessary condition, that is, it is combined with the other terms using a logical **AND**:

$$(\boldsymbol{gene_1}\ \textbf{OR}\ \boldsymbol{gene_2}\ \textbf{OR}\ \ldots\ \boldsymbol{gene_N})\ \textbf{AND}\ \boldsymbol{Context} \quad (2)$$

where each $\boldsymbol{gene_i}$ (i=1,..,N) is the official gene name of one of the N query term DEGs. To extract relevant interactions, only papers that mention the context of interest were included. It is important to note that one context may have multiple aliases (e.g. “coronavirus”, “COVID-19”, and “SARS-CoV-2” are all referring to the same disease). The user can increase the scope of the retrieved papers by combining all possible context aliases with a logical **OR**.

3.2.3 Using queries in disease explanation

All machine reading statements from the INDRA database (described in Section 2.4.3) were retrieved if they were associated with at least one paper in the reading set. Although the query term DEGs that were selected following the method described in in the previous sections are likely to participate in these extracted interactions, it is important to note that the interactions output by readers will include many other relevant genes and proteins. Thus, these extracted interactions are expected to provide the information on intracellular signaling networks that is potentially critical for the context originally selected by the user and included as a term in the generated query (equation 2).

The types of biological processes and signaling pathways these interactions are involved in reveal the relevance of extracted interactions. PANTHER [105] was used to calculate enriched GO terms in the protein-protein interactions within the interaction sets for each query. To assess whether enriched GO terms are similar, NaviGo was used to calculate the Resnik similarity score

between all GO terms (described in [106]). The signaling pathways and biological processes that were represented in the paper sets for each query were studied by determining highly enriched GO terms.

3.2.4 Query design case studies

To demonstrate the usefulness of the automated query design methodology, results are shown for four different contexts. For each context, two queries were automatically designed, one with an expected large number of output papers, and one with an expected small number of output papers. These results illustrate how DEGs can be used to formulate queries that output relevant papers, and how the annotation score affects the volume of papers. These results also show that the papers contain interactions that are closely related and are involved in the same GO biological processes.

Four publicly available RNA-seq datasets were selected using the Expression Atlas [102]. These four datasets provide gene expression data for both control and disease state in SARS-CoV-2 [107], ulcerative colitis [108], glioblastoma multiforme [109], and thyroid carcinoma [110]. All four datasets express transcription in transcripts per million (TPM) and include the $\boldsymbol{d_X}$ values computed for the disease state with respect to the control state. The following experiments used the $\boldsymbol{d_X}$ values that were provided with selected datasets. These case studies cover three substantial topics in biomedical research – autoimmune disorders, cancer, and viral infections. Using differential gene expression data from these diseases illustrates how biological data can provide valuable information for automatically designed targeted queries.

3.2.5 Selection of queries

To design a query that retrieves a small reading set, as discussed in Section 3.2.2, the effect of the cut-off value $\boldsymbol{C}$=15 for the annotation score sum was explored, and to design a query that retrieves a large reading set, the cut-off value $\boldsymbol{C}$=60 was used (Table 10). The queries generated for all four contexts for these two cut-off values are listed in Table 11. Notably, the same cut-off value $\boldsymbol{C}$for different datasets may result in queries with a different number of terms. This can be explained by the UniProt annotation score of the top (with large $\boldsymbol{d_X}$) DEGs in the datasets. Due to differences in experiment techniques, environmental conditions, or other factors, gene expression datasets from different samples and labs will likely show differences in the top DEGs. Consider a hypothetical example where queries are formulated based on two pancreatic cancer datasets, A and B, and choose the cut-off $\boldsymbol{C}$=10. For dataset A, this value is achieved after adding two DEG query terms, since the DEGs with highest $\boldsymbol{d_X}$ values are P53 and MDM2, which are both very well-known proteins with an annotation score of 5. For dataset B, the threshold is not passed until five DEG query terms are added. The top five most differentially expressed genes are small non-coding RNAs, which are generally poorly studied, and each has an annotation score of 2.

Table 11. Six automatically formulated queries for three diseases. Each disease has two associated queries, which are expected to retrieve different sized reading sets.

Context	# of DEGs with $d_X \geq 2.0$	C	Query
thyroid carcinoma	5026	15	**Q1**: (GABRB2 or LIPH or KLHDC8A or LIPI) and "thyroid carcinoma"
		60	**Q2**: (GABRB2 or LIPH or KLHDC8A or LIPI or LINC02471 or PRR15 or MTRNR2L12 or CIDEC or RTL4 or SLIT1 or ZCCHC12 or TRPC5 or LRP4 or RXRG or METTL7B or CDH3) and "thyroid carcinoma"
ulcerative colitis	1476	15	**Q3**: (AL035661.1 or HP or NECAB1 or MCEMP1) and "ulcerative colitis"
		60	**Q4**: (AL035661.1 or HP or NECAB1 or MCEMP1 or ANKRD22 or ARG1 or BMX or MMP9 or S100A12 or SCART1 or SLC2A14 or SLC1A3 or SLC12A5-AS1 or OLAH or ACHE) and "ulcerative colitis"
COVID-19	32	15	**Q5**: (MX1 or DDX60 or PARP9) and (SARS-CoV2 or COVID-19 or coronavirus)
		60	**Q6**: (MX1 or DDX60 or PARP9 or DDX58 or HELZ2 or CMPK2 or OAS3 or STAT1 or HERC6 or DTX3L or IFIT1 or SAMD9 or AL445490.1 or TAS2R4 or AC147651.1 or AC004253.1) and (SARS-CoV2 or COVID-19 or coronavirus)
Glioblastoma	3300	15	**Q7**: (HOXD9 or PLA2G2A or HOXD10) and (Glioblastoma or GBM)
		60	**Q8**: (HOXD9 or PLA2G2A or HOXD10 or HOXD13 or HOXA5 or HOXD8 or DLGAP5 or HOXA10 or SAA1 or HOXC10 or AC092017.1 or AC011742.1 or AL160286.1 or MIR663AHG or ELL2P1 or TOP2A or IGHA1) and (Glioblastoma or GBM)

3.2.6 Paper retrieval

Papers were retrieved using PubMed as the most up-to-date and comprehensive source for biomedical literature, without any filters for article type, year, or journal. However, results were restricted to only those papers with valid PMCIDs, to ensure that all papers can be processed with state-of-the-art machine readers. Once queries had been formulated for each use case, they were used to search PubMed. Figure 10a shows the number of papers retrieved as a function of how many of the top DEGs are used as query terms. As expected, as the number of terms increase, so does the number of retrieved papers. However, many query terms, in conjunction with the context term, add no additional papers to the reading set. This indicates that some of these DEGs have not been explored much or mentioned in papers in the context of the relevant disease, and therefore, they may be a fruitful avenue for exploration.

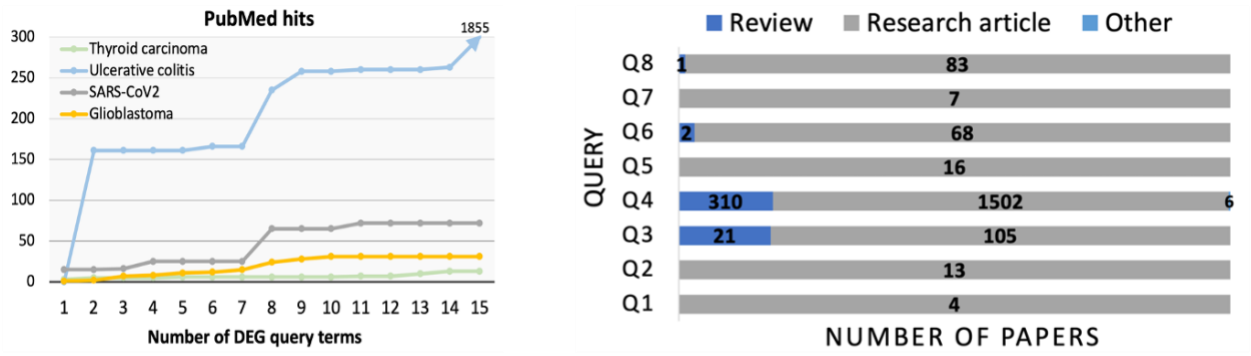


Figure 10. Number of papers found in PubMed, based on how many of the top DEGs were used as query terms. (b) Distribution of paper types by query.

Figure 10b shows that, as the number of extracted papers in the reading output increases, the distribution of article types also changes. The composition of the reading set was studied by classifying each paper as either a research article, review, or other (books, documents, etc.). In large reading sets, reviews are slightly more common than in small reading sets, which is due to one or more query term DEGs having better representation in PubMed. Well-studied genes and proteins are more likely to be included in reviews than novel, relatively unknown genes. Since the scope of reviews and research articles differ drastically, they are expected to contribute differently to the number of extracted interactions.

3.2.7 Validation of Extracted Interactions

The statements from the INDRA database were analyzed to validate the paper sets retrieved from each query. Figure 11a shows the number of extracted interactions for each query. The number of interactions is dependent upon the number of papers, as well as the representation of the context and DEG query terms in PubMed. The top 10 enriched GO terms were determined for each query and sorted using the false discovery rate (FDR) [111]. The average Resnik similarity

score between the top 10 GO terms for each of the eight queries was calculated, where a higher score indicates more similarity between GO terms. Finally, Figure 11b shows the percent of DEG query terms that are present in the list of extracted interactions. These results, taken together, show that these queries retrieve papers that contain relevant signaling events that can be interpreted by machine readers, and describe highly related biological processes. In general, this method of increasing the cut-off value $\boldsymbol{C}$ not only retrieves more papers, but it also increases the number of signaling events extracted by readers, without a sizeable cost to relevance, as assessed by GO term semantic similarity.

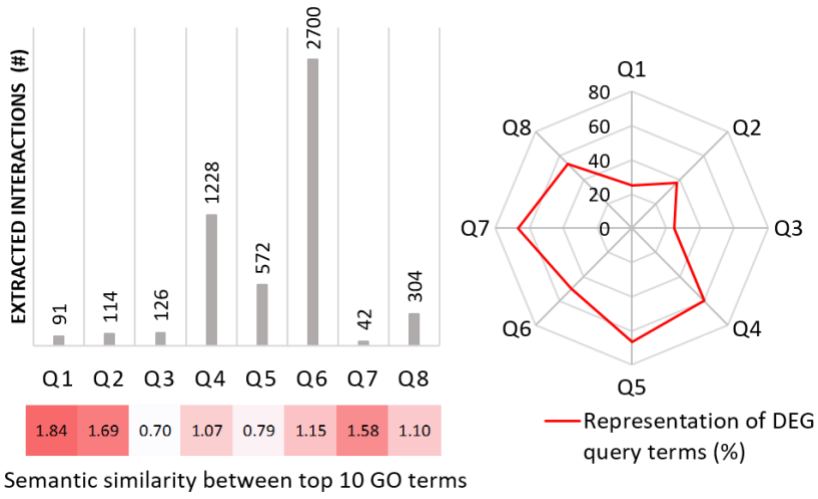


Figure 11. Number of interactions extracted from INDRA for each query, as well as the average pairwise Resnik similarity score for the top 10 enriched GO terms (left), and the percent of DEGs used as query terms in each case study that are present in the set of extracted interactions.

3.3 Managing Interaction and Network (re-)Usability through Evaluation of Trustworthiness (MINUET)

This section describes a methodology for generating complete and accurate cellular signaling and interaction networks, based on topological features as well as existing data on interactions from several databases. The network curation tool (MINUET) can be used in the process of automated network verification, a much-needed step for fast and accurate network curation. To this end, the main contributions of this work are a methodology to verify causal network models of cellular signaling.

3.3.1 MINUET workflow

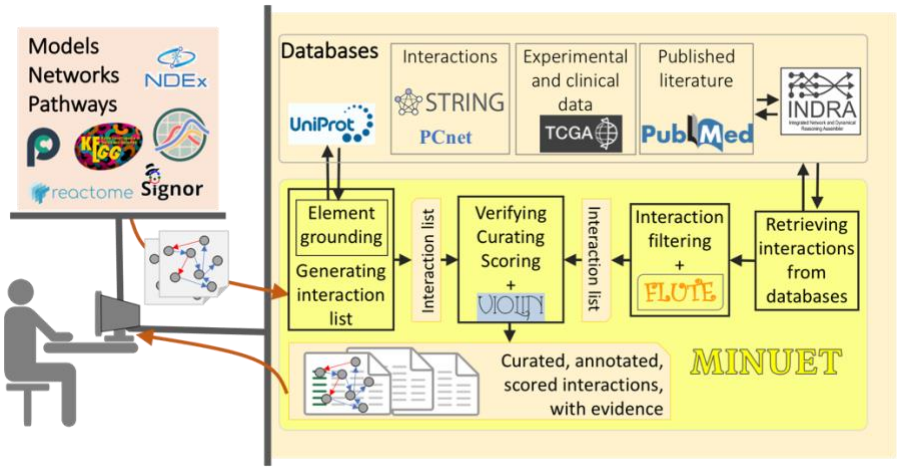


Figure 12. MINUET workflow.

MINUET (Managing Interaction and Network (re-)Usability through Evaluation of Trustworthiness) is a platform for automated network verification and curation. MINUET is conducts model verification that relies on both knowledge and data, the vast published literature and publicly available databases; it is *flexible* as it can conduct both context-aware and context-

independent verification; it is *versatile*, allowing users to conduct in-design verification during model creation, post-design verification of existing models, or a comparison of models to verify them against each other; finally, it is *fast*, due to its automated steps for retrieving and comparing interactions. MINUET contributes to "the four r's" in several ways: it facilitates and evaluates *reliability* and *reusability* of models and information; it increases the potential for *reproducibility* of predictions by identifying structural differences between models; and it assesses the *replicability* of outcomes and observations by collecting evidence from knowledge and data sources.

As shown in Figure 12, the main input for MINUET is a model network, that is, a list of all its entities and interactions. These networks can be found in model repositories, interaction databases, and metadatabases (see Sections 2.4.2, 2.4.3, and 2.5.3). MINUET utilizes the information from several databases to confirm the network structure, by retrieving information about network nodes, retrieving relevant literature, and providing support for interactions. The INDRA database, PCnet, and FLUTE (described in greater detail in Section 3.1), are used to verify the network edges. Information about network coverage (how many frequently mutated, therefore highly important, genes are included) is supplied by TCGA.

The first step in the MINUET workflow is to ground entities by finding their unique identifiers (IDs), and this process is dependent on entity type. For genes and proteins, the network verification method utilizes the UniProt API to automatically determine their standard IDs. Specifically, gene and protein names were standardized by mapping to both the Human Genome Nomenclature Committee (HGNC) identifier and the HGNC-approved name. These two ID types are very common and allow for comparison to many online resources. The automated ID mapping

for genes and proteins allows for fully automated verification of gene regulatory networks, protein-protein interaction networks, and in general, many cell-signaling networks.

3.3.2 Automated network curation with MINUET

The automated network curation method has two main steps, literature search in PubMed, followed by the use of INDRA API to retrieve all statements. To assemble a network that includes relevant, commonly affected pathways for a given context (e.g., disease, cell type, tissue, or a biological state), a query was used as an input to the network assembly method. This query has two parts connected with logical AND, and each part is a list of terms connected by logical OR operator. One list of terms leads to the retrieval of papers from the desired context, and the other list of terms refers to relevant signaling networks and pathways (e.g., "signaling OR network OR pathway OR cascade OR interaction OR regulation"). For well-studied contexts or broad queries that return a large number of papers (in the order of 10,000+), the most relevant papers were selected, as determined by the PubMed's Best Match feature [112]. The second step of the automated assembly method takes as input the standardized paper identifiers for the set of context-specific papers found in the first step and searches the INDRA database using its API to find all statements (i.e., interactions) in this paper set.

The returned statements were limited to those that have only two distinct entities. This eliminates statements that represent an edge joining more than two nodes, such as a statement describing complex formation. The number of entities in a statement was restricted in order to be able to compare all statements to PCnet and other existing networks, which represent steps in a signaling pathway as one-to-one interactions. As a result, the automatically assembled network is

composed entirely of INDRA statements, where the entities are the nodes, and the interactions stored in the INDRA statement are edges.

Several filters to the automated curation workflow, including those used in the network verification approach, improved the quality of this network. These filters are based on the information included in each INDRA statement. First, the belief score can be applied as a cut-off, since it is determined with respect to the amount of evidence to support the interaction; an interaction with a higher belief score is less likely to be a false positive (an invalid or non-existent relationship between two entities). To filter by belief score, the cut-off was 0.85, and all interactions below that score were discarded. Next, MINUET can also filter by interaction type, that is, direct or indirect, as stated by INDRA. For some interactions, INDRA contains evidence on whether an interaction is direct or indirect. Other statements do not contain any evidence on the interaction type. By selecting for only direct interactions or those with high belief scores, the final network contains fewer low-confidence interactions.

MINUET automatically compares grounded model networks to INDRA statements. This step takes one input parameter, the type of network (directed or undirected). For directed networks, it iterates through all interactions in the network, and retrieves all statements that match the entity identifiers, as well as the direction and sign. For undirected networks, it retrieves statements that match the entity identifiers without checking direction or sign. The output of this step is INDRA statements that support interactions in the input network.

Besides automatically comparing the input network with INDRA, MINUET also automatically compares the network to all interactions in PCnet. By comparing model interactions to PCnet interactions, MINUET identifies which interactions in the model have support from multiple curated signaling networks. While PCnet interactions are highly supported, they are

undirected, and independent of context. This distinguishes PCnet from INDRA, which contains many interactions that are directed and contextual, but have lower confidence. A local copy of PCnet was stored as a plain text file, which is also freely accessible and available for download from the Network Data Exchange (NDEx) [83].The first step in this case is a conversion of the directed network to an undirected one, since PCnet contains only undirected interactions. This method then iterates through all model interactions and compares them to all PCnet interactions. Finally, it outputs a list of all interactions within the signaling network that are verified by PCnet. PCnet can be used by itself to verify a network, or in conjunction with INDRA.

FLUTE is capable of selecting high confidence interactions and filtering out many incorrectly read interactions within a reading set (see Section 3.1). FLUTE encompasses several types of biological entities including proteins and genes, chemicals, and biological processes, while PCnet is composed of only proteins. FLUTE is also able to provide a score for the interaction confidence, unlike PCnet. In contrast to INDRA, the FLUTE database contains interactions with a high level of human oversight. FLUTE provides an extra level of scrutiny over INDRA, while still being less restrictive than PCnet. The FLUTE tool was specifically designed for interaction filtering, unlike PCnet, which is a network. Table 12 summarizes the characteristics of the three sources, INDRA, PCnet, and FLUTE, that are used in the automated verification method.

Table 12. Comparison between INDRA, PCnet, and FLUTE.

	INDRA	PCnet	FLUTE
Machine reading results?	Yes	No	No
Database results?	Sometimes	Yes	Yes
Interaction scoring method?	Yes	Yes*	Yes
Directed interactions?	Sometimes	No	No
Interaction mechanism?	Sometimes	No	No
Manual curation?	Sometimes	Yes	Sometimes

The methods used for automated network verification can also be utilized for post-design curation, thus providing verified interactions by construction. In cases where an existing (baseline) model, i.e., its underlying network, fails to capture the full detail of the studied system, an automated extension method retrieves new interactions (extensions) to improve network scope. The goal is to explore whether these methods can help identify these new important entities and interactions to be included in the baseline network.

4.0 GBM stem cell model

This chapter describes an executable model of GBM stem cells that is a general representation of key kinase signaling in these cells. Besides being grounded in GBM-specific literature, and encompassing over a dozen critical pathways, this model is also capable of being parameterized based on biological data. For three GBM stem cell lines, there is available RNA-seq, RPPA, whole exome sequencing, and *in vitro* kinase inhibition results. These three cell lines (MGG8 [113], GS11-1 [114], and GS6-22 [96]) are all patient-derived, and show different patterns of gene expression, protein phosphorylation, and mutations. These cell lines also exhibit morphological characteristics of cancer stem cells.

4.1 Curation of the GBM signaling network

The goal is to create a standard "baseline" model that can be parameterized for any genomic alteration. The interactions themselves should be based on common behavior of GBM stem cells, so that the baseline can be adapted to fit a wide range of genomic profiles. To create signaling network models, a manual assembly process outlined in Figure 13 is commonly used. The first step is a selection of relevant biological system components that have been shown to play a role in the disease of interest. These components are supplied by a number of sources, including expert knowledge or different publicly accessible databases. These databases may curate canonical signaling pathways, such as KEGG [24] or PANTHER [105], or they may collect data on individual interactions, such as STRING or BioGRID. Interaction databases provide curated,

often high-confidence data on signaling pathways in disease or normal cell conditions [66]. A disease network can be supplemented with experimental data that is cell- or patient-specific, such as genes or proteins that have differential expression, somatic mutations, or altered signaling capacity. The final list of biological molecules composes the nodes of the network. The edges are created between nodes based on existing data from literature and interaction databases.

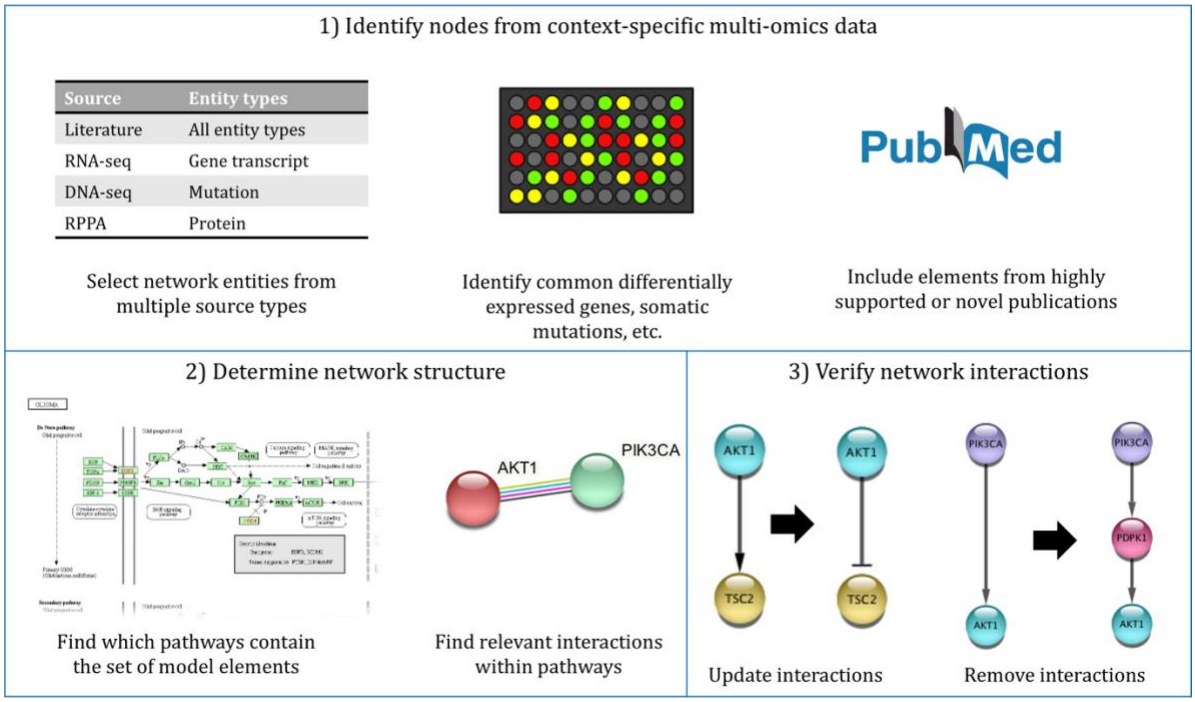


Figure 13. Manual network curation process.

The network focused on pathways upstream of cell cycle or apoptosis, since these pathways directly affect tumor survival, and proteins within these pathways are often implicated as either tumor suppressors or oncogenes [115-117]. The network also incorporated several pathways critical for growth and development (Hippo [33, 118], Hedgehog [119], Notch [15, 98, 120], etc.). In order to understand and explain the response mechanisms to kinase inhibitors, the model includes the key signaling elements and pathways downstream of 11 kinases, CDK6, AKT, EGFR, ERK, GSK3B, Chk1/2, AURA/B, PKC, PI3K, PDGFR, and VEGFR.

A simplified view of the baseline model is shown in (Figure 14). This baseline model has 415 elements and 531 interactions between them, and is represented in the BioRECIPE format (see Appendix A- GBM model rules). The model includes the following element types: protein (amount), protein (active), genes, chemicals, biological processes, and mutations. To accurately represent events such as gene transcription and protein translation, the motifs described in [121] were used. New motifs were developed to standardize the representation of interactions within the model, and these motifs are listed in Appendix B– Updated motifs. Subcellular compartments such as the mitochondria and nucleus are represented to accurately model Cytochrome C release and gene transcription, respectively. There are several key observables, mainly proliferation, apoptosis, and cell cycle progression. Note that proliferation is regulated by two elements, cell cycle progression and apoptosis.

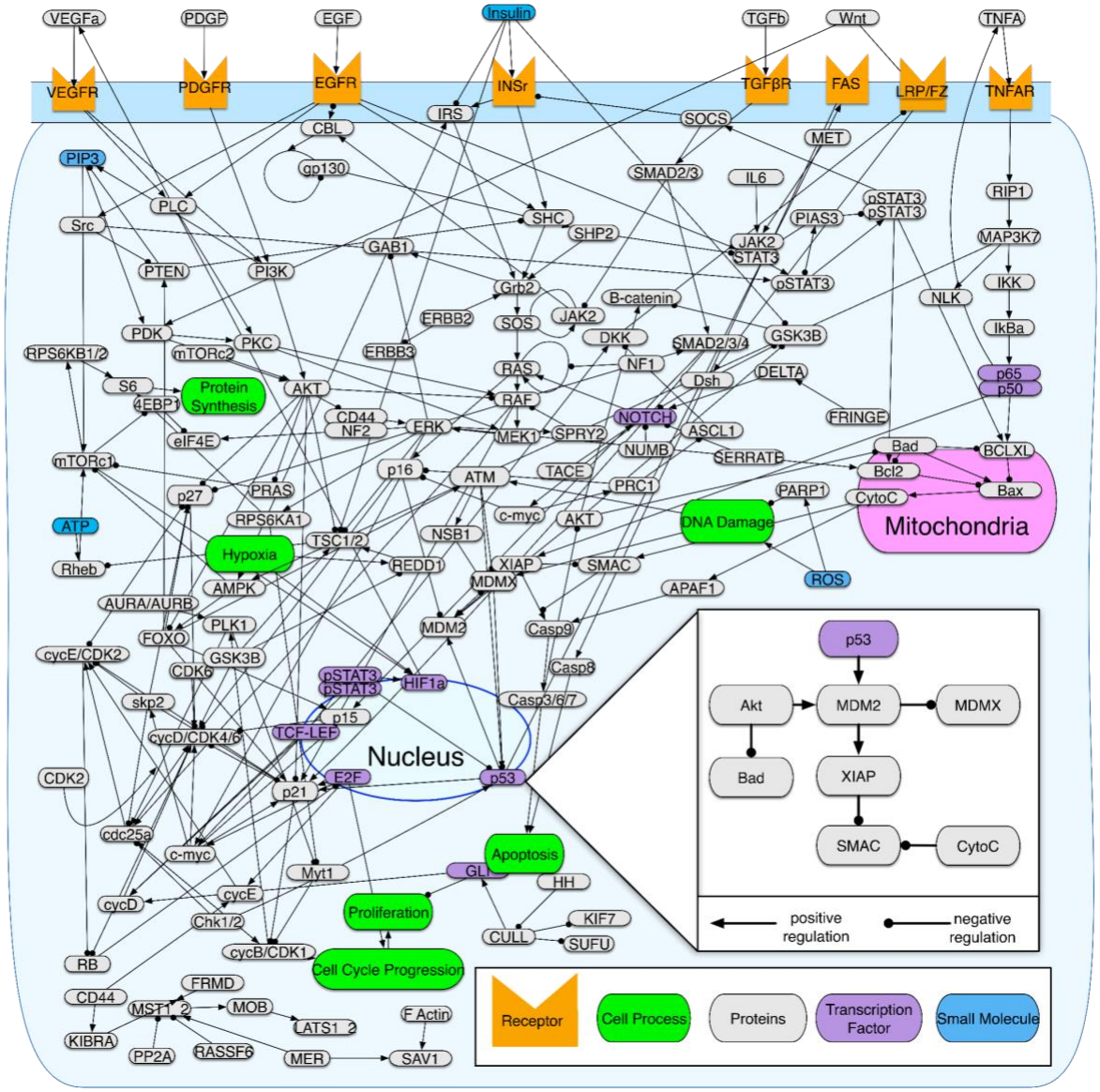


Figure 14. GBM stem cell signaling network.

4.2 Verification

4.2.1 Verification with literature and database resources

The GBM network was verified using MINUET (Section 3.3). Figure 15 shows the overlap between the INDRA DB, GBM network, and PCnet, the intersection between each two of them, as well as the intersection between all three. For these purposes, only interactions that could be verified by database resources were considered. Any interaction representing gene transcription or translation was not verified, and any non-PPIs were discarded, due to their low representation rate in INDRA, leaving 279 interactions. Each of the 279 interactions in the GBM network is found in INDRA, confirming the existence of these interactions, as well as their direction and sign. Consequently, all model interactions in the GBM network that are supported by PCnet are also present in INDRA, and they form GBM_{PCnet}. Thus, the set of 208 interactions in the GBM_{PCnet} network confirmed by PCnet and INDRA, indicates that the majority of interactions in GBM are both high-confidence and have a supported mechanism.

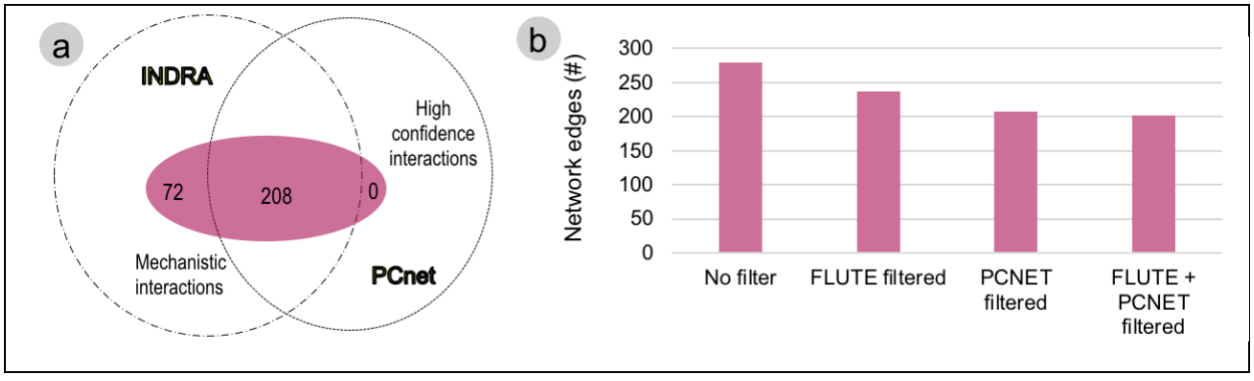


Figure 15. (a) Overlap between the GBM network, INDRA, and PCnet, (b) size of each four GBM networks.

4.2.2 Verification against existing models

We compared the literature support of the GBM network versus other published GBM networks. We examined two networks publicly accessible from literature- Jean-Quartier et al 2020 [122] and Tuncbag et al 2016 [123], and two networks from databases- the KEGG GBM network [24] and the SIGNOR GBM [58] network. Additionally, we retrieved from NDEx [83] a TCGA RNA-miRNA interaction network, $TCGA^{miRNA}$. It should be noted that $TCGA^{miRNA}$ is not a mechanistic signaling network like the others; rather, it is a correlation network derived from gene expression data. Table 13 summarizes the characteristics of these five existing GBM disease networks, as well as GBM_{PCnet} described in the previous section.

Table 13. Characteristics of GBM networks.

Network	Node #	Edge #	Average Clustering Coefficient	Connected Components	Hub Nodes	
					#	%
GBM	134	279	0.07	1	18	13.43
GBM_{PCnet}	118	207	0.06	1	11	9.09
Jean Quartier et al 2020	538	911	0.24	5	26	4.83
$TCGA^{miRNA}$	278	2287	0.00	1	207	74.46
SIGNOR GBM	26	46	0.10	1	11	42.31
Tuncbag et al 2016	191	242	0.043	1	8	4.29
KEGG GBM	45	47	0.10	3	0	0.00

Figure 16a-d shows the INDRA, PCnet, and the GBM network overlap with Jean Quartier et al 2020, Tuncbag et al 2016, KEGG GBM and SIGNOR GBM networks, respectively. We find that, while the GBM network outperforms existing GBM networks in terms of INDRA representation, PCnet representation is more comparable. JeanQuart20 has the highest percentage of interactions represented in PCnet. $TCGA^{miRNA}$ is again, the least supported, since it is a

correlation network of gene-miRNA interactions, which has no presence in PCnet, and only infrequent mentions in INDRA.

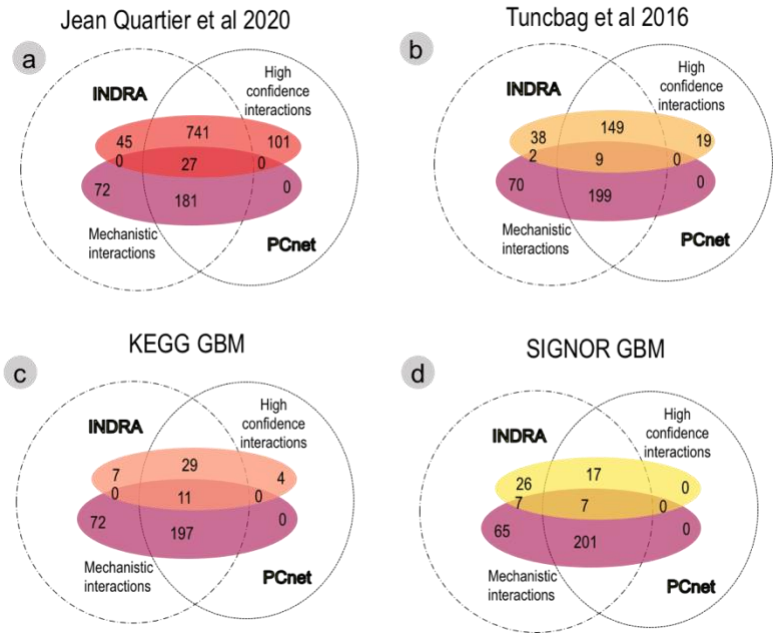


Figure 16. Overlap between INDRA, PCnet, the GBM network (purple), and (a) Jean Quartier et al 2020, (b) Tuncbag et al 2016, (c) the KEGG GBM pathway, and (d) the SIGNOR GBM pathway.

Additionally, we compared the overlap between each pair of GBM networks in terms of shared interactions (Figure 17). While all five networks are intended to address the same disease signaling network, we find that there is very small overlap. For example, the maximum overlap is between the GBM network and Jean Quartier et al 2020, and even this overlap is only 28 interactions, making it 10.04% of the GBM network and 3.07% of the Jean Quartier 2020 network. This disparity is most likely due to differences in represented pathways.

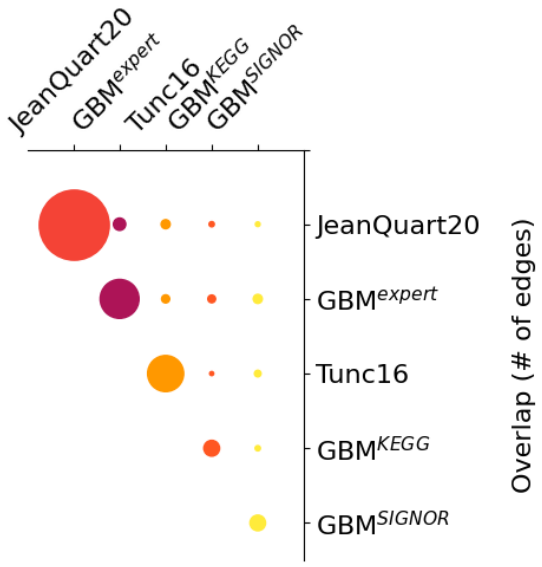


Figure 17. Overlap in number of interactions between the GBM network and four other GBM networks.

4.2.3 Verification using graph features

Properties independent of node and edge identity can also be used to verify the network. While these numbers alone cannot verify the network, they can provide a measure of how useful the network is, and whether it resembles a signaling network. These numbers confirm that the preliminary baseline network is well connected, without any disconnected nodes or isolated clusters. The network nodes form long paths, typical for signaling networks, instead of star-like clusters. Table 13 also lists the average clustering coefficient, and the number and frequency of hub nodes for the GBM and GBM_{PCnet} networks. The clustering coefficient is a metric of the connectedness of each node within a network [124]. The clustering coefficient for both networks is more indicative of a network that describes a real-world phenomenon than a randomly generated one [125]. Cancer signaling networks depend on the existence of hub nodes, which are highly susceptible to chemical inhibition. A hub node is defined as a node with >7 edges, which includes

both incoming and outgoing edges. Both manually assembled networks show a hub node frequency of approximately 1 in 10 (9-13%).

4.2.4 Verification with TCGA gene set

MINUET uses overlap between other networks or an interaction database to validate networks. This approach works well if there is an established consensus network for the pathway or network that can be used for benchmarking. However, there are many published GBM signaling networks, and they differ in both the size and content of the network. In addition to MINUET, the network can be verified with genes that are known to be involved in GBM signaling networks by comparing to the Cancer Genome Atlas (TCGA) [126] . For applications in GBM signaling, we will also provide real examples of verified signaling pathways between benchmark genes and network observables. The proposed approach will show that the GBM contains many benchmark genes, and highly-supported interactions between these genes.

Here, we produce a list of genes commonly indicated in GBM pathology (hereafter referred to as the "TCGA gene set"). TCGA-GBM will be the most comprehensive and reputable source for this list. We retrieved the list of all observed somatic mutations within the TCGA-GBM dataset, and ranked genes by likelihood of a somatic mutation. We chose the 15 genes most likely to be mutated as the TCGA gene set. We set the cut-off to be intentionally restrictive, to ensure that all genes in the TCGA gene set were commonly implicated in GBM signaling. For reference, all genes in this set are mutated in at least 10% of cases in TCGA-GBM.

The results of the TCGA gene set overlap with all GBM networks can be seen in Table 14. We find that the GBM network contains the largest overlap with the TCGA gene set (5 nodes respectively). We find that the edges connected to any node in TCGA gene set is also highly likely

to be verified by INDRA. Finally, we also see that the average INDRA belief score for these edge sets are high (>>0.65).

Table 14. Genes from the TCGA gene set and their overlap with GBM networks.

Network	PTEN	TTN	TP53	EGFR	NF1	PIK3R1	RB1	**total**
GBM network	✓		✓	✓	✓		✓	**5**
Jean Quartier et al 2020			✓	✓			✓	**3**
$TCGA^{miRNA}$								**0**
SIGNOR	✓			✓	✓	✓		**4**
Tuncbag 2016			✓	✓		✓		**3**
KEGG	✓		✓	✓			✓	**4**
total	**3**	**0**	**4**	**5**	**2**	**2**	**3**	

We also show the specific identities of overlapping genes in the GBM networks and the TCGA gene set in Table 14. We find that EGFR was the most commonly represented gene, followed by PTEN. TTN, which was mutated in at least 32.57% of cases, was not represented in any network. These genes and proteins may be involved in the canonical GBM signaling pathway (PTEN, EGFR, etc.) or they may be novel (TTN, NF1). These results demonstrate that even large networks (with hundreds of nodes and edges) may still leave out key genes and proteins implicated in disease.

4.3 Initialization

This section describes a systematic approach for parameterizing the baseline model using available data to model individual GBM stem cell lines. This method of initializing elements produces realistic model behavior that reflects differences between samples. For this method, we first perform fold-change analysis for all model genes. A two-fold change was considered significant, and indicative of a gene that is differentially expressed in at least one sample. For example, in Figure 18, we show the expression of three genes across three samples, where ***gene 1*** is differentially expressed in ***cell line A***. It is possible for a gene to be differentially expressed in all samples, as is the case with ***gene 2***. Therefore, we set the number of activity levels for all model elements based on the number of samples, since the maximum number of statistically significant pairwise comparisons is the number of samples. We then assume that the median value is the default for all genes. Any gene that is over-expressed is then matched with a corresponding activity level >1, and the reverse for under-expressed genes.

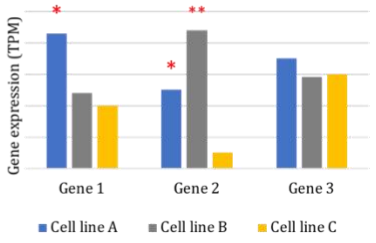
Algorithm Initialization maximizing cell line differences for element A

$i_0 \leftarrow \left\lfloor \frac{|cellLines|-1}{2} \right\rfloor$ *#Default initial value*

for j **in** *cellLines* **do**:
  for k **in** *cellLines* **do**:
    $foldC_{j,k} = gene\ A_j / gene\ A_k$ *#Pairwise gene expression comparison*
    $rank_{j,k} = \lfloor foldC_{j,k}/2 \rfloor$

$levels_A \leftarrow |\ \text{UNIQUE}(rank)\ |$ *#Levels needed to initialize element A*
$l_{min} = i_0 - \lfloor (levels_A - 1)/2 \rfloor$
$l_{max} = i_0 + \lfloor levels_A/2 \rfloor$

for j **in** *cellLines* **do**:
  if $\text{MAX}(rank_j) > 0$ **do**:
    $i_j^A = l_{max} - \text{MAX}(rank) - \text{MAX}(rank_j)$
  else:
    $i_j^A = l_{min}$



	Cell line A	Cell line B	Cell line C
Gene 1	2	1	1
Gene 2	1	2	0
Gene 3	1	1	1

Figure 18. Cell line specific initialization method.

To choose starting values for genes in the model, we use RNA-seq data. Many genes show different expression patterns in our three cell lines, which makes this data ideal for parametrizing our model. Another data type we can use to parameterize the model is whole-exome sequencing data. The mutations that we modeled in our preliminary studies are activating PDGFR, activating MDM2, and an inactivating PTEN mutation for MGG8 cells, and inactivating PTEN mutation for GS6-22 cells. Between whole-exome sequencing and RNA-seq data, we get a different set of starting conditions and active mutations.

4.4 Kinase inhibition experiment results

The *in silico* knockout experiments are both accurate and cell line-specific. The model reveals the mechanistic cause of kinase inhibition *in silico.* For example, the AKT inhibition scenario is visualized in Figure 19. On the left, we show the ten shortest pathways between Akt and its inhibitor (blue diamonds) and proliferation. Note that Akt regulates both apoptosis and proliferation, in contradictory ways. Without dynamic modeling, it would not be possible to determine the effect of Akt inhibition. DiSH simulation results predict that Akt inhibition will decrease proliferation, which is consistent with the results of the *in vitro* kinase inhibition data. Figure 19 (right) traces the effect of Akt inhibition. The trajectories show the control (no inhibitor, blue) and Akt inhibitor (orange). As a sanity check, we see that Akt has high activity in the control throughout the course of the simulation. However, Akt quickly drops to 0 if inhibitor is present. In the absence of Akt, which directly inhibits pro-apoptotic Bad, Cytochrome C (CytoC) release increases. This leads to an increase in a Caspase (Casp9), and finally apoptosis. Once a cell dies,

it can no longer contribute to tumor proliferation. This results in total inhibition of GBM stem cell proliferation.

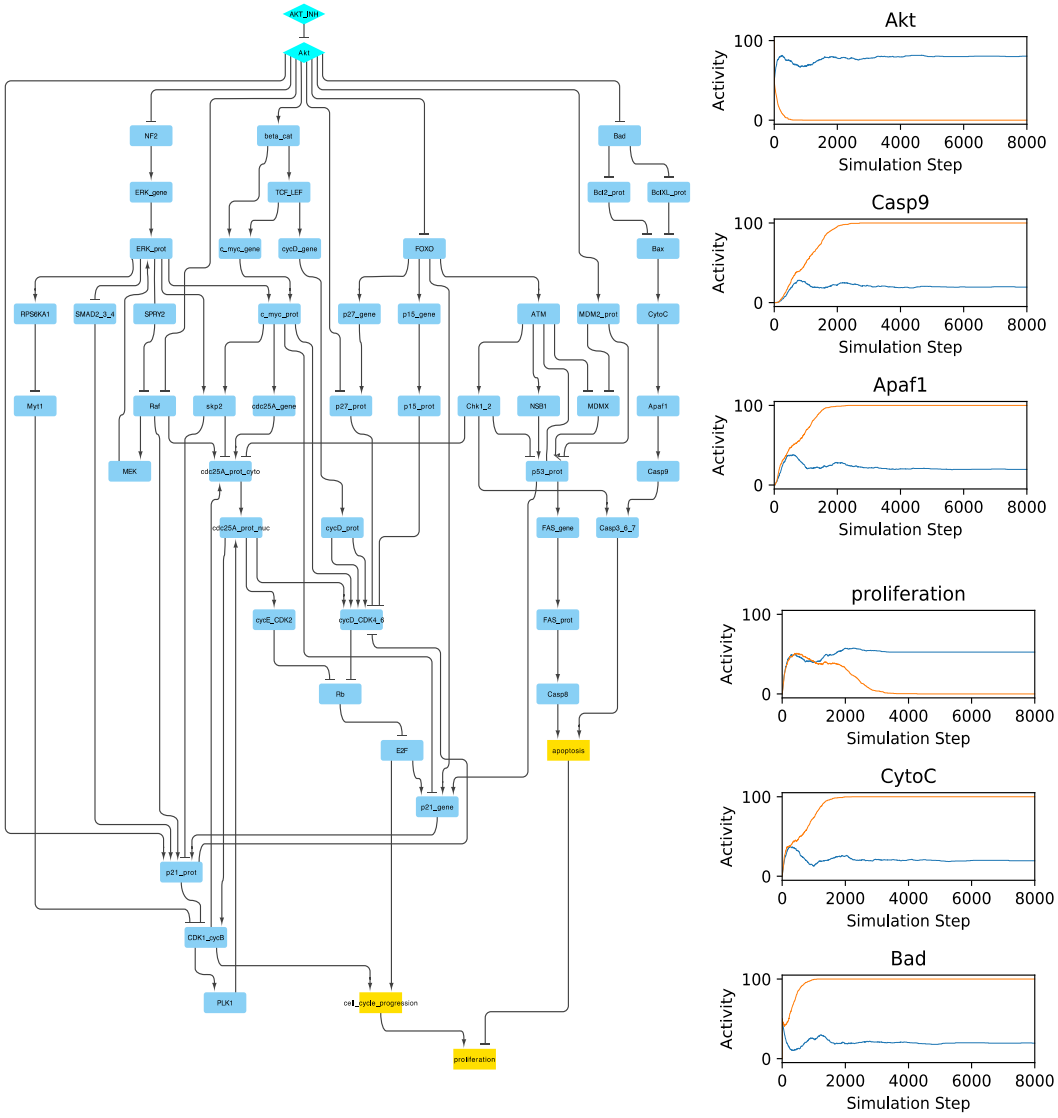
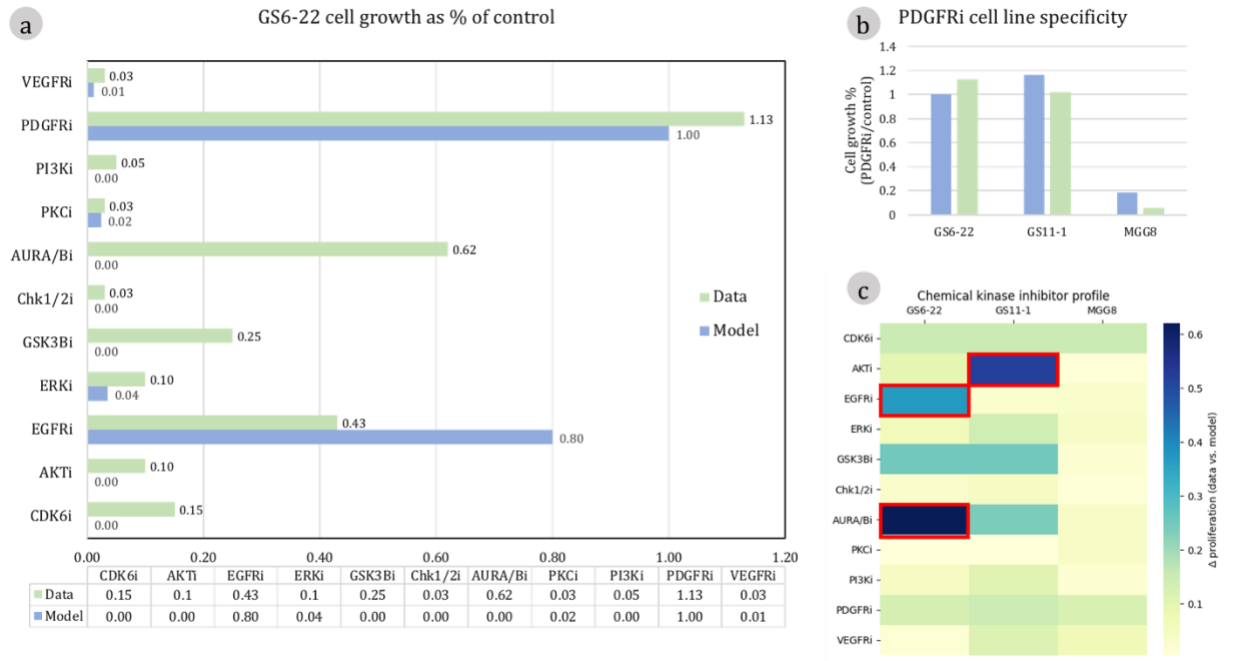


Figure 19. The top ten shortest pathways between AKT and proliferation. AKT and its inhibitor (teal diamonds) indirectly regulate several major observables (yellow). (Right) Simulation trajectories for the control (blue) and AKT inhibition (orange) that support the mechanistic conclusions. While there are several possible mechanisms that AKT can influence proliferation, the simulation results reveal that inhibition of AKT causes upregulation of pro-apoptotic factors (CytoC, Casp9, etc.) which inhibit proliferation.

Beyond the simple example of AKT inhibition, simulations were conducted considering 11 kinase inhibition scenarios for all three cell lines (33 in total), and in 27 of these scenarios the model accurately predicts whether kinase inhibition causes a decrease in GBM stem cell survival (Figure 20(a)). In other words, for the three cell lines, many in silico kinase knockout experiments agree with expected outcomes, and the model encapsulates many well-known pathways. The kinase knockout experiments are cell line specific as well (Figure 20(b)), where the model accurately predicts different responses to kinase inhibition across cell lines. For those few scenarios where the behavior is not expected (), it is often the case that the model accurately predicts the effect of kinase inhibition in two cell lines, but not one of them. This could be caused by a missing interaction in the model, or we are missing a differentially expressed gene or mutation that causes one cell line to react differently. However, these scenarios do indicate a potential avenue for further study *in vitro*.



	CDK6i	AKTi	EGFRi	ERKi	GSK3Bi	Chk1/2i	AURA/Bi	PKCi	PI3Ki	PDGFRi	VEGFRi
Data	0.15	0.1	0.43	0.1	0.25	0.03	0.62	0.03	0.05	1.13	0.03
Model	0.00	0.00	0.80	0.04	0.00	0.00	0.00	0.02	0.00	1.00	0.01

Figure 20. Kinase inhibition results for the three GBM stem cell lines.

5.0 Integration with DySE.

5.1 DySE pipeline.

This section shows how methods for model parameterization, literature selection, network verification, and reading filtration fit into DySE pipeline and enhance methodology for automation of model assembly, extension, and testing.

5.1.1 Comparison of models with VIOLIN

We used VIOLIN to compare the similarity between networks as measured by shared interactions. To show how interactions are represented differently between sources, we compare the GBM network to the KEGG GBM network, the SIGNOR GBM network, Tuncbag et al 2016, and JeanQuart20. Since $TCGA^{miRNA}$ has no overlap with any network, we do not show VIOLIN results for this network. First, we use VIOLIN to compare with manually curated networks (Figure 21). We find that both networks are far more likely to contain extensions than any other interaction type. We also find more corroborations than either contradictions or flagged interactions. These results indicate that The GBM network differs from other curated networks not due to errors in the sign or direction of the interaction, and that the difference is due to different node and edge sets.

Next, we used VIOLIN to compare against networks that were generated using automated methods (Figure 21). We find similar results as the curated networks, however, VIOLIN does identify many flagged interactions. The flagged interactions can be attributed to the presence of feedback and feedforward loops in the union of the GBM model and the automatically generated

networks. The manually curated networks are much smaller than the automatically generated networks, and so VIOLIN flags many more interactions when comparing two large networks.

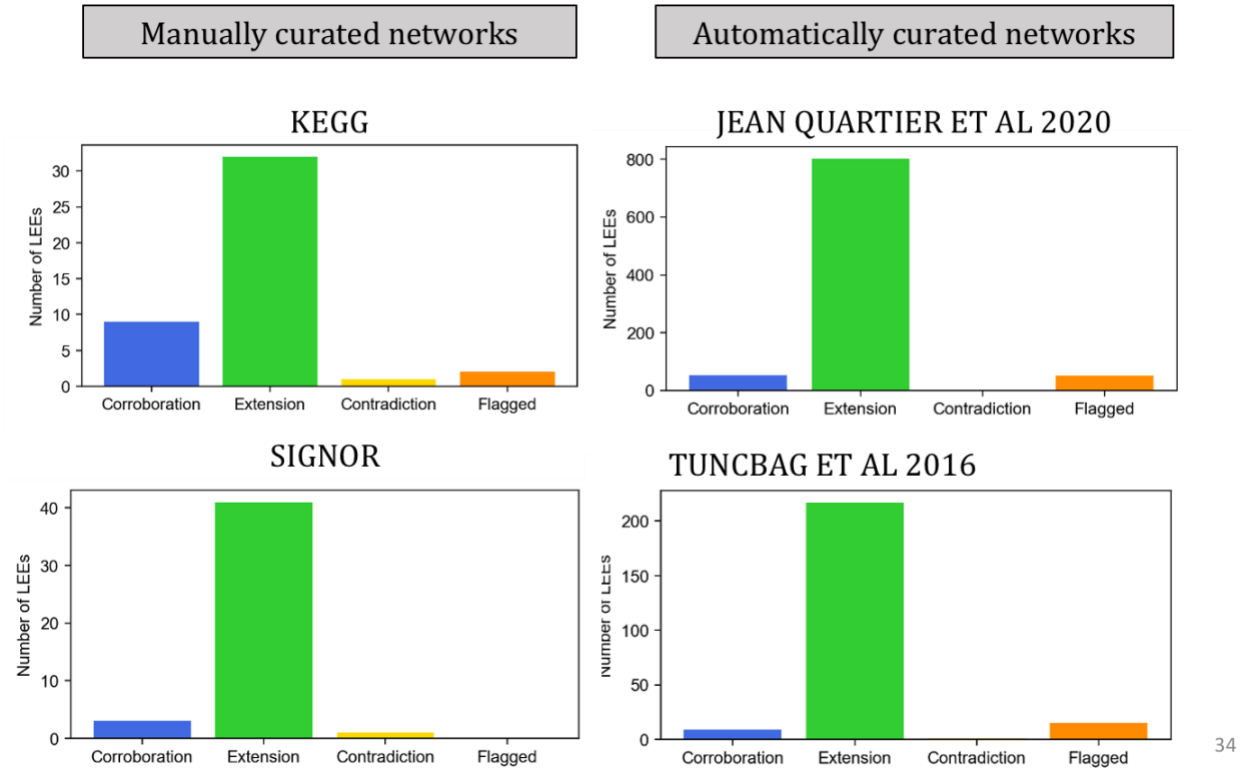
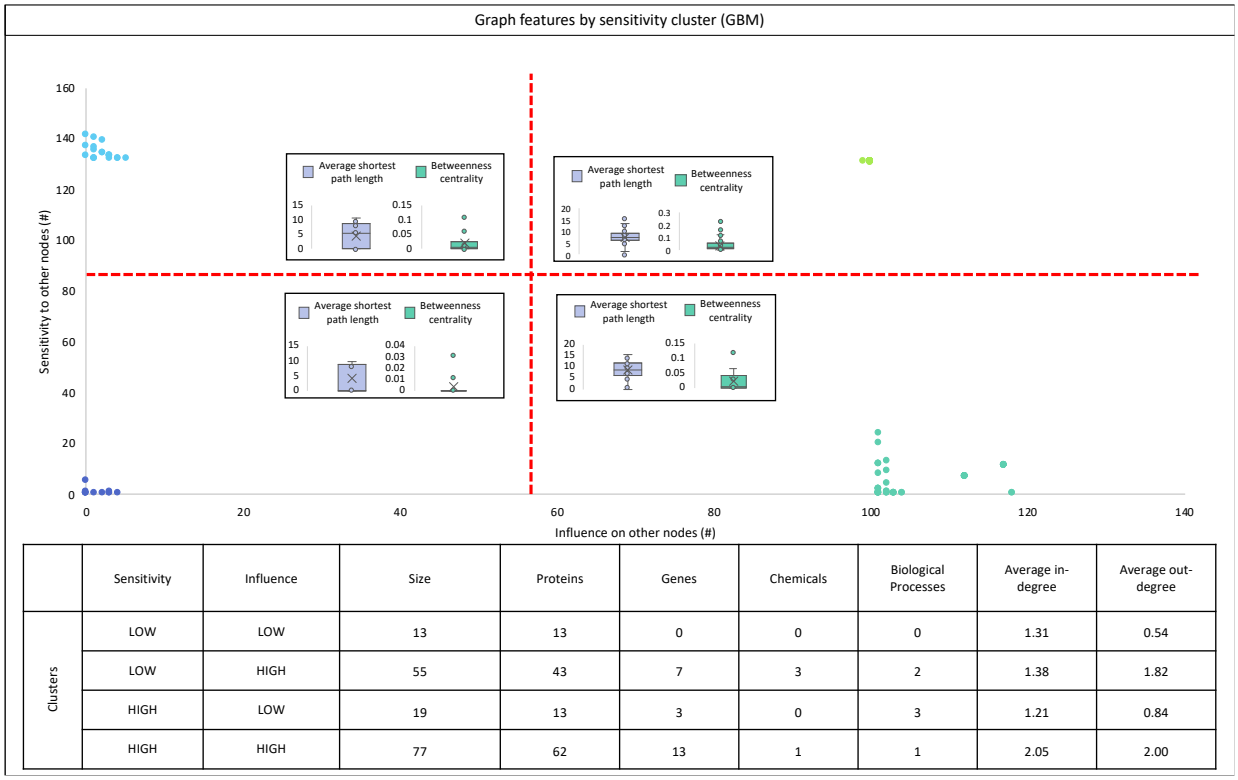


Figure 21. Comparison of the GBM model to other networks.

5.1.2 Selecting initialization methodology with PIANO



	Sensitivity	Influence	Size	Proteins	Genes	Chemicals	Biological Processes	Average in-degree	Average out-degree
Clusters	LOW	LOW	13	13	0	0	0	1.31	0.54
	LOW	HIGH	55	43	7	3	2	1.38	1.82
	HIGH	LOW	19	13	3	0	3	1.21	0.84
	HIGH	HIGH	77	62	13	1	1	2.05	2.00

Figure 22. Sensitivity analysis clusters for the GBM network. Elements that are both highly influential and highly sensitive (light green cluster) will have initial values that are more influential and downstream elements, and more susceptible to upstream elements.

To ascertain which nodes should be initialized algorithmically, the sensitivity and influence for each node was evaluated using the Pathway Importance Analyzer for Network Optimization (PIANO) [127]. Using linkage clustering, where the number of clusters was determined manually, we get well-defined clusters (Figure 22). Elements that are both highly influential and highly sensitive (light green cluster) are likely to have the greatest effect on model outcomes, and be highly dependent on the initial values of model elements. We show that sensitivity analysis in conjunction with linkage clustering can identify which model elements have initial values that are

most influential on model outcomes and that are not easily imputed from missing values. For the GBM model case study, we show the distribution of two graph features (betweenness centrality and average shortest path length) for each cluster. While average shortest path length correlates with influence, betweenness centrality is correlated with nodes that are highly sensitive and influential (top right).

6.0 Future work and discussion

In Section 3.0, I presented novel methodology for integrating literature and data for model curation. Incorporating DEGs in literature queries improves relevancy of the resulting literature corpus, and controls the size of the reading sets. For frameworks that use machine reading to extract potential model elements and interactions from literature, a filtration method can be used to guarantee that only high confidence interactions are added to the model. The filtering tool, FLUTE, enables this selection using publicly available data. FLUTE not only decreases the number of interactions that need to be tested for model improvement, it also keeps only the high-quality interactions. In conjunction with FLUTE, MINUET is capable of verifying network interactions. These methods reduce the amount of work needed for curating models both manually and automatically and ensures that any curated models rely on biologically accurate knowledge.

In Section 4.0, I show a GBM stem cell model that is consistent with multiple high confidence literature and database sources. Additionally, this model is parameterizable from data, allowing for accurate, cell line specific predictions of kinase inhibition. In Section 5.0, I show how the methodology from the DySE framework can be used to improve the GBM model.

Future work includes additional improvements to the FLUTE tool. While FLUTE is capable of returning high-quality interactions, it also discards accurate interactions depending on the threshold used. Although the optional thresholds for between-paper duplicates and for recent publications increase the recall of correct interactions, these thresholds are highly context-specific, and precision is penalized in some cases. These literature-based filters can help further reduce the time needed for manual review of interactions, but they do not fully eliminate the necessity for human intervention. To accommodate novel machine reading results that are accurate, additional

features that draw from natural language processing can be added, such as trigger words, or analyzing sentence structure. These features could help to judge the quality of the reading output that would complement databases with historical information, and they could provide further insight into the reliability of individual interactions without penalizing novel interactions.

Future directions also include refining the query formulation methodology, as well as expanding the results. The relative presence of different diseases in PubMed affects the size of the reading set, independent of the number of gene query terms. Additionally, since this method hinges on a list of affected genes or proteins with quantifiable differences from a control state, other measures of relative changes in cell function could also be used. Data on changes in post-translational modification of proteins, changes in epigenetic markers such as methylation, open chromatin, or histone modifications, or even somatic mutations could also be used, especially as such entities and events can be output by the state-of-the-art machine reading. Testing these methods on different datasets would help showcase its applicability and use for model curation.

Finally, the GBM model can be utilized in the future to guide patient treatments. By acquiring genomic data for additional cell lines, the model can be automatically parameterized and extended using the DySE framework. The model can then be used to predict effective chemical treatments *in vitro* with minimal human intervention.

Appendix A - GBM model rules

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
1	AKT_prot_act	Protein (active)	AKT1	AKT_prot	
2	AKT_gene	Gene	AKT1		
3	AKT_prot	Protein (amount)	AKT1	AKT_gene	
4	AMPK_prot_act	Protein (active)	PRKAA2	{AMPK_prot^}[MAP3K7_prot_act,Hypoxia]	
5	AMPK_gene	Gene	PRKAA2		
6	AMPK_prot	Protein (amount)	PRKAA2	AMPK_gene	
7	Apaf1_prot_act	Protein (active)	APAF1	(Apaf1_prot,CytoC_prot_act)	
8	Apaf1_gene	Gene	APAF1		
9	Apaf1_prot	Protein (amount)	APAF1	Apaf1_gene	
10	apoptosis	BiologicalProcess		Casp3_6_7_prot_act^,Casp8_prot_act	
11	ASCL1_prot_act	Protein (active)	ASCL1	ASCL1_prot	PRC1_prot_act
12	ASCL1_gene	Gene	ASCL1		
13	ASCL1_prot	Protein (amount)	ASCL1	ASCL1_gene	
14	astro_diff	BiologicalProcess		dimer_p_STAT3_prot_act	
15	ATM_prot_act	Protein (active)	ATM	(ATM_prot,FOXO_prot_act)	
16	ATM_gene	Gene	ATM		
17	ATM_prot	Protein (amount)	ATM	ATM_gene	
18	ATP	Chemical			
19	AURA_B_prot_act	Protein (active)	AURKA	AURA_B_prot	
20	AURA_B_gene	Gene	AURKA		
21	AURA_B_prot	Protein (amount)	AURKA	AURA_B_gene	
22	Bad_prot_act	Protein (active)	BAD	Bad_prot	AKT_prot_act,Bcl2_prot_act
23	Bad_gene	Gene	BAD		
24	Bad_prot	Protein (amount)	BAD	Bad_gene	
25	Bax_prot_act	Protein (active)	BAX	Bad_prot_act	Bcl2_prot_act,BclXL_prot_act
26	Bax_gene	Gene	BAX		
27	Bax_prot	Protein (amount)	BAX	Bax_gene	
28	Bcl2_gene	Gene	BCL2	dimer_p_STAT3_prot_act	
29	Bcl2_prot_act	Protein (active)	BCL2	(Bcl2_prot,ERK_prot_act)	Bad_prot_act
30	Bcl2_prot	Protein (amount)	BCL2	Bcl2_gene	
31	BclXL_gene	Gene	BCL2L1	dimer_p_STAT3_prot_act,p50_p65_prot_act	
32	BclXL_prot_act	Protein (active)	BCL2L1	(BclXL_prot,ERK_prot_act)	Bad_prot_act
33	BclXL_prot	Protein (amount)	BCL2L1	BclXL_gene	
34	beta_cat_prot_act	Protein (active)	CTNNB1	(beta_cat_prot,AKT_prot_act)	GSK3B_prot_act
35	beta_cat_gene	Gene	CTNNB1		
36	beta_cat_prot	Protein (amount)	CTNNB1	beta_cat_gene	
37	c_myc_gene	Gene	MYC	E2F_prot_act,(TCF_LEF_prot_act,beta_cat_prot_act), NOTCH_prot_act	
38	c_myc_prot_act	Protein (active)	MYC	(c_myc_prot,ERK_prot_act)	
39	c_myc_prot	Protein (amount)	MYC	c_myc_gene	
40	Casp3_6_7_prot_act	Protein (active)	CASP3	(Casp3_6_7_prot,Casp9_prot_act^)	
41	Casp3_6_7_gene	Gene	CASP3		
42	Casp3_6_7_prot	Protein (amount)	CASP3	Casp3_6_7_gene	
43	Casp8_prot_act	Protein (active)	CASP8	(Casp8_prot,FAS_prot_act^)	
44	Casp8_gene	Gene	CASP8		
45	Casp8_prot	Protein (amount)	CASP8	Casp8_gene	
46	Casp9_prot_act	Protein (active)	CASP9	(Casp9_prot,Apaf1_prot_act^)	
47	Casp9_gene	Gene	CASP9		
48	Casp9_prot	Protein (amount)	CASP9	Casp9_gene	
49	CBL_prot_act	Protein (active)	CBL	CBL_prot	
50	CBL_gene	Gene	CBL		

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
51	CBL_prot	Protein (amount)	CBL	CBL_gene	
52	CD44_prot_act	Protein (active)	CD44	CD44_prot	
53	CD44_gene	Gene	CD44		
54	CD44_prot	Protein (amount)	CD44	CD44_gene	
55	cdc25A_gene	Gene	CDC25A	(c_myc_prot_act,dimer_p_STAT3_prot_act)	
56	cdc25A_prot_act_cyto	Protein (active)	CDC25A	(cdc25A_prot,Raf_dimer_prot_act,CDK1_cycB_prot_act)	Chk1_2_prot_act
57	cdc25A_prot_act_nuc	Protein (active)	CDC25A	(cdc25A_prot_act_cyto,PLK1_prot_act)	
58	cdc25A_prot	Protein (amount)	CDC25A	cdc25A_gene	
59	CDK1_cycB_prot_act	Protein (active)	CDK1	(CDK1_cycB_prot,cdc25A_prot_act_nuc)	p21_prot_act,MYT1_prot_act
60	CDK1_cycB_gene	Gene	CDK1		
61	CDK1_cycB_prot	Protein (amount)	CDK1	CDK1_cycB_gene	
62	CDK2_prot_act	Protein (active)	CDK2	(protein_synthesis,CDK1_cycB_prot)	
63	CDK2_gene	Gene	CDK2		
64	CDK2_prot	Protein (amount)	CDK2	CDK2_gene	
65	CDK6_prot_act	Protein (active)	CDK6	(CDK6_prot,protein_synthesis)	
66	CDK6_gene	Gene	CDK6		
67	CDK6_prot	Protein (amount)	CDK6	CDK6_gene	
68	cell_cycle_progression	BiologicalProcess		CDK1_cycB_prot_act,E2F_prot_act	
69	Chk1_2_prot_act	Protein (active)	CHEK1	(ATM_prot_act,Chk1_2_prot)	
70	Chk1_2_gene	Gene	CHEK1		
71	Chk1_2_prot	Protein (amount)	CHEK1	Chk1_2_gene	
72	cycD_CDK4_6_prot_act	Protein (active)	CCND1	(cycD_prot_act,CDK6_prot_act),c_myc_prot_act, cdc25A_prot_act_nuc	(p15_prot_act,p16_prot_act), (p21_prot_act,p27_prot_act)
73	cycD_gene	Gene	CCND1	E2F_prot_act,dimer_p_STAT3_prot_act, TCF_LEF_prot_act,GLI_prot_act,NOTCH_prot_act	
74	cycD_prot_act	Protein (active)	CCND1	(cycD_prot,GSK3B_prot_act)	
75	cycD_prot	Protein (amount)	CCND1	cycD_gene	
76	cycE_CDK2_prot_act	Protein (active)	CCNE1	(cycE_prot_act,CDK2_prot_act),cdc25A_prot_act_nuc	
77	cycE_gene	Gene	CCNE1		
78	cycE_prot_act	Protein (active)	CCNE1	{cycE_prot}[CD44_prot_act]	
79	cycE_prot	Protein (amount)	CCNE1	cycE_gene	
80	CytoC_prot_act	Protein (active)	CYC1	(CytoC_prot,Bax_prot_act^)	
81	CytoC_gene	Gene	CYC1		
82	CytoC_prot	Protein (amount)	CYC1	CytoC_gene	
83	DELTA_prot_act	Protein (active)	DLL1	DELTA_prot	
84	DELTA_gene	Gene	DLL1		
85	DELTA_prot	Protein (amount)	DLL1	DELTA_gene	
86	dimer_p_STAT3_prot_act	Protein (active)	STAT3	p_STAT3_prot_act	PIAS3_prot_act
87	DKK_gene	Gene	DKK1	TCF_LEF_prot_act	ASCL1_prot_act
88	DKK_prot_act	Protein (active)	DKK1	DKK_prot	
89	DKK_prot	Protein (amount)	DKK1	DKK_gene	
90	dna_damage	BiologicalProcess		ROS	PARP1_prot_act
91	DVL_prot_act	Protein (active)	DVL1	(DVL_prot,LRP_Fz_prot_act)	
92	DVL_gene	Gene	DVL1		
93	DVL_prot	Protein (amount)	DVL1	DVL_gene	
94	E2F_prot_act	Protein (active)	E2F1	E2F_prot	Rb_prot_act
95	E2F_gene	Gene	E2F1		
96	E2F_prot	Protein (amount)	E2F1	E2F_gene	
97	EGF_prot_act	Protein (active)	EGF	EGF_prot	
98	EGF_gene	Gene	EGF		
99	EGF_prot	Protein (amount)	EGF	EGF_gene	
100	EGFR_prot	Protein (amount)	EGFR	EGFR_gene	

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
151	HH_prot_act	Protein (active)	SHH	HH_prot	
152	HH_gene	Gene	SHH		
153	HH_prot	Protein (amount)	SHH	HH_gene	
154	Hif1a_gene	Gene	HIF1A	dimer_p_STAT3_prot_act	
155	Hif1a_prot_act	Protein (active)	HIF1A	(Hif1a_prot,Hypoxia,mTORc1_prot_act)	
156	Hif1a_prot	Protein (amount)	HIF1A	Hif1a_gene	
157	Hypoxia	BiologicalProcess			
158	IkB_a_prot_act	Protein (active)	NFKBIA	(IkB_a_prot,IKK_prot_act)	
159	IkB_a_gene	Gene	NFKBIA		
160	IkB_a_prot	Protein (amount)	NFKBIA	IkB_a_gene	
161	IKK_prot_act	Protein (active)	IKBKB	(IKK_prot,MAP3K7_prot_act)	
162	IKK_gene	Gene	IKBKB		
163	IKK_prot	Protein (amount)	IKBKB	IKK_gene	
164	IL6_prot_act	Protein (active)	IL6	IL6_prot	
165	IL6_gene	Gene	IL6		
166	IL6_prot	Protein (amount)	IL6	IL6_gene	
167	INSR_prot	Protein (active)	INSR	INSR_gene	
168	INSR_prot_act	Protein (active)	INSR	(INSR_prot,Insulin)	SOCS_prot_act
169	INSR_gene	Gene	INSR		
170	Insulin	Chemical			
171	IRS_prot_act	Protein (active)	IRS1	INSR_prot_act,AMPK_prot_act	Insulin
172	IRS_gene	Gene	IRS1		
173	IRS_prot	Protein (amount)	IRS1	IRS_gene	
174	JAK2_prot_act	Protein (active)	JAK2	JAK2_prot	SOCS_prot_act,CBL_prot_act
175	JAK2_gene	Gene	JAK2		
176	JAK2_prot	Protein (amount)	JAK2	JAK2_gene	
177	KIBRA_gene	Gene	WWC1	(CD44_prot_act,NF2_prot_act)	
178	KIBRA_prot_act	Protein (active)	WWC1	KIBRA_prot	
179	KIBRA_prot	Protein (amount)	WWC1	KIBRA_gene	
180	KIF7_prot_act	Protein (active)	KIF7	KIF7_prot	
181	KIF7_gene	Gene	KIF7		
182	KIF7_prot	Protein (amount)	KIF7	KIF7_gene	
183	LATS1_2_prot_act	Protein (active)	LATS1	(LATS1_2_prot,MOB_prot_act)	
184	LATS1_2_gene	Gene	LATS1		
185	LATS1_2_prot	Protein (amount)	LATS1	LATS1_2_gene	
186	LRP_Fz_gene	Gene	LRP		
187	LRP_Fz_prot	Protein (amount)	LRP	LRP_Fz_gene	
188	LRP_Fz_prot_act	Protein (active)	LRP	(LRP_Fz_prot,Wnt_prot_act)	DKK_prot_act
189	MAP3K7_prot_act	Protein (active)	MAP3K7	(MAP3K7_prot,RIP1_prot_act)	
190	MAP3K7_gene	Gene	MAP3K7		
191	MAP3K7_prot	Protein (amount)	MAP3K7	MAP3K7_gene	
192	MDM2_gene	Gene	MDM2	p53_prot_act,MUT_MDM2	
193	MDM2_prot_act	Protein (active)	MDM2	(MDM2_prot,(AKT_prot_act,!p16_prot_act))	PTEN_prot_act
194	MDM2_prot	Protein (amount)	MDM2	MDM2_gene	
195	MDMX_prot_act	Protein (active)	MDM4	(MDMX_prot,dna_damage)	MDM2_prot_act,ATM_prot_act
196	MDMX_gene	Gene	MDM4		
197	MDMX_prot	Protein (amount)	MDM4	MDMX_gene	
198	MEK_prot_act	Protein (active)	MEK	(MEK_prot,Raf_dimer_prot_act), (MEK_prot,PKC_prot_act)	
199	MEK_gene	Gene	MEK		
200	MEK_prot	Protein (amount)	MEK	MEK_gene	

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
201	MER_prot_act	Protein (active)	MER	MER_prot	
202	MER_gene	Gene	MER		
203	MER_prot	Protein (amount)	MER	MER_gene	
204	Met_prot_act	Protein (active)	Met	Met_prot	
205	Met_gene	Gene	Met		
206	Met_prot	Protein (amount)	Met	Met_gene	
207	MOB_prot_act	Protein (active)	MOB	(MOB_prot,MST1_2_prot_act)	
208	MOB_gene	Gene	MOB		
209	MOB_prot	Protein (amount)	MOB	MOB_gene	
210	MST1_2_prot_act	Protein (active)	MST1	(MST1_2_prot,SAV1_prot_act)	RASSF6_prot_act,PP2A_prot_act
211	MST1_2_gene	Gene	MST1		
212	MST1_2_prot	Protein (amount)	MST1	MST1_2_gene	
213	mTORc1_prot_act	Protein (active)	MTOR	(mTORc1_prot,PIP3,PRAS_prot_act,!AMPK_prot_act), (mTORc1_prot,Rheb_prot_act),(mTORc1_prot,RPS6KA1_prot_act)	Hif1a_prot_act
214	mTORc1_gene	Gene	MTOR		
215	mTORc1_prot	Protein (amount)	MTOR	mTORc1_gene	
216	mTORc2_prot_act	Protein (active)	MTOR	mTORc2_prot	
217	mTORc2_gene	Gene	MTOR		
218	mTORc2_prot	Protein (amount)	MTOR	mTORc2_gene	
219	MUT_CDKN2A	Mutation			
220	MUT_MDM2	Mutation			
221	MUT_PDGFRA	Mutation			
222	MUT_PTEN	Mutation			
223	MYT1_prot_act	Protein (active)	Myt1	MYT1_prot	PLK1_prot_act,RPS6KA1_prot_act
224	MYT1_gene	Gene	Myt1		
225	MYT1_prot	Protein (amount)	Myt1	MYT1_gene	
226	Nestin_prot_act	Protein (active)	NES	Nestin_prot	
227	Nestin_prot	Protein (amount)	NES	Nestin_gene	
228	Nestin_gene	Gene	NES	NOTCH_prot_act	
229	neuronal_diff	BiologicalProcess		ASCL1_prot_act	
230	NF1_prot_act	Protein (active)	NF1	NF1_prot^	
231	NF1_gene	Gene	NF1		
232	NF1_prot	Protein (amount)	NF1	NF1_gene^	
233	NF2_prot_act	Protein (active)	NF2	NF2_prot	AKT_prot_act
234	NF2_gene	Gene	NF2		
235	NF2_prot	Protein (amount)	NF2	NF2_gene	
236	NLK_prot_act	Protein (active)	NLK	(NLK_prot,MAP3K7_prot_act)	
237	NLK_gene	Gene	NLK		
238	NLK_prot	Protein (amount)	NLK	NLK_gene	
239	NOTCH_prot_act	Protein (active)	NOTCH1	(NOTCH_prot,GSK3B_prot_act),(NOTCH_prot, {DELTA_prot_act}[FRINGE_prot_act])	DVL_prot_act,NUMB_prot_act, SERRATE_prot_act
240	NOTCH_gene	Gene	NOTCH1		
241	NOTCH_prot	Protein (amount)	NOTCH1	NOTCH_gene	
242	NSB1_prot_act	Protein (active)	NSB1	(NSB1_prot,ATM_prot_act)	
243	NSB1_gene	Gene	NSB1		
244	NSB1_prot	Protein (amount)	NSB1	NSB1_gene	
245	NUMB_prot_act	Protein (active)	NUMB	NUMB_prot	
246	NUMB_gene	Gene	NUMB		
247	NUMB_prot	Protein (amount)	NUMB	NUMB_gene	
248	p_STAT3_prot_act	Protein (active)	STAT3	(STAT3_prot,JAK2_prot_act,SRC_prot_act)	PIAS3_prot_act
249	p15_gene	Gene	CDKN2B	FOXO_prot_act,SMAD2_3_4_prot_act	(SMAD2_3_4_prot_act, c_myc_prot_act)
250	p15_prot_act	Protein (active)	CDKN2B	p15_prot	

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
251	p15_prot	Protein (amount)	CDKN2B	p15_gene	
252	p16_prot_act	Protein (active)	CDKN2A	(p16_prot,RAS_prot_act),(p16_prot,!c_myc_prot_act)	PRC1_prot_act,2*MUT_CDKN2A
253	p16_gene	Gene	CDKN2A		
254	p16_prot	Protein (amount)	CDKN2A	p16_gene	
255	p21_gene	Gene	CDKN1A	FOXO_prot_act,E2F_prot_act,p53_prot_act	c_myc_prot_act
256	p21_prot_act	Protein (active)	CDKN1A	(p21_prot,Raf_dimer_prot_act,SMAD2_3_4_prot_act, AKT_prot_act)	SKP2_prot_act
257	p21_prot	Protein (amount)	CDKN1A	p21_gene	
258	p27_gene	Gene	CDKN1B	FOXO_prot_act	
259	p27_prot_act	Protein (active)	CDKN1B	p27_prot	SKP2_prot_act,AKT_prot_act, cycE_CDK2_prot_act
260	p27_prot	Protein (amount)	CDKN1B	p27_gene	
261	p50_p65_prot_act	Protein (active)	RELA	(p50_p65_prot,IkB_a_prot_act)	
262	p50_p65_gene	Gene	RELA		
263	p50_p65_prot	Protein (amount)	RELA	p50_p65_gene	
264	p53_gene	Gene	TP53		dimer_p_STAT3_prot_act
265	p53_prot_act	Protein (active)	TP53	(p53_prot,ATM_prot_act),(p53_prot,ATM_prot_act)	(MDM2_prot_act,MDMX_prot_act), Chk1_2_prot_act
266	p53_prot	Protein (amount)	TP53	p53_gene	
267	PARP1_prot_act	Protein (active)	PARP1	(PARP1_prot_act,ROS)	
268	PARP1_gene	Gene	PARP1		
269	PARP1_prot	Protein (amount)	PARP1	PARP1_gene	
270	PDGF_prot_act	Protein (active)	PDGFB	PDGF_prot	
271	PDGF_gene	Gene	PDGFB		
272	PDGF_prot	Protein (amount)	PDGFB	PDGF_gene	
273	PDGFRA_gene	Gene	PDGFRA	2*MUT_PDGFRA	
274	PDGFRA_prot_act	Protein (active)	PDGFRA	(PDGF_prot_act,PDGFRA_prot)	
275	PDGFRA_prot	Protein (amount)	PDGFRA	PDGFRA_gene	
276	PDK_prot_act	Protein (active)	PDK	(PDK_prot,PIP3)	
277	PDK_gene	Gene	PDK		
278	PDK_prot	Protein (amount)	PDK	PDK_gene	
279	PI3K_prot_act	Protein (active)	PI3K	{PI3K_prot}[VEGFR_prot_act,PDGFRA_prot_act, RAS_prot_act]	PTEN_prot_act
280	PI3K_gene	Gene	PI3K		
281	PI3K_prot	Protein (amount)	PI3K	PI3K_gene	
282	PIAS3_prot_act	Protein (active)	PIAS3	(PIAS3_prot,p_STAT3_prot_act)	
283	PIAS3_gene	Gene	PIAS3		
284	PIAS3_prot	Protein (amount)	PIAS3	PIAS3_gene	
285	PIP3	Chemical		PI3K_prot_act	PTEN_prot_act
286	PKC_prot_act	Protein (active)	PRKCA	(PKC_prot,PDK_prot_act,PLC_prot_act)	
287	PKC_gene	Gene	PRKCA		
288	PKC_prot	Protein (amount)	PRKCA	PKC_gene	
289	PLC_prot_act	Protein (active)	PLCG1	(PLC_prot,EGFR_prot_act),(PLC_prot,VEGFR_prot_act), (PLC_prot,PDGFRA_prot_act)	
290	PLC_gene	Gene	PLCG1		
291	PLC_prot	Protein (amount)	PLCG1	PLC_gene	
292	PLK1_prot_act	Protein (active)	PLK1	(PLK1_prot,AURA_B_prot_act,CDK1_cycB_prot_act)	
293	PLK1_gene	Gene	PLK1		
294	PLK1_prot	Protein (amount)	PLK1	PLK1_gene	
295	PP2A_prot_act	Protein (active)	PPP2CA	PP2A_prot	Chk1_2_prot_act
296	PP2A_gene	Gene	PPP2CA		
297	PP2A_prot	Protein (amount)	PPP2CA	PP2A_gene	
298	PRAS_prot_act	Protein (active)	AKT1S1	PRAS_prot	AKT_prot_act
299	PRAS_gene	Gene	AKT1S1		
300	PRAS_prot	Protein (amount)	AKT1S1	PRAS_gene	
301	PRC1_gene	Gene	PRC1	c_myc_prot_act	
302	PRC1_prot_act	Protein (active)	PRC1	PRC1_prot	
303	PRC1_prot	Protein (amount)	PRC1	PRC1_gene	
304	proliferation	BiologicalProcess		cell_cycle_progression	apoptosis
305	protein_synthesis	BiologicalProcess		S6_prot_act	

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
306	PTEN_gene	Gene	PTEN	FOXO_prot_act	
307	PTEN_prot_act	Protein (active)	PTEN	PTEN_prot	SRC_prot_act
308	PTEN_prot	Protein (amount)	PTEN	(PTEN_gene,!MUT_PTEN)	
309	Raf_dimer_prot_act	Protein (active)	RAF1	{(Raf_dimer_prot,RAS_prot_act)} [PKC_prot_act,!AKT_prot_act]	
310	Raf_dimer_gene	Gene	RAF1		
311	Raf_dimer_prot	Protein (amount)	RAF1	Raf_dimer_gene	
312	RAS_gene	Gene	HRAS		
313	RAS_prot_act	Protein (active)	HRAS	(RAS_prot,SOS_prot_act,EGFR_prot_act)	NF1_prot_act
314	RAS_prot	Protein (amount)	HRAS	RAS_gene	
315	RASSF6_prot_act	Protein (active)	RASSF6	RASSF6_prot	
316	RASSF6_gene	Gene	RASSF6		
317	RASSF6_prot	Protein (amount)	RASSF6	RASSF6_gene	
318	Rb_prot_act	Protein (active)	RB1	Rb_prot	(cycE_CDK2_prot_act, cycD_CDK4_6_prot_act)
319	Rb_gene	Gene	RB1		
320	Rb_prot	Protein (amount)	RB1	Rb_gene	
321	REDD1_prot_act	Protein (active)	DDIT4	(REDD1_prot,Hypoxia)	GSK3B_prot_act
322	REDD1_gene	Gene	DDIT4		
323	REDD1_prot	Protein (amount)	DDIT4	REDD1_gene	
324	Rheb_prot_act	Protein (active)	RHEB	(Rheb_prot,ATP)	TSC1_2_prot_act
325	Rheb_gene	Gene	RHEB		
326	Rheb_prot	Protein (amount)	RHEB	Rheb_gene	
327	RIP1_prot_act	Protein (active)	RIPK1	(RIP1_prot,TNFR_prot_act)	
328	RIP1_gene	Gene	RIPK1		
329	RIP1_prot	Protein (amount)	RIPK1	RIP1_gene	
330	ROS	Chemical			Hypoxia
331	RPS6KA1_prot_act	Protein (active)	RPS6KA1	(RPS6KA1_prot,ERK_prot_act)	
332	RPS6KA1_gene	Gene	RPS6KA1		
333	RPS6KA1_prot	Protein (amount)	RPS6KA1	RPS6KA1_gene	
334	RPS6KB1_2_prot_act	Protein (active)	RPS6KB1	(RPS6KB1_2_prot,mTORc1_prot_act)	
335	RPS6KB1_2_gene	Gene	RPS6KB1		
336	RPS6KB1_2_prot	Protein (amount)	RPS6KB1	RPS6KB1_2_gene	
337	S6_prot_act	Protein (active)	RPS6	(S6_prot,RPS6KA1_prot_act,RPS6KB1_2_prot_act)	
338	S6_gene	Gene	RPS6		
339	S6_prot	Protein (amount)	RPS6	S6_gene	
340	SAV1_prot_act	Protein (active)	SAV1	(SAV1_prot,F_ACTIN_prot_act)	
341	SAV1_gene	Gene	SAV1		
342	SAV1_prot	Protein (amount)	SAV1	SAV1_gene	
343	SERRATE_prot_act	Protein (active)	SRRT	SERRATE_prot	
344	SERRATE_gene	Gene	SRRT		
345	SERRATE_prot	Protein (amount)	SRRT	SERRATE_gene	
346	SHC_prot_act	Protein (active)	SHC1	(SHC_prot,EGFR_prot_act),(SHC_prot,INSR_prot_act)	PTEN_prot_act
347	SHC_gene	Gene	SHC1		
348	SHC_prot	Protein (amount)	SHC1	SHC_gene	
349	SHP2_prot_act	Protein (active)	PTPN11	SHP2_prot	
350	SHP2_gene	Gene	PTPN11		
351	SHP2_prot	Protein (amount)	PTPN11	SHP2_gene	
352	SKP2_prot_act	Protein (active)	SKP2	(SKP2_prot,c_myc_prot_act),(SKP2_prot,ERK_prot_act)	
353	SKP2_gene	Gene	SKP2		
354	SKP2_prot	Protein (amount)	SKP2	SKP2_gene	
355	SMAC_prot_act	Protein (active)	DIABLO	(SMAC_prot,CytoC_prot_act)	XIAP_prot_act
356	SMAC_gene	Gene	DIABLO		
357	SMAC_prot	Protein (amount)	DIABLO	SMAC_gene	
358	SMAD2_3_prot_act	Protein (active)	SMAD2	(SMAD2_3_prot,TGFBR_prot_act)	
359	SMAD2_3_gene	Gene	SMAD2		
360	SMAD2_3_prot	Protein (amount)	SMAD2	SMAD2_3_gene	

#	Element IDs	Element Type	HGNC Symbol	Positive Regulators	Negative Regulators
361	SMAD2_3_4_prot_act	Protein (active)	SMAD2	(SMAD2_3_prot_act,SMAD4_prot)	ERK_prot_act
362	SMAD2_3_4_gene	Gene	SMAD2		
363	SMAD2_3_4_prot	Protein (amount)	SMAD2	SMAD4_gene	
364	SOCS_gene	Gene	SOCS	dimer_p_STAT3_prot_act	
365	SOCS_prot_act	Protein (active)	SOCS	SOCS_prot	
366	SOCS_prot	Protein (amount)	SOCS	SOCS_gene	
367	SOS_prot_act	Protein (active)	SOS	(SOS_prot,Grb2_prot_act),(SOS_prot,PDGFRA_prot_act)	
368	SOS_gene	Gene	SOS		
369	SOS_prot	Protein (amount)	SOS	SOS_gene	
370	SPRY2_prot	Protein (amount)	SPRY2	SPRY2_gene	
371	SPRY2_prot_act	Protein (active)	SPRY2	SPRY2_prot	
372	SPRY2_gene	Gene	SPRY2	ERK_prot_act	
373	SRC_prot_act	Protein (active)	SRC	(SRC_prot,EGFR_prot_act)	
374	SRC_gene	Gene	SRC		
375	SRC_prot	Protein (amount)	SRC	SRC_gene	
376	STAT3_prot_act	Protein (active)	STAT3	(STAT3_prot,EGFR_prot_act),(STAT3_prot, Met_prot_act), (STAT3_prot,SRC_prot_act)	
377	STAT3_gene	Gene	STAT3		
378	STAT3_prot	Protein (amount)	STAT3	STAT3_gene	
379	Stemness	BiologicalProcess		NOTCH_prot_act,dimer_p_STAT3_prot_act, GLI_prot_act,TCF_LEF_prot_act	
380	SUFU_prot_act	Protein (active)	SUFU	SUFU_prot	
381	SUFU_gene	Gene	SUFU		
382	SUFU_prot	Protein (amount)	SUFU	SUFU_gene	
383	TACE_prot_act	Protein (active)	ADAM17	TACE_prot	
384	TACE_gene	Gene	ADAM17		
385	TACE_prot	Protein (amount)	ADAM17	TACE_gene	
386	TCF_LEF_prot_act	Protein (active)	TCF7	(TCF_LEF_prot,SMAD2_3_4_prot_act),(TCF_LEF_prot, beta_cat_prot_act)	NLK_prot_act
387	TCF_LEF_gene	Gene	TCF7		
388	TCF_LEF_prot	Protein (amount)	TCF7	TCF_LEF_gene	
389	TGFB_prot_act	Protein (active)	TGFB1	TGFB_prot	
390	TGFB_gene	Gene	TGFB1		
391	TGFB_prot	Protein (amount)	TGFB1	TGFB_gene	
392	TGFBR_prot	Protein (amount)	TGFBR1	TGFBR_gene	
393	TGFBR_prot_act	Protein (active)	TGFBR1	(TGFBR_prot, TGFB_prot_act)	
394	TGFBR_gene	Gene	TGFBR1		
395	TNF_gene	Gene	TNF	p50_p65_prot_act	
396	TNF_prot_act	Protein (active)	TNF	TNF_prot	
397	TNF_prot	Protein (amount)	TNF	TNF_gene	
398	TNFR_prot	Protein (amount)	TNFRSF1A	TNFR_gene	
399	TNFR_prot_act	Protein (active)	TNFRSF1A	(TNFR_prot,TNF_prot_act)	
400	TNFR_gene	Gene	TNFRSF1A		
401	TSC1_2_prot_act	Protein (active)	TSC1	(TSC1_2_prot,REDD1_prot_act),(TSC1_2_prot,AMPK_prot_act), (TSC1_2_prot,GSK3B_prot_act,AMPK_prot_act)	AKT_prot_act,ERK_prot_act, (ERK_prot_act,Insulin)
402	TSC1_2_gene	Gene	TSC1		
403	TSC1_2_prot	Protein (amount)	TSC1	TSC1_2_gene	
404	VEGF_gene	Gene	VEGFA	Hif1a_prot_act	
405	VEGF_prot_act	Protein (active)	VEGFA	VEGF_prot	
406	VEGF_prot	Protein (amount)	VEGFA	VEGF_gene	
407	VEGFR_prot	Protein (amount)	KDR	VEGFR_gene	
408	VEGFR_prot_act	Protein (active)	KDR	(VEGFR_prot, VEGF_prot_act)	
409	VEGFR_gene	Gene	KDR		
410	Wnt_prot_act	Protein (active)	WNT1	Wnt_prot	
411	Wnt_gene	Gene	WNT1		
412	Wnt_prot	Protein (amount)	WNT1	Wnt_gene	
413	XIAP_prot_act	Protein (active)	XIAP	(XIAP_prot,MDM2_prot_act)	
414	XIAP_gene	Gene	XIAP		
415	XIAP_prot	Protein (amount)	XIAP	XIAP_gene	

Appendix B – Updated motifs

Motif 1: Complex formation

A_B = {A,B}[Regulator1,Regulator2, …]

Motif 2: Simplified pathway

Pathway = A

Motif 3: Chemical reaction, enzymatic reaction, and PTM

protein_modified = (protein_unmodified, small_molecule)

Example:

RAS_GTP = (RAS,GTP)

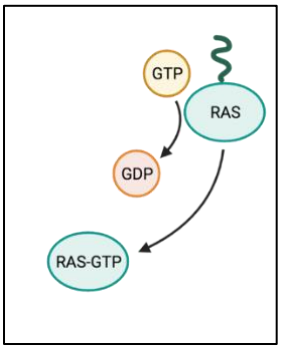


Figure 23. Example of an enzymatic reaction.

Motif 4: Markers

biological process = positive_marker_1,…,positive_marker_N, !negative_marker_1, … ,!negative marker_M

Example:
Inflammation = TNFa,IL6,IL12,IL1B
Phagocytosis = !TNFa,!IL6,!IL12,!IL1B

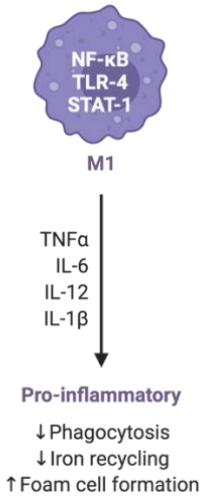


Figure 24. Example of markers regulating a biological process.

Motif 5: Indirect interaction

protein_downstream = protein_upstream

Example:

A = !B
B= A
C= A,!E
D= A
E= !A

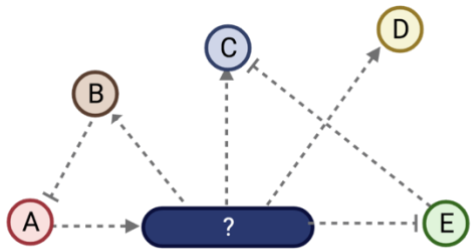


Figure 25. An example network with missinf information, and the subsequent infered indirect interactions.

Bibliography

[1] H. Kitano, "Computational systems biology," *Nature*, vol. 420, no. 6912, pp. 206-210, 2002/11/01 2002, doi: 10.1038/nature01254.

[2] I. Albert, J. Thakar, S. Li, R. Zhang, and R. Albert, "Boolean network simulations for life scientists," *Source Code for Biology and Medicine*, vol. 3, no. 1, p. 16, 2008 2008, doi: 10.1186/1751-0473-3-16.

[3] B. N. Kholodenko, "Cell-signalling dynamics in time and space," *Nature Reviews Molecular Cell Biology*, vol. 7, no. 3, pp. 165-176, 2006/03/01 2006, doi: 10.1038/nrm1838.

[4] U. Ben-David, R. Beroukhim, and T. R. Golub, "Genomic evolution of cancer models: perils and opportunities," (in eng), *Nat Rev Cancer*, vol. 19, no. 2, pp. 97-109, Feb 2019, doi: 10.1038/s41568-018-0095-3.

[5] M. E. Davis, "Glioblastoma: Overview of Disease and Treatment," *Clinical journal of oncology nursing*, vol. 20, no. 5 Suppl, pp. S2-S8, 2016, doi: 10.1188/16.CJON.S1.2-8.

[6] A. Soeda, A. Hara, T. Kunisada, S.-i. Yoshimura, T. Iwama, and D. M. Park, "The Evidence of Glioblastoma Heterogeneity," *Scientific reports*, Article vol. 5, p. 7979, 2015, doi: https://doi.org/10.1038/srep07979.

[7] J. D. Lathia, S. C. Mack, E. E. Mulkearns-Hubert, C. L. Valentim, and J. N. Rich, "Cancer stem cells in glioblastoma," (in eng), *Genes & development*, vol. 29, no. 12, pp. 1203-17, Jun 15 2015, doi: 10.1101/gad.261982.115.

[8] P. C. De Witt Hamer, "Small molecule kinase inhibitors in glioblastoma: a systematic review of clinical studies," (in eng), *Neuro-oncology*, vol. 12, no. 3, pp. 304-316, 2010, doi: 10.1093/neuonc/nop068.

[9] K. Sayed, C. A. Telmer, A. A. Butchy, and N. Miskov-Zivanov, "Recipes for Translating Big Data Machine Reading to Executable Cellular Signaling Models," Cham, 2018: Springer International Publishing, in Machine Learning, Optimization, and Big Data, pp. 1-15.

[10] J. Bjorne, F. Ginter, S. Pyysalo, J. Tsujii, and T. Salakoski, "Complex event extraction at PubMed scale," *Bioinformatics*, vol. 26, no. 12, pp. i382-90, Jun 15 2010, doi: 10.1093/bioinformatics/btq180.

[11] D. Szklarczyk *et al.*, "The STRING database in 2017: quality-controlled protein-protein association networks, made broadly accessible," *Nucleic Acids Res*, vol. 45, no. D1, pp. D362-D368, 2017, doi: 10.1093/nar/gkw937.

[12] M. Ashburner *et al.*, "Gene Ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, p. 25, 05/01/online 2000, doi: 10.1038/75556.

[13] M. Kuhn, C. von Mering, M. Campillos, L. J. Jensen, and P. Bork, "STITCH: interaction networks of chemicals and proteins," (in eng), *Nucleic Acids Res*, vol. 36, no. Database issue, pp. D684-8, Jan 2008, doi: 10.1093/nar/gkm795.

[14] J. G.T. Zañudo, S. N. Steinway, and R. Albert, "Discrete dynamic network modeling of oncogenic signaling: Mechanistic insights for personalized treatment of cancer," *Current Opinion in Systems Biology*, vol. 9, pp. 1-10, 2018/06/01/ 2018, doi: https://doi.org/10.1016/j.coisb.2018.02.002.

[15] R. Bazzoni and A. Bentivegna, "Role of Notch Signaling Pathway in Glioblastoma Pathogenesis," (in eng), *Cancers (Basel)*, vol. 11, no. 3, p. 292, 2019, doi: 10.3390/cancers11030292.

[16] T. S. Christensen, A. P. Oliveira, and J. Nielsen, "Reconstruction and logical modeling of glucose repression signaling pathways in Saccharomyces cerevisiae," *BMC Systems Biology*, vol. 3, no. 1, p. 7, 2009/01/14 2009, doi: 10.1186/1752-0509-3-7.

[17] R. E. Baker, J. M. Peña, J. Jayamohan, and A. Jérusalem, "Mechanistic models versus machine learning, a fight worth fighting for the biological community?," (in eng), *Biol Lett*, vol. 14, no. 5, May 2018, doi: 10.1098/rsbl.2017.0660.

[18] P. Nurse, "Biology must generate ideas as well as data," *Nature*, vol. 597, no. 7876, pp. 305-305, 2021, doi: 10.1038/d41586-021-02480-z.

[19] J. Saez-Rodriguez and N. Blüthgen, "Personalized signaling models for personalized treatments," *Molecular Systems Biology*, vol. 16, no. 1, p. e9042, 2020, doi: https://doi.org/10.15252/msb.20199042.

[20] M. D. Wilkinson *et al.*, "The FAIR Guiding Principles for scientific data management and stewardship," *Scientific Data*, vol. 3, no. 1, p. 160018, 2016/03/15 2016, doi: 10.1038/sdata.2016.18.

[21] E. U. Azeloglu and R. Iyengar, "Signaling networks: information flow, computation, and decision making," (in eng), *Cold Spring Harb Perspect Biol*, vol. 7, no. 4, pp. a005934-a005934, 2015, doi: 10.1101/cshperspect.a005934.

[22] O. Brandman and T. Meyer, "Feedback loops shape cellular signals in space and time," (in eng), *Science*, vol. 322, no. 5900, pp. 390-395, 2008, doi: 10.1126/science.1160617.

[23] N. J. Eungdamrong and R. Iyengar, "Modeling cell signaling networks," (in eng), *Biol Cell*, vol. 96, no. 5, pp. 355-362, 2004, doi: 10.1016/j.biolcel.2004.03.004.

[24] M. Kanehisa, Y. Sato, M. Kawashima, M. Furumichi, and M. Tanabe, "KEGG as a reference resource for gene and protein annotation," *Nucleic Acids Research*, vol. 44, no. D1, pp. D457-D462, 2016, doi: 10.1093/nar/gkv1070.

[25] J. Van Meenen *et al.*, "Making Biomedical Sciences publications more accessible for machines," *Medicine, Health Care and Philosophy*, vol. 25, no. 2, pp. 179-190, 2022/06/01 2022, doi: 10.1007/s11019-022-10069-0.

[26] N. Fiorini, D. J. Lipman, and Z. Lu, "Towards PubMed 2.0," (in eng), *Elife*, vol. 6, p. e28801, 2017, doi: 10.7554/eLife.28801.

[27] A. A. Ahmed, A. D. Mohamed, M. Gener, W. Li, and E. Taboada, "YAP and the Hippo pathway in pediatric cancer," (in eng), *Mol Cell Oncol*, vol. 4, no. 3, p. e1295127, 2017, doi: 10.1080/23723556.2017.1295127.

[28] J. Bohère *et al.*, "Shavenbaby and Yorkie mediate Hippo signaling to protect adult stem cells from apoptosis," (in eng), *Nat Commun*, vol. 9, no. 1, p. 5123, Nov 30 2018, doi: 10.1038/s41467-018-07569-0.

[29] M. Maugeri-Saccà and R. De Maria, "The Hippo pathway in normal development and cancer," (in eng), *Pharmacol Ther*, vol. 186, pp. 60-72, Jun 2018, doi: 10.1016/j.pharmthera.2017.12.011.

[30] M. Maugeri-Saccà and R. De Maria, "Hippo pathway and breast cancer stem cells," (in eng), *Crit Rev Oncol Hematol*, vol. 99, pp. 115-22, Mar 2016, doi: 10.1016/j.critrevonc.2015.12.004.

[31] J. S. Mo, H. W. Park, and K. L. Guan, "The Hippo signaling pathway in stem cell biology and cancer," (in eng), *EMBO Rep*, vol. 15, no. 6, pp. 642-56, Jun 2014, doi: 10.15252/embr.201438638.

[32] J. H. Park, J. E. Shin, and H. W. Park, "The Role of Hippo Pathway in Cancer Stem Cell Biology," (in eng), *Mol Cells*, vol. 41, no. 2, pp. 83-92, Feb 28 2018, doi: 10.14348/molcells.2018.2242.

[33] U. Basu-Roy *et al.*, "Sox2 antagonizes the Hippo pathway to maintain stemness in cancer cells," (in eng), *Nat Commun*, vol. 6, p. 6411, Apr 2 2015, doi: 10.1038/ncomms7411.

[34] A. Sesagiri Raamkumar, S. Foo, and N. Pang, "Using author-specified keywords in building an initial reading list of research papers in scientific paper retrieval and recommender systems," *Information Processing & Management*, vol. 53, no. 3, pp. 577-594, 2017/05/01/ 2017, doi: https://doi.org/10.1016/j.ipm.2016.12.006.

[35] A. A. A. Abdulla, H. Lin, B. Xu, and S. K. Banbhrani, "Improving biomedical information retrieval by linear combinations of different query expansion techniques," *BMC bioinformatics*, vol. 17, no. 7, p. 238, 2016/07/25 2016, doi: 10.1186/s12859-016-1092-8.

[36] I. Wesley-Smith and J. D. West, "Babel: A Platform for Facilitating Research in Scholarly Article Discovery," presented at the Proceedings of the 25th International Conference Companion on World Wide Web, Montréal, Québec, Canada, 2016. [Online]. Available: https://doi.org/10.1145/2872518.2890517.

[37] S. Saleh and P. Pecina, "Term Selection for Query Expansion in Medical Cross-Lingual Information Retrieval," in *Advances in Information Retrieval*, Cham, L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, and D. Hiemstra, Eds., 2019// 2019: Springer International Publishing, pp. 507-522.

[38] O. Etzioni, M. Banko, and M. J. Cafarella, "Machine Reading," in *AAAI*, 2006, vol. 6, pp. 1517-1519.

[39] D. Chicco, "Ten quick tips for machine learning in computational biology," *BioData mining*, vol. 10, no. 1, p. 35, 2017.

[40] J. Fluck and M. Hofmann-Apitius, "Text mining for systems biology," *Drug discovery today*, vol. 19, no. 2, pp. 140-144, 2014.

[41] L. A. Hurst *et al.*, "TNFα drives pulmonary arterial hypertension by suppressing the BMP type-II receptor and altering NOTCH signalling," (in eng), *Nature communications*, vol. 8, pp. 14079-14079, 2017, doi: 10.1038/ncomms14079.

[42] J. Björne and T. Salakoski, "Generalizing biomedical event extraction," presented at the Proceedings of the BioNLP Shared Task 2011 Workshop, Portland, Oregon, 2011.

[43] K. B. Cohen, D. Demner-Fushman, S. Ananiadou, and J.-i. Tsujii, "Proceedings of BioNLP 15," *Proceedings of BioNLP 15*, 2015.

[44] L. Li, J. Wan, J. Zheng, and J. Wang, "Biomedical event extraction based on GRU integrating attention mechanism," *BMC bioinformatics*, journal article vol. 19, no. 9, p. 285, August 13 2018, doi: 10.1186/s12859-018-2275-2.

[45] B. M. Gyori, J. A. Bachman, K. Subramanian, J. L. Muhlich, L. Galescu, and P. K. Sorger, "From word models to executable models of signaling networks using automated assembly," *Molecular systems biology*, vol. 13, no. 11, pp. 954-954, 2017, doi: 10.15252/msb.20177651.

[46] J. Bjorne and T. Salakoski, "TEES 2.2: Biomedical Event Extraction for Diverse Corpora," (in eng), *BMC bioinformatics*, vol. 16 Suppl 16, p. S4, 2015, doi: 10.1186/1471-2105-16-s16-s4.

[47] C. T. Hoyt *et al.*, "Re-curation and rational enrichment of knowledge graphs in Biological Expression Language," *Database,* vol. 2019, 2019, doi: 10.1093/database/baz068.
[48] B. M. Gyori, C. T. Hoyt, and A. Steppi, "Gilda: biomedical entity text normalization with machine-learned disambiguation as a service," *bioRxiv,* p. 2021.09.10.459803, 2021, doi: 10.1101/2021.09.10.459803.
[49] S. Pundir, M. Magrane, M. J. Martin, C. O'Donovan, and U. Consortium, "Searching and Navigating UniProt Databases," *Curr Protoc Bioinformatics,* vol. 50, pp. 1.27.1-1.27.10, 2015, doi: 10.1002/0471250953.bi0127s50.
[50] C. UniProt, "UniProt: a worldwide hub of protein knowledge," (in eng), *Nucleic acids research,* vol. 47, no. D1, pp. D506-D515, 2019, doi: 10.1093/nar/gky1049.
[51] The UniProt Consortium, "UniProt: the universal protein knowledgebase," *Nucleic Acids Research,* vol. 45, no. D1, pp. D158-D169, 2017, doi: 10.1093/nar/gkw1099.
[52] K. Degtyarenko *et al.*, "ChEBI: a database and ontology for chemical entities of biological interest," (in eng), *Nucleic acids research,* vol. 36, no. Database issue, pp. D344-D350, 2008, doi: 10.1093/nar/gkm791.
[53] The Gene Ontology Consortium, "Expansion of the Gene Ontology knowledgebase and resources," *Nucleic Acids Research,* vol. 45, no. D1, pp. D331-D338, 2017, doi: 10.1093/nar/gkw1108.
[54] The Gene Ontology Consortium, "The Gene Ontology Resource: 20 years and still GOing strong," *Nucleic Acids Research,* vol. 47, no. D1, pp. D330-D338, 2018, doi: 10.1093/nar/gky1055.
[55] A. Kozomara, M. Birgaoanu, and S. Griffiths-Jones, "miRBase: from microRNA sequences to function," *Nucleic Acids Research,* vol. 47, no. D1, pp. D155-D162, 2019, doi: 10.1093/nar/gky1141.
[56] A. Fabregat *et al.*, "The Reactome Pathway Knowledgebase," (in eng), *Nucleic Acids Res,* vol. 46, no. D1, pp. D649-d655, Jan 4 2018, doi: 10.1093/nar/gkx1132.
[57] R. Oughtred *et al.*, "The BioGRID interaction database: 2019 update," (in eng), *Nucleic Acids Res,* vol. 47, no. D1, pp. D529-d541, Jan 8 2019, doi: 10.1093/nar/gky1079.
[58] L. Licata *et al.*, "SIGNOR 2.0, the SIGnaling Network Open Resource 2.0: 2019 update," *Nucleic Acids Research,* vol. 48, no. D1, pp. D504-D510, 2020, doi: 10.1093/nar/gkz949.
[59] I. Rodchenkov *et al.*, "Pathway Commons 2019 Update: integration, analysis and exploration of pathway data," *Nucleic Acids Research,* vol. 48, no. D1, pp. D489-D497, 2019, doi: 10.1093/nar/gkz946.
[60] M. Martens *et al.*, "WikiPathways: connecting communities," (in eng), *Nucleic Acids Res,* vol. 49, no. D1, pp. D613-d621, Jan 8 2021, doi: 10.1093/nar/gkaa1024.
[61] G. Joshi-Tope *et al.*, "Reactome: a knowledgebase of biological pathways," *Nucleic Acids Research,* vol. 33, no. suppl_1, pp. D428-D432, 2005, doi: 10.1093/nar/gki072.
[62] D. Szklarczyk *et al.*, "The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored," (in eng), *Nucleic Acids Res,* vol. 39, no. Database issue, pp. D561-8, Jan 2011, doi: 10.1093/nar/gkq973.
[63] C. von Mering *et al.*, "STRING: known and predicted protein-protein associations, integrated and transferred across organisms," (in eng), *Nucleic Acids Res,* vol. 33, no. Database issue, pp. D433-7, Jan 1 2005, doi: 10.1093/nar/gki005.
[64] T. S. Keshava Prasad *et al.*, "Human Protein Reference Database--2009 update," (in eng), *Nucleic Acids Res,* vol. 37, no. Database issue, pp. D767-72, Jan 2009, doi: 10.1093/nar/gkn892.

[65] F. Ceccarelli, D. Turei, A. Gabor, and J. Saez-Rodriguez, "Bringing data from curated pathway resources to Cytoscape with OmniPath," (in eng), *Bioinformatics*, vol. 36, no. 8, pp. 2632-2633, Apr 15 2020, doi: 10.1093/bioinformatics/btz968.

[66] J. K. Huang *et al.*, "Systematic Evaluation of Molecular Networks for Discovery of Disease Genes," *Cell Syst*, vol. 6, no. 4, pp. 484-495 e5, Apr 25 2018, doi: 10.1016/j.cels.2018.03.001.

[67] N. del Toro *et al.*, "The IntAct database: efficient access to fine-grained molecular interaction data," *Nucleic Acids Research*, vol. 50, no. D1, pp. D648-D653, 2021, doi: 10.1093/nar/gkab1006.

[68] T. Slater, "Recent advances in modeling languages for pathway maps and computable biological networks," *Drug Discovery Today*, vol. 19, no. 2, pp. 193-198, 2014/02/01/ 2014, doi: https://doi.org/10.1016/j.drudis.2013.12.011.

[69] N. L. Novère *et al.*, "The Systems Biology Graphical Notation," *Nat Biotechnol*, vol. 27, no. 8, pp. 735-741, 2009/08/01 2009, doi: 10.1038/nbt.1558.

[70] E. Demir *et al.*, "The BioPAX community standard for pathway data sharing," *Nat Biotechnol*, vol. 28, no. 9, pp. 935-942, 2010/09/01 2010, doi: 10.1038/nbt.1666.

[71] The MeLoDy Lab. "BioRECIPE documentation." https://melody-biorecipe.readthedocs.io/en/latest/ (accessed.

[72] S. M. Keating *et al.*, "SBML Level 3: an extensible format for the exchange and reuse of biological models," *Molecular Systems Biology*, vol. 16, no. 8, p. e9110, 2020, doi: https://doi.org/10.15252/msb.20199110.

[73] P. Di Lena, G. Wu, P. L. Martelli, R. Casadio, and C. Nardini, "MIMO: an efficient tool for molecular interaction maps overlap," *BMC bioinformatics*, vol. 14, no. 1, p. 159, 2013/05/15 2013, doi: 10.1186/1471-2105-14-159.

[74] Y. Tian, R. C. McEachin, C. Santos, D. J. States, and J. M. Patel, "SAGA: a subgraph matching tool for biological graphs," *Bioinformatics*, vol. 23, no. 2, pp. 232-239, 2006, doi: 10.1093/bioinformatics/btl571.

[75] C. Hansen, J. Kisslinger, N. Krishna, E. Holtzapple, Y. Ahmed, and N. Miskov-Zivanov, "Classifying Literature Extracted Events for Automated Model Extension," ed: bioRxiv, 2021.

[76] C. Vega, V. Grouès, M. Ostaszewski, R. Schneider, and V. Satagopam, "BioKC: a collaborative platform for systems biology model curation and annotation," *bioRxiv*, p. 2020.10.01.322438, 2020, doi: 10.1101/2020.10.01.322438.

[77] P. Gawron *et al.*, "MINERVA—a platform for visualization and curation of molecular interaction networks," *npj Systems Biology and Applications*, vol. 2, no. 1, p. 16020, 2016/09/22 2016, doi: 10.1038/npjsba.2016.20.

[78] B. K. Kuntal, A. Dutta, and S. S. Mande, "CompNet: a GUI based tool for comparison of multiple biological interaction networks," *BMC bioinformatics*, vol. 17, no. 1, p. 185, 2016/04/26 2016, doi: 10.1186/s12859-016-1013-x.

[79] P. Shannon *et al.*, "Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks," *Genome Research*, vol. 13, no. 11, pp. 2498-2504, 2003, doi: 10.1101/gr.1239303.

[80] D. Morselli Gysi *et al.*, "Whole transcriptomic network analysis using Co-expression Differential Network Analysis (CoDiNA)," *PLOS ONE*, vol. 15, no. 10, p. e0240523, 2020, doi: 10.1371/journal.pone.0240523.

[81] V. Chelliah *et al.*, "BioModels: ten-year anniversary," *Nucleic Acids Research*, vol. 43, no. D1, pp. D542-D548, 2014, doi: 10.1093/nar/gku1181.
[82] T. Helikar *et al.*, "The Cell Collective: Toward an open and collaborative approach to systems biology," *BMC Systems Biology*, vol. 6, no. 1, p. 96, 2012/08/07 2012, doi: 10.1186/1752-0509-6-96.
[83] D. Pratt *et al.*, "NDEx, the Network Data Exchange," *Cell Syst*, vol. 1, no. 4, pp. 302-305, 2015, doi: 10.1016/j.cels.2015.10.001.
[84] F. Büchel *et al.*, "Path2Models: large-scale generation of computational models from biochemical pathway maps," *BMC Systems Biology*, vol. 7, no. 1, p. 116, 2013/11/01 2013, doi: 10.1186/1752-0509-7-116.
[85] A. Chatr-aryamontri *et al.*, "MINT: the Molecular INTeraction database," (in eng), *Nucleic Acids Res*, vol. 35, no. Database issue, pp. D572-4, Jan 2007, doi: 10.1093/nar/gkl950.
[86] L. A. Harris *et al.*, "BioNetGen 2.2: advances in rule-based modeling," (in eng), *Bioinformatics (Oxford, England)*, vol. 32, no. 21, pp. 3366-3368, 2016, doi: 10.1093/bioinformatics/btw469.
[87] K. Sayed, Y. Kuo, A. Kulkarni, and N. Miskov-Zivanov, "DiSH simulator: Capturing dynamics of cellular signaling with heterogeneous knowledge," in *2017 Winter Simulation Conference (WSC)*, 3-6 Dec. 2017 2017, pp. 896-907, doi: 10.1109/WSC.2017.8247841.
[88] N. Miskov-Zivanov, M. S. Turner, L. P. Kane, P. A. Morel, and J. R. Faeder, "The Duration of T Cell Stimulation Is a Critical Determinant of Cell Fate and Plasticity," *Science Signaling*, vol. 6, no. 300, pp. ra97-ra97, 2013, doi: 10.1126/scisignal.2004217.
[89] C. A. Telmer *et al.*, "Dynamic system explanation: DySE, a framework that evolves to reason about complex systems - lessons learned," presented at the Proceedings of the Conference on Artificial Intelligence for Data Discovery and Reuse, Pittsburgh, Pennsylvania, 2019.
[90] R. S. Wang, A. Saadatpour, and R. Albert, "Boolean modeling in systems biology: an overview of methodology and applications," (in eng), *Phys Biol*, vol. 9, no. 5, p. 055001, Oct 2012, doi: 10.1088/1478-3975/9/5/055001.
[91] J. D. Schwab, S. D. Kühlwein, N. Ikonomi, M. Kühl, and H. A. Kestler, "Concepts in Boolean network modeling: What do they all mean?," *Computational and Structural Biotechnology Journal*, vol. 18, pp. 571-582, 2020/01/01/ 2020, doi: https://doi.org/10.1016/j.csbj.2020.03.001.
[92] N. Miskov-Zivanov, P. Zuliani, Q. Wang, E. M. Clarke, and J. R. Faeder, "High-level modeling and verification of cellular signaling," in *2016 IEEE International High Level Design Validation and Test Workshop (HLDVT)*, 7-8 Oct. 2016 2016, pp. 162-169, doi: 10.1109/HLDVT.2016.7748271.
[93] R. Zhang *et al.*, "Network model of survival signaling in large granular lymphocyte leukemia," *Proceedings of the National Academy of Sciences*, vol. 105, no. 42, pp. 16308-16313, 2008, doi: 10.1073/pnas.0806447105.
[94] J. L. Puga, M. Krzywinski, and N. Altman, "Bayesian networks," *Nature Methods*, vol. 12, no. 9, pp. 799-800, 2015/09/01 2015, doi: 10.1038/nmeth.3550.
[95] S. Y. Lee, "Temozolomide resistance in glioblastoma multiforme," *Genes & Diseases*, vol. 3, no. 3, pp. 198-210, 2016/09/01/ 2016, doi: https://doi.org/10.1016/j.gendis.2016.04.007.
[96] S. Kulkarni, S. Goel-Bhattacharya, S. Sengupta, and B. H. Cochran, "A Large-Scale RNAi Screen Identifies SGK1 as a Key Survival Kinase for GBM Stem Cells," (in eng), *Mol Cancer Res*, vol. 16, no. 1, pp. 103-114, Jan 2018, doi: 10.1158/1541-7786.Mcr-17-0146.

[97] M. Ceccarelli *et al.*, "Molecular Profiling Reveals Biologically Discrete Subsets and Pathways of Progression in Diffuse Glioma," (in eng), *Cell,* vol. 164, no. 3, pp. 550-63, Jan 28 2016, doi: 10.1016/j.cell.2015.12.028.

[98] B. B. Liau *et al.*, "Adaptive Chromatin Remodeling Drives Glioblastoma Stem Cell Plasticity and Drug Tolerance," (in eng), *Cell stem cell,* vol. 20, no. 2, pp. 233-246.e7, Feb 2 2017, doi: 10.1016/j.stem.2016.11.003.

[99] M. A. Valenzuela-Escarcega, G. Hahn-Powell, M. Surdeanu, and T. Hicks, "A Domain-independent Rule-based Framework for Event Extraction," in *ACL*, 2015.

[100] J. N. Weinstein *et al.*, "The cancer genome atlas pan-cancer analysis project," *Nature genetics,* vol. 45, no. 10, p. 1113, 2013.

[101] E. Clough and T. Barrett, "The Gene Expression Omnibus Database," (in eng), *Methods Mol Biol,* vol. 1418, pp. 93-110, 2016, doi: 10.1007/978-1-4939-3578-9_5.

[102] I. Papatheodorou *et al.*, "Expression Atlas: gene and protein expression across multiple studies and organisms," (in eng), *Nucleic acids research,* vol. 46, no. D1, pp. D246-D251, 2018, doi: 10.1093/nar/gkx1158.

[103] H. Huang, S. Zhang, W. J. Shen, H. S. Wong, and D. Xie, "Gene set enrichment ensemble using fold change data only," (in eng), *J Biomed Inform,* vol. 57, pp. 189-203, Oct 2015, doi: 10.1016/j.jbi.2015.07.019.

[104] M. Crow, N. Lim, S. Ballouz, P. Pavlidis, and J. Gillis, "Predictability of human differential gene expression," *Proceedings of the National Academy of Sciences,* vol. 116, no. 13, p. 6491, 2019, doi: 10.1073/pnas.1802973116.

[105] H. Mi, A. Muruganujan, D. Ebert, X. Huang, and P. D. Thomas, "PANTHER version 14: more genomes, a new PANTHER GO-slim and improvements in enrichment analysis tools," *Nucleic Acids Research,* vol. 47, no. D1, pp. D419-D426, 2018, doi: 10.1093/nar/gky1038.

[106] Q. Wei, I. K. Khan, Z. Ding, S. Yerneni, and D. Kihara, "NaviGO: interactive tool for visualization and functional similarity and coherence analysis with gene ontology," *BMC bioinformatics,* vol. 18, no. 1, p. 177, 2017/03/20 2017, doi: 10.1186/s12859-017-1600-5.

[107] D. Blanco-Melo *et al.*, "SARS-CoV-2 launches a unique transcriptional signature from in vitro, ex vivo, and in vivo systems," *bioRxiv,* p. 2020.03.24.004655, 2020, doi: 10.1101/2020.03.24.004655.

[108] A. Mo *et al.*, "Disease-specific regulation of gene expression in a comparative analysis of juvenile idiopathic arthritis and inflammatory bowel disease," *Genome Medicine,* vol. 10, 12/01 2018, doi: 10.1186/s13073-018-0558-x.

[109] B. J. Gill *et al.*, "MRI-localized biopsies reveal subtype-specific differences in molecular and cellular composition at the margins of glioblastoma," *Proceedings of the National Academy of Sciences,* vol. 111, no. 34, pp. 12550-12555, 2014, doi: 10.1073/pnas.1405839111.

[110] V. Costa *et al.*, "New somatic mutations and WNK1-B4GALNT3 gene fusion in papillary thyroid carcinoma," (in eng), *Oncotarget,* vol. 6, no. 13, pp. 11242-11251, 2015/05// 2015, doi: 10.18632/oncotarget.3593.

[111] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: a practical and powerful approach to multiple testing," *Journal of the Royal statistical society: series B (Methodological),* vol. 57, no. 1, pp. 289-300, 1995.

[112] N. Fiorini *et al.*, "Best Match: New relevance search for PubMed," (in eng), *PLoS biology,* vol. 16, no. 8, pp. e2005343-e2005343, 2018, doi: 10.1371/journal.pbio.2005343.

[113] H. Wakimoto *et al.*, "Maintenance of primary tumor phenotype and genotype in glioblastoma stem cells," (in eng), *Neuro-oncology,* vol. 14, no. 2, pp. 132-44, Feb 2012, doi: 10.1093/neuonc/nor195.

[114] M. M. Sherry-Lynes, S. Sengupta, S. Kulkarni, and B. H. Cochran, "Regulation of the JMJD3 (KDM6B) histone demethylase in glioblastoma stem cells by STAT3," (in eng), *PLoS One,* vol. 12, no. 4, p. e0174775, 2017, doi: 10.1371/journal.pone.0174775.

[115] S. Parylo, A. Vennepureddy, V. Dhar, P. Patibandla, and A. Sokoloff, "Role of cyclin-dependent kinase 4/6 inhibitors in the current and future eras of cancer treatment," (in eng), *J Oncol Pharm Pract,* vol. 25, no. 1, pp. 110-129, Jan 2019, doi: 10.1177/1078155218770904.

[116] L. Annovazzi *et al.*, "The DNA damage/repair cascade in glioblastoma cell lines after chemotherapeutic agent treatment," *International journal of oncology,* vol. 46, no. 6, pp. 2299-2308, 2015, doi: 10.3892/ijo.2015.2963.

[117] C. Trejo-Solís *et al.*, "Autophagic and Apoptotic Pathways as Targets for Chemotherapy in Glioblastoma," (in eng), *Int J Mol Sci,* vol. 19, no. 12, p. 3773, 2018, doi: 10.3390/ijms19123773.

[118] R. Yang *et al.*, "The Hippo transducer TAZ promotes cell proliferation and tumor formation of glioblastoma cells through EGFR pathway," (in eng), *Oncotarget,* vol. 7, no. 24, pp. 36255-36265, Jun 14 2016, doi: 10.18632/oncotarget.9199.

[119] K. Wang, D. Chen, Z. Qian, D. Cui, L. Gao, and M. Lou, "Hedgehog/Gli1 signaling pathway regulates MGMT expression and chemoresistance to temozolomide in human glioblastoma," *Cancer Cell International,* vol. 17, no. 1, p. 117, 2017/12/04 2017, doi: 10.1186/s12935-017-0491-x.

[120] A. H. Shih and E. C. Holland, "Notch signaling enhances nestin expression in gliomas," (in eng), *Neoplasia (New York, N.Y.),* vol. 8, no. 12, pp. 1072-82, Dec 2006, doi: 10.1593/neo.06526.

[121] K. Sayed, C. A. Telmer, and N. Miskov-Zivanov, "Motif modeling for cell signaling networks," in *2016 8th Cairo International Biomedical Engineering Conference (CIBEC)*, 15-17 Dec. 2016 2016, pp. 114-117, doi: 10.1109/CIBEC.2016.7836133.

[122] C. Jean-Quartier, F. Jeanquartier, and A. Holzinger, "Open Data for Differential Network Analysis in Glioma," (in eng), *Int J Mol Sci,* vol. 21, no. 2, Jan 15 2020, doi: 10.3390/ijms21020547.

[123] N. Tuncbag *et al.*, "Network Modeling Identifies Patient-specific Pathways in Glioblastoma," (in eng), *Scientific reports,* vol. 6, p. 28668, 2016, doi: 10.1038/srep28668.

[124] N. Masuda, M. Sakaki, T. Ezaki, and T. Watanabe, "Clustering Coefficients for Correlation Networks," (in English), *Frontiers in Neuroinformatics,* Original Research vol. 12, no. 7, 2018-March-15 2018, doi: 10.3389/fninf.2018.00007.

[125] A. E. Teschendorff, C. R. S. Banerji, S. Severini, R. Kuehn, and P. Sollich, "Increased signaling entropy in cancer requires the scale-free property of protein interaction networks," *Scientific reports,* vol. 5, pp. 9646-9646, 2015 2015, doi: 10.1038/srep09646.

[126] R. McLendon *et al.*, "Comprehensive genomic characterization defines human glioblastoma genes and core pathways," *Nature,* vol. 455, no. 7216, pp. 1061-1068, 2008/10/01 2008, doi: 10.1038/nature07385.

[127] G. Zhou, K.-W. Liang, and N. Miskov-Zivanov, *Intervention Pathway Discovery via Context-Dependent Dynamic Sensitivity Analysis*. 2019.