Towards the Automation of Expanding Dynamic Network Models with Knowledge from Literature

by

Yasmine Ahmed

B.S. in Systems and Biomedical Engineering, Cairo University, Egypt, 2014

M.Sc. in Biomedical Engineering, School of Communication and Information Technology, Nile University, Egypt, 2016

Submitted to the Graduate Faculty of the

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2022

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This dissertation was presented

by

Yasmine Ahmed

It was defended on

April 7, 2022

and approved by

Zhi-Hong Mao, Ph.D., Professor, Department of Electrical and Computer Engineering

Samuel Dickerson, Ph.D., Assistant professor, Department of Electrical and Computer Engineering

Heng Huang, Ph.D., Professor, Department of Electrical and Computer Engineering

Réka Albert, Distinguished Professor of Physics and Biology, Pennsylvania State University

Dissertation Director: Natasa Miskov-Zivanov, Ph.D., Assistant professor, Department of Electrical and Computer Engineering

Towards the Automation of Expanding Dynamic Network Models with Knowledge from Literature

Yasmine Ahmed, PhD

University of Pittsburgh, 2022

Creating computational models of world complex systems, including intracellular and intercellular bionetworks, geopolitical, economic, environmental, and agricultural world systems, is a time and labor-intensive task which is often limited by the knowledge and experience of modelers. This has naturally led to the emergence of the idea of automating the process of building new models or extending existing models, which could have a significant potential in enabling rapid, consistent, comprehensive and robust analysis of complex systems.

Inspired by this idea, we propose in this work different novel approaches for expanding models using the information extracted from literature with machine reading engines. Our proposed approaches combine machine reading with clustering, and graph theoretical analysis to create an automated framework for efficient model assembly. Furthermore, by automatically extending models with the information published in literature, our proposed methods allow for collecting the existing information in a consistent and comprehensive way. This, in turn, facilitates information reuse, data reproducibility, and replacing hundreds/thousands of manual experiments, thereby reducing the time needed for the advancement of knowledge.

We tested how well each method can reproduce manually built and curated models in different biological domains, when provided with varying amount of information in the baseline model and in the machine reading output. In particular, we have demonstrated the reliability of the proposed methods using three different selected models, namely, T cell differentiation, T cell large

granular lymphocyte, and pancreatic cancer cell. Experimental results reveal considerable improvements of our approaches over other related methods. Moreover, using our automated model extension approach, we are able to efficiently find the best set of extensions to reproduce the manually extended models. Besides demonstrating automated reconstruction of a model that was previously built manually, our methods can assemble multiple models that satisfy desired system properties. As such, it replaces large number of tedious or even impractical manual experiments and guides alternative hypotheses and interventions in biological systems. Finally, we explored different model versions and system property testing results in order to develop a heuristic to modify model update rules.

Keywords: Automated model assembly, model recommendation, graph clustering, text mining, natural language processing, statistical model checking, stochastic simulations, Boolean modeling, discrete mechanistic models, query answering, colitis associated colon cancer.

Table of Contents

List of Tables

List of Figures

Preface

All the praises and thanks be to Almighty God, Allah, Who has guided me to this, and never could I have found guidance, were it not that Allah has guided me.

I would like to express my sincere gratitude to my advisor, Professor Natasa Miskov-Zivanov, whose motivating force, insightful feedback and well-bounded advice helped me to significantly improve my work. Besides her expertise in formulating research questions and methodology, she is friendly, cooperative, and good-natured.

I would also like to thank the members of my dissertation committee, Professor Zhi-Hong Mao, Professor Samuel Dickerson, Professor Heng Huang and Professor Réka Albert.

I would like to acknowledge my colleagues and collaborators at Melody lab, Cheryl Telmer, Khaled Sayed, Kara Bocan, Sandra Cascio, Adam Butchy, Emilee Holtzapple, Gaoxiang Zhou, Casey Hansen and Stefan Andjelkovic.

I am deeply grateful to my Dad for his assistance, support, prayers and belief in me. You will always be my first love, my backbone and my best friend.

I would like to extend my sincere thanks to my beloved Mom, my irreplaceable life partner Mohamed Abd-Elmagid and our beautiful daughter Alya, my charming mother-in-law, my sisters: Narimane, Lamia and Salma Ahmed and my brothers: Islam (gone from our sights but never from our hearts) and Mohamed.

I would like to thank my professors for their insightful comments and suggestions, in specific, Dr. Amro Eljaroudi, Dr. Mahmoud Elnokali, Dr. Ahmed Fahmy, Dr. Mohamed Owis, Dr. Tamer Basha, Dr. Mahmoud Ismail and Dr. Inas Yassine.

I would like to offer my special thanks to my friends, Nada Younes, Mena Yehya, Tasnim Akram, Rana Ahmed, Doaa Mahmoud, Sara Chérif, Walaa Gomaa, Eman Habib, Aya Fawzy, Mona Ramadan, Safaa Mohamed, Zinab Gamal, Shaimaa Eleseily, Ayshenur, Yassin Khalifa, Faezeh Movahedi, Ahmed Hassan, Rashad Eletreby, Amr Mahmoud, Mohamed Zaghloul, Randa Osama, Shahd Bahaa, Yasmine Mahmoud, Alaa Ibrahim, Busra Susam, Zeineb Bouzid, Wafaa Abuelrub, Norhane Charkawi, Menna Achraf, Susanna Matta. Thank you all for being a part of this journey. I could not have achieved that without your endless support.

1.0 Introduction

1.1 Motivation

Modeling has an important role in the process of explaining complex systems or extending existing models with new information. It guides data collection, allows for capturing the dynamics of these systems, helps identify gaps in our understanding, and thus, often leads to new questions and the search for missing information (Clarke & Fisher, 2020; Epstein, 2008a).

In biology, there are several approaches that have been introduced to model the dynamics of biological systems. The ordinary differential equations (ODEs) is a common approach used to describe the rate of production or consumption of cellular components in a reaction network (Faeder et al., 2009). Systems modeled with ODEs follow the *event-based modeling* formalism, in which the system can be fully represented by a set of *events*. One possible limitation of this type modeling approach is with *granularity* when the number of network components (such as receptors, ligands, kinases) is large. Even with the availability of fast methods such as BioNetGen (Smith et al., 2012) and RuleBender (Smith et al., 2012), such approach requires quantitative knowledge about network details necessary for accurate ODE-based models. Modelers are familiar only with indirect cause-effect relationships for a number of events in the network, since there is a limited knowledge about exact mechanisms and parameters necessary to create ODEs.

Therefore, it is often necessary and more practical to work at higher levels of abstraction due to a lack of information required to develop such models. *Element-based modeling* approaches have been suggested to overcome the issue of missing information while still providing important insights into system behavior (R. Albert & Thakar, 2014; Miskov-Zivanov, Turner, et al., 2013a;

Schwab et al., 2020a). In element-based approaches, a model consists of multiple elements (which determines granularity) with defined interactions. Discrete variables (which determines the resolution) are associated with each element, and regulation functions or update rules allow simulation of the model by calculating next-state values of each element variable over time. Element-based approaches balance model simplicity with accuracy (Sayed, Telmer, et al., 2018; C. A. Telmer, Bocan, et al., 2019). This approach studies an element as a unit, in which the states change over time as determined by its regulation functions. In this work, we focus on element-based models and in particular, on discrete models. In section 2.1, we show a detailed example of a discrete model of colon cancer that we have recently created.

Model creation is highly dependent on human input, it requires reading hundreds of papers to extract useful information, incorporating background and common-sense knowledge of domain experts, and conducting wet lab experiments(J. Fisher & Henzinger, 2007).

Moreover, the amount of biological data is constantly growing, further augmenting the issues of data inconsistency and fragmentation(Frisoni et al., 2021; Valenzuela-Escárcega et al., 2015). Therefore, the automation of model building, and even more, of model extension, when new information becomes available, or when the domain knowledge advances, is a critical next step for computational modeling. Such automation will not only lead to more efficient modeling due to reducing the amount of slow human interventions, but will also allow for more consistent, comprehensive and robust modeling process.

Recently, there has been a push in the field of synthetic biology to automate the entire pathway of model assembly, starting with collecting biological interactions, assembling a model, and performing simulations. Approaches like Path2Models (Büchel et al., 2013) and INDRA(Gyori et al., 2017a), will automatically generate quantitative or qualitative models based

on the granularity of the information they are given. Unfortunately, the utility and accuracy of the assembled models are dependent upon the modeling approach, biological context, and correctness of the starting information. These techniques rely on accurate information, and their performance suffers when the interaction information is inaccurate, incomplete, or from a different biological context.

1.2 Dissertation Scope

In this dissertation, we are mostly interested in assembling computational models that explain how biomolecular signaling pathways regulate cell functions (G.T. Zañudo et al., 2018; Schwab et al., 2020b). Usually, modelers start with a few seed components and their interactions, which summarizes domain experts' knowledge about the system, or with a baseline model that can be found in curated public model databases such as Reactome (Fabregat et al., 2018), STRING (von Mering et al., 2005), KEGG (Aoki & Kanehisa, 2005), or in published literature. Depending on the questions to be answered by the model, the baseline model is often further extended with the information extracted from literature or obtained from experts (Miskov-Zivanov, 2015).

In order to automate the collection of articles and information extraction, one begins with a formal search query, which is defined according to a question posed about the modeled system. The search query guides automated selection of articles that contain relevant information from published literature databases. As the biomedical literature mining tools are becoming essential for the high throughput extraction of knowledge from scientific papers, we use in our work existing machine reading engines. We then use the extracted information to extend or assemble models in order to answer questions about the system under investigation (Etzioni et al., 2006).

In (Liang et al., 2017), the authors proposed a method that starts with a baseline model and selects interactions automatically extracted from published work. The goal of (Liang et al., 2017) was to build a model that satisfies pre-defined requirements or to identify new therapeutic targets, formally expressed as existing or desired system properties. As results in (Liang et al., 2017) demonstrate, automatic model extension is a promising approach for accelerating modeling, and consequently, disease treatment design. The authors in (Liang et al., 2017) organize the information extracted from literature into layers, based on their proximity to the baseline model. Recently, another extension method that uses a Genetic Algorithm (GA) was proposed in (Sayed, Bocan, et al., 2018). The GA-based approach was able to extract a set of extensions that led to the desired behavior of the final extended model. The disadvantages of the GA-based approach include non-determinism, as the solution may vary across multiple algorithm executions on the same inputs, as well as issues with scalability.

Therefore, there is a pressing need for developing a tool and a methodology that automatically and efficiently assembles the information extracted from available literature into models, tests the newly assembled models, and selects the most suitable model to address user questions. In contrast to the previously published work (Liang et al., 2017; Sayed, Bocan, et al., 2018), our approach focuses on identifying clusters of strongly connected elements in the newly extracted information, that have a measurable impact when added to the model. Once the interactions extracted from the literature are clustered, we score their performance on a selected set of system properties, using stochastic simulation methods (Sayed, Kuo, et al., 2018) and statistical model checking (Wang, 2016). The scoring helps determine which clusters to add to the baseline model. The process of selecting and scoring the interactions takes at most a few hours to

execute thousands of experiments *in silico*, which would take days, or months, or would be impractical to conduct in vivo or in vitro.

1.3 Dissertation Contributions

In this work, our main goal is to develop versatile and efficient methods that can be used to extend many different dynamic network models automatically and without human interventions. In particular, we are interested in developing innovative approaches that utilize new combinations of extracted data to address previously intractable questions. This is achieved by collecting the existing information from literature in a rapid, consistent and comprehensive way, which in turns will facilitate information reuse and data reproducibility, and will replace hundreds or thousands of manual experiments, since the time needed for the advancement of knowledge is reduced. The contributions of this work include:

Main contribution #1 We propose ACCORDION (ACCelerating and Optimizing model RecommenDatIONs), a new method to extend dynamic network models, that combines clustering with simulation and formal analysis in order to test the possible candidate models on a set of formally written desired system properties.

Main contribution #2 We propose MELOGRAPH (Model Extension using coLlabOration GRAPH), an automated framework for rapid model assembly that combines machine reading, the frequency class-based metric, and graph analysis. Using MELOGRAPH, we will explore the utility of several graph centrality metrics for guiding model extension.

Main contribution #3 We propose CLARINET (CLARIfying NETworks), an automated, fast methodology and a tool that selects relevant knowledge from published papers and suggests

most useful model extensions. CLARINET extracts the events from literature as a collaboration graph, including several metrics that rely on the event occurrence and co-occurrence frequency in literature.

Main contribution #4 Exploring different dynamic network model versions and system property testing results in order to develop a heuristic to modify model update rules.

Main contribution #5 Applying our proposed model extension methods on several other models in different biological domains, e.g., T cell differentiation, T cell large granular lymphocyte, and pancreatic cancer cell.

1.4 Dissertation Organization

The dissertation is organized as follows. In chapter 2, we provide a background on discrete models, the nature and the representation of information extracted from literature, in addition to the model representation format that will be used throughout the thesis. We will also provide a brief explanation of two different model analysis approaches, common graph network metrics, and an overview of the three use cases for three published models namely, T cell differentiation model, Pancreatic cancer cell model, that we will use to demonstrate the main steps of our proposed methods. In Chapter 3, 4 and 5, we describe the details of implementation and the results of our proposed methods and tools, ACCORDION, MELOGRAPH and CLARINET. For each method, we describe a detailed explanation of the methodology as well as the main contributions and the outcomes of applying each tool on the use cases described in Chapter 2. In chapter 6, we present a comprehensive comparative analysis between our proposed methods and other competitive methods that have been recently proposed. In chapter 7, we summarize the conclusions of the

research presented in this dissertation and the possible future directions to be investigated as the next steps for this research.

2.0 Background

2.1 Computational Model of Signaling Pathways in Ulcerative Colitis (UC) and Colitis-Associated Colon Cancer (CACC)

Computational mechanistic models of within-cell signal transduction networks can explain how these networks integrate internal and external inputs to give rise to the appropriate cellular response (Gómez Tejeda Zañudo et al., 2017). These models can be used in cancer cells, whose aberrant decision-making regarding their survival or death, proliferation or quiescence can be connected to errors in the state of nodes or edges of the signal transduction network. Reaction network models require detailed quantitative data that are not always available (Miskov-Zivanov, Turner, et al., 2013a). However, the use of logic-based, discrete dynamic models does not require quantitative parameters needed in a reaction network model, but rather enables the development of complex qualitative networks. Moreover, it enables the identification of results that are mainly due to the organization of the signaling network, and those that also depend on the kinetics of individual events. Network-based models such as this will play an increasing role in the rational design of high-order therapeutic combinations.

In this study, we have uncovered novel regulatory axes interconnecting macrophages and colon cells involved in ulcerative colitis (UC) and colitis-associated colon cancer (CACC). Patients with UC have an increased risk of developing CACC. Changes in glycosylation of the oncoprotein MUC1 commonly occur in chronic inflammation, including UC, and this abnormally glycosylated MUC1 has been shown to promote cancer development and progression. What causes changes in glycosylation of MUC1 is not known. Gene expression profiling of myeloid cells in inflamed and

malignant colon tissues showed increased expression levels of inflammatory macrophage-associated cytokines compared to normal tissues. We analyzed the involvement of these cytokines and macrophages that produced them in the induction of aberrant MUC1 glycoforms. A co-culture system was used first to examine the effects of M1 and M2 macrophages on glycosylation-related enzymes in colon cancer cells. M2-like macrophages induced the expression of the glycosyltransferase ST6GALNAC1, an enzyme that adds sialic acid to O-linked GalNAc residues, promoting the formation of tumor-associated sialyl-Tn (sTn) O-glycans. Immunostaining of UC and CACC tissue samples confirmed the elevated number of M2-like macrophages as well as high expression of ST6GALNAC1 and the altered MUC1-sTn glycoform on colon cells. Cytokine arrays and blocking antibody experiments indicated that the macrophage-dependent ST6GALNAC1 activation was mediated by IL-13 and CCL17. We demonstrated that IL-13 promoted phosphorylation of STAT6 to activate transcription of ST6GALNAC1. A computational model of signaling pathways was assembled and used to test IL-13 inhibition as a possible therapy. Our findings reveal a novel cellular crosstalk between colon cells and macrophages within the inflamed and malignant colon that contributes to the pathogenesis of UC and CACC.

The protein interactions involved in signaling pathways investigated in this study were collected from the literature to be entered using the BioRECIPES tabular format (Sayed, Telmer, et al., 2018), described in Section 2.3, a model representation format that includes, for each model element: name, type (protein, gene or a chemical), cellular location, number of possible discrete states, and formatted list of regulators. Three levels are used to represent activation or inhibition of elements. Specifically, level 0 if the element has low activity, level 1 if the element has moderate activity and 2 has high activity. From the tabular representation, an executable discrete model is created using element update functions generated from the formatted regulator lists. Simulations

of the model were performed using the publicly available stochastic simulator, DiSH (Sayed, Kuo, et al., 2018), described in Section 2.4.1.

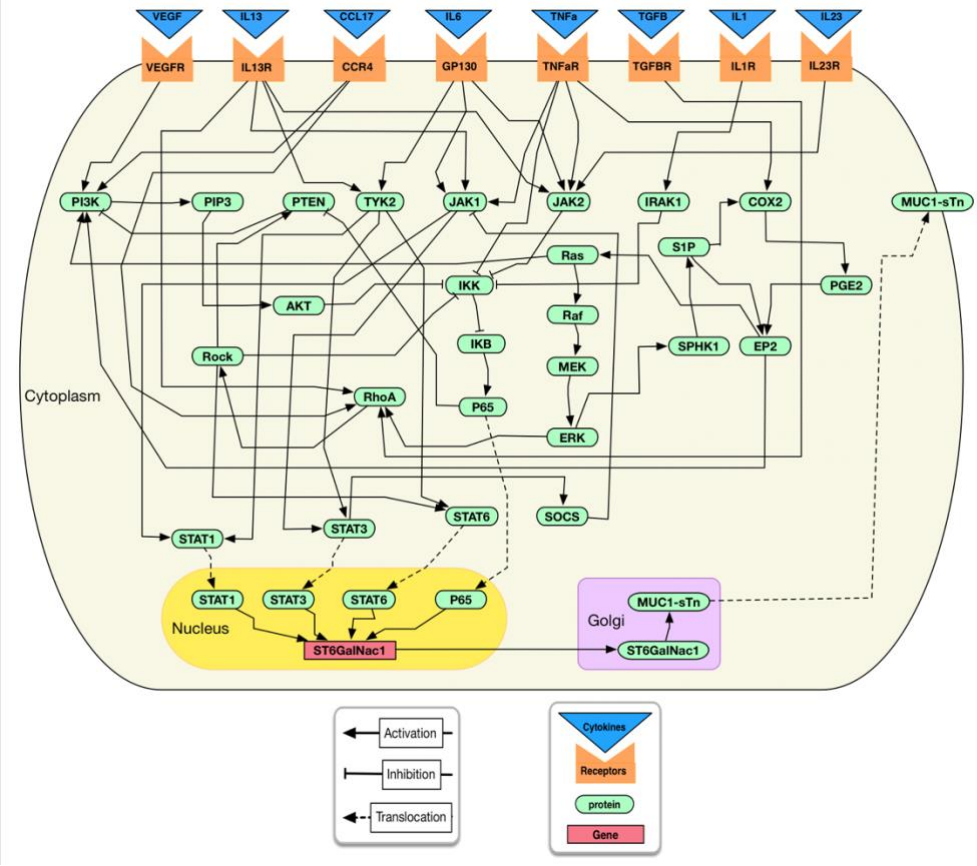


Figure 1 Interaction map of the UC and CACC model. Pointed arrows represent activation; blunted arrows represent inhibition. The cytokines (triangles) were selected from experiments and represented as inputs, these ligands bound to the receptors (orange shape) at the plasma membrane, and the signal was transduced across the membrane by activating the receptors. Signaling cascades then relayed the signal through the cytosol to the transcription factors STAT1, STAT3, STAT6, and AKT. The latter were translocated into the nucleus to regulate the ST6GALNAC1 gene (rectangle) to influence the amount of enzyme in the Golgi and ultimately the glycosylation of the extracellular sTn form of MUC1

Different experimental conditions, scenarios, are defined by assigning initial values to all model elements, and a set of inputs for scenarios of normal, UC, CACC and IL-13 inhibitor. Many independent runs of the scenario represent multiple cells in an experiment that have the same starting point but traverse through time steps differently. From these individual simulation runs,

we computed average trajectories to plot and visualize element behavior over time. We assembled a computational model that incorporates our experimental data and what has been reported in the literature to simulate signaling pathways potentially involved in the development of UC and CACC (Figure 1). The granular modeling framework utilizes a standardized tabular framework (Section 2.3) and the DiSH simulator (Section 2.4.1). In this modeling approach discrete elements change state depending on the influences of positive or negative regulators. The inputs are ligands present in the tumor microenvironment (Figure 2(a)) that stimulate signaling cascades to influence the ST6GALNAC1 gene and MUC1 protein of colon cells (Figure 1). The stochastic simulation results reflected the experimental results for the normal, UC and CACC scenarios, when microenvironment signaling ligands are used as inputs (Figure 2(a), (b)). Higher levels of IL13 and CCL17 observed in M2 co-culture with HT-29 cells and CACC result in increased transcription of ST6GALNAC1 and MUC1 sTn mediated by p-STAT6 and P65. Introduction of an IL-13 inhibitor resulted in decreased sTn only when introduced to UC and not CACC suggesting early intervention as a therapeutic strategy (Figure 2(c)). The model has IL-6 and IL-13 activating STAT6, however IL-6 is present in CACC and therefore the IL-13 inhibitor alone does not prevent ST6GALNAC1 transcription as the model still predicts its expression will occur downstream of IL-6. Experimental data and the computational model (Kvorjak et al., 2019) indicate that the IKK/IKB/P65 pathway is active in the presence of IL13 and CCL17 regulating the PI3K/AKT pathway through inhibition of PTEN.

The proposed computational model also provided indications for the involvement of NF-kB and suggests future studies including VEGF-A signaling. Further studies will investigate these results and will allow for rapid testing of multiple scenarios to replace difficult and expensive experiments.

Cytokines	Normal	UC	CACC
CCL17	Low	High	High
IL-1	Low	High	Low
IL-13	Low	Medium	High
IL-23	Low	High	Medium
IL-6	Low	Low	High
TGFB	Low	Medium	High
TNFA	Low	Medium	Low
VEGFA	Low	High	High

(a)

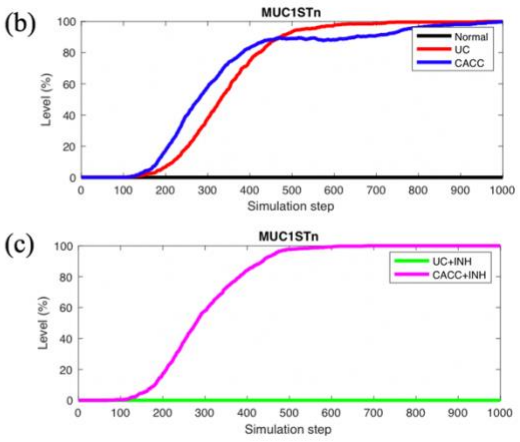


Figure 2 (a) Table showing cytokine input levels for UC and CACC. (b) Simulation results showing the average behavior from 200 runs over 1000 time steps for MUC1 sTn for three different scenarios (Normal, UC and CACC). (c) Simulation results showing the average behavior from 200 runs over 1000 time steps for MUC1 sTn for two different scenarios (UC + IL-13 inhibitor and CACC + IL-13 inhibitor)

2.2 Information Extraction from Literature

Extraction from literature usually starts with a question, for example, "How is PTEN regulation involved in T-cell fate?" We can write these questions as logical expressions (Figure 3(a)). These formally written queries are used to search public literature databases (e.g., PubMed (Roberts, 2001)) as illustrated in (Figure 3(b)). Once the relevant papers are selected, they are sent to machine reading engines for automated extraction of information (Figure 3(b)).

The state-of-the-art automated reading engines ((Burns et al., 2016; Valenzuela-escárcega et al., 2018)) are capable of finding hundreds of thousands of events in cellular signaling pathways from thousands of papers, in a few hours. Events in the machine reading output represent interactions between biochemical entities, such as post-translational modifications (e.g., binding, phosphorylation, ubiquitination, etc.), transcription, translation, translocation, and increase or

decrease of amount or activity. In the context of biomedical literature, entities are usually proteins, chemicals, genes, and RNAs, although sometimes they also represent biological processes. For each extracted entity, reading engines provide its name, the database where it is characterized, and the database identifier (ID) for the entity. Machine reading also collects the evidence, usually a sentence from which the event was extracted. For our case study, we used an open-source reading engine, REACH (Valenzuela-escárcega et al., 2018), to quickly obtain information from biomedical literature. In (Figure 4(a)), we show two example sentences. The REACH reading engine extracts events into an interaction-based format shown in (Figure 4(b)). We will refer to the list of interactions retrieved from literature in this format as *reading output*. The graphical representation or the interaction map of REACH outputs is illustrated in (Figure 4(c)), pointed arrows represent activation (increase in the amount or activity), blunt arrows represent inhibition (decrease in the amount or activity).

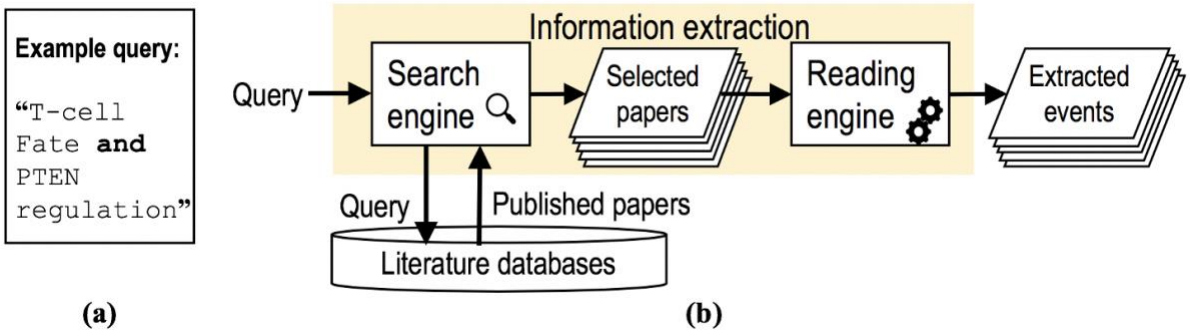


Figure 3 (a) Example query used to select relevant papers, (b) Main components of information extraction from relevant papers.

(a)

Example 1: However, recent data implicate the transcription factor FoxO1 in the induction of Treg, and phosphorylation of FoxO1 by Akt was established as a mechanism by which Akt/mTOR signaling inhibits Treg differentiation.

Example 2: Transient suppression of PTEN at low dose arises from the inclusion of two regulatory mechanisms in the model: the requirement for MEK1 to activate PTEN, and inhibition of mTORC2 by Akt

(b)

Regulator name	Regulated name	Interaction type	Regulator type	Uniprot ID	Regulated type	Uniprot ID
Akt	FoxO1	Inhibition (--|)	Protein	P31749	Protein	Q12778
Akt	mTORC2	Inhibition (--|)	Protein	P31749	Protein	P42345
MEK1	PTEN	Activation (→)	Protein	Q02750	Protein	P60484

(c)

Figure 4 (a) Two example sentences with highlighted entities and events that are extracted by machine readers, (b) Tabular outputs from REACH engine when reading example sentences from (a), (c) Graphical representation of REACH outputs.

2.3 Model Representation and Executable Models

We can automatically translate the three rows in the table in (Figure 4(b)) into the element-based BioRECIPES format (Sayed, Telmer, et al., 2018), which is then used as input to the executable model generation (see Section 2.2). The BioRECIPES tabular model representation format is illustrated in (Figure 5(a)) with several examples of molecules and interactions in T cells (Miskov-Zivanov, Turner, et al., 2013a). In the examples, PTEN is positively regulated by Foxp3, and negatively regulated by TCR. Ras has one positive regulator, TCR, and no negative regulators. IL-2 has positive and negative regulators, Ras and Foxp3, respectively.

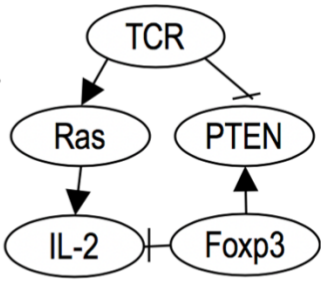
The BioRECIPES representation format includes, for each model element: (i) name, (ii) type (protein, gene, RNA, or a chemical), (iii) identifier from a database (e.g., UniProt (Bateman et al., 2017)), (iv) variable that represents state, and (v) set of regulators. While the BioRECIPES format is a sufficient representation for all the relevant element and interaction information, all interactions in a model can also be represented as a directed graph $G(V, E)$, with a set of nodes V and a set of directed edges E. Each node $v \in V$ corresponds to one model element, and each edge $e(v_i, v_j) \in E$ represents a directed interaction in which element v_i regulates element v_j. The graphical representation of all model interactions is often referred to as an influence map, and it is especially useful for the extension methods proposed in this work, as will be discussed in Chapter 3. In (Figure 5(b)), we show a graph of element interactions that are listed in the table. As can be seen in the graph, we include the information about the sign of the interaction in the form of arrow type, a pointed arrow represents positive regulation (activation), while a blunt arrow represents negative regulation (inhibition).

We will refer to the set of regulators of an element as its influence set, distinguishing between positive and negative regulators. Additionally, we can define a vector of all variables representing states of model elements as $x = (x_1,\ldots, x_N)$, where $N=|V|$ is the total number of model elements. If we use Boolean variables, then $x_i \in \{0, 1\}$, where $i=1,\ldots..,N$. Next, we can assign a state transition function to any model element, which defines a state change of the element, given the states of its regulators. We will refer to these functions as *element update rules* and to the model with update rules as an *executable model*. In the case of Boolean variables representing element states, the basic operations are AND (*), OR (+) and NOT (!). For example, one version of update rules for the small graph in Figure 5 can be: PTEN = Foxp3 * !TCR, Ras = TCR, and IL-2 = Ras * !Foxp3. The choice between AND and OR operation depends on the available

information about interactions and element regulations. For example, for an element to be "activated", all necessary regulators are combined with an AND operation, and all sufficient regulators with an OR operation.

Element name	Element type	Uniprot ID	Positive regulators	Negative regulators	Initial value
TCR	Protein	P01850			0
Ras	Protein	P01116	TCR		0
IL2	Protein	P60568	Ras	Foxp3	0
PTEN	Protein	Q92574	Foxp3	TCR	1
Foxp3	Protein	Q9BZS1	TCR		0

(a)



(b)

Figure 5 (a) Tabular representation of several elements and their influence sets (positive and negative regulators) in BioRECIPES format, (b) graphical repre-sentation of elements and influence sets.

2.4 Model Analysis

In this section, we describe two methods that we use to analyze the models extended with the newly obtained information and data.

2.4.1 Stochastic Simulation

We use the DiSH simulator (Sayed, Kuo, et al., 2018) to observe dynamic behavior of the baseline model and the extended models. DiSH can simulate networks with multi-valued elements in both deterministic and stochastic manner, and we utilize both of these features in our extension

analysis, as shown later in Chapter 3. Each simulation run starts with a specified initial model state, where initial values are assigned to all model elements to represent a particular system state (e.g., naïve T cell, regulatory T cell, etc.). Next, we use element update rules to determine element state transitions. We track element changes for a predefined number of simulation steps, or until a steady state is reached (Sayed, Kuo, et al., 2018).

Furthermore, this approach has been used for simulation and analysis of discrete models, such as those described in (Miskov-Zivanov, Turner, et al., 2013a; Zhang et al., 2008) and was previously incorporated into several other tools ((I. Albert et al., 2008; Helikar et al., 2012; Naldi et al., 2018). If we assume that a simulation run has M steps, we define a trajectory of element x_i in the kth run as a time course of its state values $T_k(x_i)= (x_{i0}, x_{i1},\ldots, x_{iM})_k$ in time steps t=0,..,M. When the simulator is in the stochastic mode, in each simulation step, only one element is randomly chosen, and its new value is computed according to its update rule. Depending on the information available, the rates at which elements are updated can be different across model elements; when there is limited information about elements, we choose to use the same update rate for all elements. In either case, due to the randomness in element update order, multiple runs that start with the same initial state may result in different (non-deterministic) state transitions, and thus, in different trajectories of state changes in time. DiSH simulations output a file that includes all the simulated trajectories for all model elements, in other words, for K runs, for each model element x_i, we obtain its simulated trajectories $T(x_i)=\{T_1(x_i), T_2(x_i)\ldots, T_K(x_i)\}$. These trajectories can be used to plot and visualize behavior over time for any given element. Typically, averaged trajectories are plotted ((Miskov-Zivanov, Turner, et al., 2013a; Zhang et al., 2008)), where an average element state value is computed across all trajectories, in each simulation step.

2.4.2 Statistical Model Checking

In this work, we use statistical model checking ((Kumar-Jha et al., 2009; Wang et al., 2016)) to test all generated models against formally defined properties. Model checking is often used to verify whether a model of a system, or a system design, satisfies a set of properties describing expected behavior of the system. Each property is encoded into Bounded Linear Temporal Logic (BLTL) ((Kumar-Jha et al., 2009; Tkachev & Abate, 2013)). Here, we use statistical model checking since the state transitions are not necessarily deterministic, and we follow the simulation approach described in Section 2.3.1. To avoid a full state space search, statistical model checking conducts randomized sampling to generate simulation trajectories of the model and performs statistical analysis on those trajectories. The input to the statistical model checker is a system property expressed as a BLTL formula, and the output is a probability estimate (P) that the model satisfies a given property, under particular error interval for the estimate. For instance, let us assume that we would like to test a property that, at any point within the first s_1 time steps, element v_i becomes 1 and element v_j becomes 0, and that they both keep those values for at least s_2 time steps. We would then write the formula:

$$\boldsymbol{F^{s_1} G^{s_2} (v_i = 1 \wedge v_j = 0)} \quad \textbf{(2.1)}$$

where F^{s_1} stands for "any time in the future s_1 steps", and G^{s_2} stands for "globally for s_2 steps".

2.5 Common Graph Metrics

As a brief background for the analysis that we will describe in chapters 3 and 4, we provide here definitions of static network characteristics (average path length (*APL*), clustering coefficient (*Coeff*), and graph density) (Newman, 2003a) as well as several node and edge centrality metrics commonly used in network theory (degree, neighborhood connectivity, betweenness centrality, closeness centrality, radiality, and edge betweenness) (Scardoni & Lau, 2012). We will later discuss some of these metrics in the context of the new metrics that we propose in chapter 4.

Degree (*D*) of a node v_i is the number of its adjacent edges in an undirected graph, and in a directed graph, is the sum of the in-degree (the number of incoming edges) and the out-degree (the number of outgoing edges) of v_i.

Assuming that a distance $d(v_i, v_j)$ is the number of edges on a shortest path between nodes v_i and v_j, *APL* is computed as an average distance across all possible pairs of nodes in the graph:

$$\boldsymbol{APL} = \frac{\mathbf{1}}{|\boldsymbol{V^{new,*}}|\cdot(|\boldsymbol{V^{new,*}}|-\mathbf{1})} \cdot \sum_{\boldsymbol{v_i, v_j \in V^{new,*}, v_i \neq v_j}} \boldsymbol{d(v_i, v_j)} \quad \textbf{(2.2)}$$

where $|V^{new,*}|$ is the number of nodes in the graph. If there is no path between v_i and v_j, then $d(v_i, v_j)=0$.

The clustering coefficient (*Coeff*) (Newman, 2003a) is computed for each node v_i in a directed graph as:

$$\boldsymbol{Coeff(v_i)} = \frac{\boldsymbol{T(v_i)}}{\boldsymbol{D^{tot}(v_i)\cdot(D^{tot}(v_i)-1)-2\cdot D^{\leftrightarrow}(v_i)}} \quad \textbf{(2.3)}$$

where $T(v_i)$ is the number of triangles (three connected nodes) in the graph that contain node v_i, and $D^{\leftrightarrow}(v_i)$ is the reciprocal of $D(v_i)$. *Coeff* is a number between 0 and 1, and therefore, if an average *Coeff* value, computed across all graph nodes, approaches 0, the graph is more likely to contain stars, while when this value approaches 1, the graph is a clique.

The graph density (Newman, 2003b) is defined for a directed graph as:

$$\boldsymbol{graph\ density} = \frac{|E^{new,*}|}{|V^{new,*}|(|V^{new,*}|-1)} \quad \textbf{(2.4)}$$

where $|E^{new,*}|$ is the number of edges and $|V^{new,*}|$ is the number of nodes in the graph. A graph is considered to be dense if the number of edges is close to the maximum number of possible edges, therefore, the graph density is close to 1 for a dense graph and close to 0 for a sparse graph.

Neighborhood Connectivity (*NC*) of a node is an average of the *D* value of all its neighbors, where a neighbor is a node connected with the given node via an edge. Nodes with more neighbors tend to have these neighbors more connected (larger *NC*), while nodes with few neighbors usually have their neighbors less connected (smaller *NC*).

Betweenness Centrality (*BC*) of a node is the number of shortest paths between any other couple of nodes, that pass through the given node. High *BC* value for a node indicates that the node, for certain paths, is crucial to maintain node connections.

Closeness Centrality (*CC*) of a node is an inverse of a sum of the lengths of shortest paths between the node and all the other nodes in the graph. Higher *CC* value for a node indicates more proximity to other nodes.

Radiality (*R*) of a node is computed by first finding the lengths of shortest paths between the node and every other node in the graph, then subtracting the value of the diameter (the maximal possible distance between nodes) from each shortest path length, and finally adding all the resulting values. If a node radiality is high compared to the average radiality of the network, this means that the node is generally closer to the other nodes, however, if the radiality is low, the node is peripheral.

2.6 Case Studies

2.6.1 Naïve T cell Differentiation Case Study

Naïve peripheral T cells are stimulated via antigen presentation to T cell receptor (TCR) and with co-stimulation at CD28 receptor. This stimulation results in the activation of several downstream pathways, feedback and feedforward loops between pathway elements, which then lead to the differentiation of naïve T cells into helper (Th) or regulatory (Treg) phenotypes. The distribution between Th and Treg cells within the T cell population depends on antigen dose; for instance, high antigen dose results in prevalence of Th cells, while low antigen dose leads to a mixed population of Th and Treg cells. The key markers that are commonly used to measure the outcomes of the naïve T cell differentiation into Th and Treg cells are IL2 and Foxp3, respectively. In other words, Th cells are characterized by high expression of IL-2 and low expression of Foxp3, and Treg cells are characterized by high expression of Foxp3 and low expression of IL-2. To demonstrate our model extension procedure, we use two existing, manually built models of T cell differentiation, from (Miskov-Zivanov, Turner, et al., 2013a) and (Hawse et al., 2015a).

2.6.1.1 Baseline Model and Golden Model

In (Miskov-Zivanov, Turner, et al., 2013a), the authors proposed a model where most of the elements are assumed to have two main levels of activity, and are therefore represented with Boolean variables, and their update rules are logic functions. Additionally, the stimulation through TCR is assumed to have three different levels, no stimulation (TCR=0), low dose (TCR=1), and high dose (TCR=2), and therefore, it is implemented using two Boolean variables. We used the

model from (Miskov-Zivanov, Turner, et al., 2013b) to create the baseline model for our case study. The interaction map of this model is provided in (Miskov-Zivanov, Turner, et al., 2013b).

In (Hawse et al., 2015a), the authors have proposed an extension of the original T cell model from (Miskov-Zivanov, Turner, et al., 2013b), a new model that improved the behavior of the original model. Specifically, in the new model in (Hawse et al., 2015a), the Foxp3 response to low dose is closer to experimental observations, that is, it is present in almost 70% of the differentiated population, while in (Miskov-Zivanov, Turner, et al., 2013b) Foxp3 was present in 100% of the differentiated population. In both models, there is a brief transient induction of Foxp3 after the stimulation with high antigen dose. We will refer to the model from (Hawse et al., 2015a) as *the golden model*. For the baseline, we used the original model from (Miskov-Zivanov, Turner, et al., 2013b), without several interactions overlapping with the golden model from (Hawse et al., 2015a) (TCR activates PIP3, PIP3 activates Akt, Akt activates mTORC2 and mTORC2 inhibits Akt). While the model from (Miskov-Zivanov, Turner, et al., 2013b) satisfied a large number of system properties, except for a few that are satisfied by the model in (Hawse et al., 2015a) only, the baseline model in its reduced shape does not satisfy a larger set of system properties. Our aim is to use our proposed extension methods to automatically expand this baseline model in order to recapitulate the behavior of the golden model.

2.6.1.2 Set of Properties

From the golden model in (Hawse et al., 2015a) and the results of its studies, we define a set of properties that our final automatically extended model (as will be shown later in chapters 3 and 4) needs to satisfy. Specifically, the properties capture observed responses of key pathway components in T cells, Foxp3, IL-2, PTEN, CD25, STAT5, AKT, mTOR, mTORC2 and FoxO1, to three scenarios: (1) no stimulation (TCR=0), (2) stimulation with low antigen dose (TCR=1),

and (3) stimulations with high antigen dose (TCR=2). The complete list of 27 properties is shown in Table 1.

Table 1 The set of properties that are observed to be true in Tcell model (Miskov-Zivanov, Turner, et al., 2013b)(Hawse et al., 2015b) **written in BLTL format as well as the corresponding scenario, description in plain English and the goal property probability values**

Prop.	Description	BLTL format	Expected
Scenario 0: No TCR			
1	Once deactivated, AKT will remain inactive until end of analyzed period	F[1500]G[50](AKT==0)	1
2	Once activated, PTEN will remain active until end of analyzed period	F[1500]G[50](PTEN==1)	0.338
3	Once deactivated, FOXP3 will remain inactive until end of analyzed period	F[1500]G[50](FOXP3==0)	1
4	Once deactivated, IL2 will remain inactive until end of analyzed period	F[1500]G[50](IL2==0)	1
5	Once deactivated, CD25 will remain inactive until end of analyzed period	F[1500]G[50](CD25==0)	1
6	Once deactivated, STAT5 will remain inactive until end of analyzed period	F[1500]G[50](STAT5==0)	1
7	Once deactivated, mTORC1 will remain inactive until end of analyzed period	F[1500]G[50](MTORC1==0)	1
8	Once deactivated, mTORC2 will remain inactive until end of analyzed period	F[1500]G[50](MTORC2==0)	1
9	Once activated, FOXO1 will remain active until end of analyzed period	F[1500]G[50](FOXO1==1)	1
Scenario 1: Low TCR			
10	Once deactivated, AKT will remain inactive until end of analyzed period	F[1500]G[50](AKT==0)	1
11	Once activated, PTEN will remain active until end of analyzed period	F[1500]G[50](PTEN==1)	0.9
12	Once activated, FOXP3 will remain active until end of analyzed period	F[1500]G[50](FOXP3==1)	0.760
13	Once deactivated, IL2 will remain inactive until end of analyzed period	F[1500]G[50](IL2==0)	1
14	Once activated, CD25 will remain active until end of analyzed period	F[1500]G[50](CD25==1)	1
15	Once activated, STAT5 will remain active until end of analyzed period	F[1500]G[50](STAT5==1)	1
16	Once deactivated, mTORC1 will remain active until end of analyzed period	F[1500]G[50](MTORC1==0)	1
17	Once activated, mTORC2 will remain active until end of analyzed period	F[1500]G[50](MTORC2==1)	1
18	Once activated, FOXO1 will remain active until end of analyzed period	F[1500]G[50](FOXO1==1)	1
Scenario 2: High TCR			
19	Once activated, AKT will remain inactive until end of analyzed period	F[1500]G[50](AKT==1)	1
20	In developing Th, PTEN decreases and remains absent	F[1500]G[50](PTEN==0)	1
21	Once deactivated, FOXP3 will remain inactive until end of analyzed period	F[1500]G[50](FOXP3==0)	1
22	Once activated, IL2 will remain active until end of analyzed period	F[1500]G[50](IL2==1)	1
23	Once activated, CD25 will remain active until end of analyzed period	F[1500]G[50](CD25==1)	1
24	Once activated, STAT5 will remain active until end of analyzed period	F[1500]G[50](STAT5==1)	1
25	Once activated, mTORC1 will remain inactive until end of analyzed period	F[1500]G[50](MTORC1==1)	1
26	Once activated, mTORC2 will remain active until end of analyzed period	F[1500]G[50](MTORC2==1)	1
27	Once activated, FOXO1 will remain active until end of analyzed period	F[1500]G[50](FOXO1==0)	1

2.6.2 T cell Large Granular Lymphocyte (T-LGL) Leukemia

The T cell large granular lymphocyte (T-LGL) leukemia is a disease characterized by an abnormal increase of cytotoxic T lymphocytes (CTLs) (Zhang et al., 2008). As described by authors in (Zhang et al., 2008), similar to normal activated CTL, leukemic T-LGL exhibit activation of multiple survival signaling pathways. Unlike normal activated CTL, leukemic T-LGL are not sensitive to Fas-induced apoptosis, a process essential for activation-induced cell death. There is no curative therapy yet known for this disease. Hence, there is a crucial need to identify potential therapeutic targets. A discrete dynamic model has been proposed in (Zhang et al., 2008) to understand the signaling components that determine the survival of CTL cells in T-LGL leukemia. The model incorporates the signaling pathways involved in normal cytotoxic T cells activation and the known deregulations of survival signaling in leukemic T-LGL. The model includes proteins, mRNAs, small molecules such as lipids and biological processes that indicate cell fate such as cytoskeleton signaling, proliferation and apoptosis.

2.6.2.1 Baseline Model and Golden Model

For the TLGL case study, the model published in (Zhang et al., 2008)will serve as the golden model, whereas the baseline model for this study will be created by removing all direct regulators of the 19 model elements that were identified by (Saadatpour et al., 2011) as therapeutic targets. According to (Saadatpour et al., 2011), these elements are: BID, Caspase, Ceramide, DISC, ERK, GAP, IL2RB, IL2RBT, JAK, MCL1, MEK, NFKB, PDGFR, PI3K, RAS, S1P, SOCS, SPHK1 and STAT3.

2.6.2.2 Set of Properties

Here, we define the set of properties that our final automatically extended model needs to satisfy as the properties that capture the observed responses of the 19 therapeutic targets in the golden model under one scenario (Table 2).

Table 2 The set of properties that are observed to be true in T-LGL(Zhang et al., 2008) **model written in BLTL format as well as the corresponding scenario, description in plain English and the goal property probability values**

Prop. #	Description	BLTL format	Expected value
1	Once deactivated, DISC will remain inactive until end of analyzed period	F[1500]G[50](DISC==0)	1
2	Once deactivated, Ceramide will remain inactive until end of analyzed period	F[1500]G[50](Ceramide==0)	1
3	Once deactivated, Caspase will remain inactive until end of analyzed period	F[1500]G[50](Caspase==0)	1
4	Once activated, SPHK1 will remain active until end of analyzed period	F[1500]G[50](SPHK1==1)	1
5	Once activated, S1P will remain active until end of analyzed period	F[1500]G[50](S1P==1)	1
6	Once activated, PDGFR will remain active until end of analyzed period	F[1500]G[50](PDGFR==1)	1
7	Once deactivated, GAP will remain inactive until end of analyzed period	F[1500]G[50](GAP==0)	1
8	Once activated, RAS will remain active until end of analyzed period	F[1500]G[50](RAS==1)	1
9	Once activated, MEK will remain active until end of analyzed period	F[1500]G[50](MEK==1)	1
10	Once activated, ERK will remain active until end of analyzed period	F[1500]G[50](ERK==1)	1
11	Once activated, IL2RBT will remain active until end of analyzed period	F[1500]G[50](IL2RBT==1)	1
12	Once activated, IL2RB will remain active until end of analyzed period	F[1500]G[50](IL2RB==1)	1
13	Once activated, STAT3 will remain active until end of analyzed period	F[1500]G[50](STAT3==1)	1
14	Once deactivated, BID will remain inactive until end of analyzed period	F[1500]G[50](BID==0)	1
15	Once activated, MCL1 will remain active until end of analyzed period	F[1500]G[50](MCL1==1)	1
16	Once deactivated, SOCS will remain inactive until end of analyzed period	F[1500]G[50](SOCS==0)	1
17	Once activated, JAK will remain active until end of analyzed period	F[1500]G[50](JAK==1)	1
18	Once activated, PI3K will remain active until end of analyzed period	F[1500]G[50](PI3K==1)	1
19	Once activated, NFKB will remain active until end of analyzed period	F[1500]G[50](NFKB==1)	1

2.6.3 Pancreatic Cancer Microenvironment

Although some cancers such as breast and colon can be managed, others such as glioma and pancreatic cancer have very poor survival rates (C. A. Telmer, Sayed, et al., 2019). As described in (C. A. Telmer, Sayed, et al., 2019), pancreatic cancer has early KRas activating mutations followed by TP53 and CDN2A inactivating mutations in the majority of tumors. Although the pancreatic cancer has a known mutational profile disrupting signaling pathways, there are no available drugs to target KRas activation or restore tumor suppressor function, and therefore, survival of patients has not improved. The modeling of pancreatic cancer is of great importance since it could reveal molecular mechanisms important for disease treatment. Therefore, the focus of such model is to include the major signaling pathways, metabolism and the tumor microenvironment. The pancreatic cancer model proposed in (C. A. Telmer, Sayed, et al., 2019) describes the hallmarks of cancer (which are represented as the processes of apoptosis, autophagy, cell cycle progression, inflammation, immune response, oxidative phosphorylation and proliferation) and suggests combinations of inhibitors as therapies.

2.6.3.1 Baseline Model and Golden Model

The model in (C. A. Telmer, Sayed, et al., 2019) serves as a golden model for our PCC case studies. We removed from the model in (C. A. Telmer, Sayed, et al., 2019) a subset of paths that have autophagy, apoptosis and proliferation as their target nodes to create three different baseline models for the PCC case studies. We designed each experiment with a different baseline model. This is achieved by removing the paths that connect a source node that initiates a specific biological process such as (autophagy, apoptosis and proliferation) to a target node which will be this biological process. For instance, in PCC BM^{Au}, according to the evidence that mTORC1

initiates autophagy (Muilenburg et al., 2014), we remove the paths that link mTORC1 and autophagy and the outcome will be our first baseline model that corresponds to PCC BM^{Au}. Similarly, for PCC BM^{Ap} and PCC BM^{Pr}, based on the fact that TGFβ1 regulates apoptosis (Siegel & Massagué, 2003) and KRas mutations enhances proliferation (Bardeesy & DePinho, 2002), we have also created the baseline models that correspond to PCC BM^{Ap} and PCC BM^{Pr}, respectively by removing the paths that connect source nodes (TGFβ1, KRas) to target nodes (apoptosis and proliferation). In this case study, the golden model will be the whole PCC model network (C. A. Telmer, Sayed, et al., 2019).

2.6.3.2 Set of Properties

Using the golden model and the descriptions from (C. A. Telmer, Sayed, et al., 2019), we wrote the BLTL expressions of 21 system properties that capture the behavior of seven key elements (apoptosis, autophagy, cell cycle progression, immune response, inflammation, oxidative phosphorylation and proliferation) under three different scenarios, (1) normal, (2) with injury and (3) with KRas, TP53 and CDN2A mutation (C. A. Telmer, Sayed, et al., 2019) (Table 3).

Table 3 The set of properties that are observed to be true in the PCC model (C. A. Telmer, Sayed, et al., 2019)**written in BLTL format as well as the corresponding scenario, description in plain English and the goal property probability values**

Prop. #	Description	BLTL format	Expected value
Scenario 0: Normal			
1	Once deactivated, apoptosis will remain inactive until end of analyzed period	F[9500]G[50](apoptosis==0)	1
2	Once deactivated, autophagy will remain inactive until end of analyzed period	F[9500]G[50](autophagy==0)	1
3	Once deactivated, cell cycle progression will remain inactive until end of analyzed period	F[9500]G[50](cell cycle progression==0)	1
4	Once deactivated, immune response will remain inactive until end of analyzed period	F[9500]G[50](immune response==0)	1
5	Once deactivated, inflammation will remain inactive until end of analyzed period	F[9500]G[50](inflammation==0)	1
6	Once deactivated, oxidative phosphorylation will remain inactive until end of analyzed period	F[9500]G[50](oxidative phosphorylation==0)	1
7	Once deactivated, proliferation will remain inactive until end of analyzed period	F[9500]G[50](proliferation==0)	1
Scenario 1: With injury			
9	Once deactivated, apoptosis will remain inactive until end of analyzed period	F[9500]G[50](apoptosis==0)	1
9	Once deactivated, autophagy will remain inactive until end of analyzed period	F[9500]G[50](autophagy==0)	1
10	Once deactivated, cell cycle progression will remain inactive until end of analyzed period	F[9500]G[50](cell cycle progression==0)	1
11	Once deactivated, immune response will remain inactive until end of analyzed period	F[9500]G[50](immune response==0)	1
12	Once activated, inflammation will remain active until end of analyzed period	F[9500]G[50](inflammation==1)	1
13	Once activated, oxidative phosphorylation will remain active until end of analyzed period	F[9500]G[50](oxidative phosphorylation==1)	1
14	Once deactivated, proliferation will remain inactive until end of analyzed period	F[9500]G[50](proliferation==0)	1
Scenario 2: With KRas, TP53, CDN2A mutation			
15	Once deactivated, apoptosis will remain inactive until end of analyzed period	F[9500]G[50](apoptosis==0)	1
16	Once activated, autophagy will remain active until end of analyzed period	F[9500]G[50](autophagy==1)	1
17	Once activated, cell cycle progression will remain active until end of analyzed period	F[9500]G[50](cell cycle progression==1)	1
18	Once deactivated, immune response will remain inactive until end of analyzed period	F[9500]G[50](immune response==0)	1
19	Once activated, inflammation will remain active until end of analyzed period	F[9500]G[50](inflammation==1)	1
20	Once deactivated, oxidative phosphorylation will remain inactive until end of analyzed period	F[9500]G[50](oxidative phosphorylation==0)	1
21	Once activated, proliferation will remain active until end of analyzed period	F[9500]G[50](proliferation==1)	1

3.0 Context-aware Information Selection and Model Recommendation with ACCORDION

3.1 Objective and Applicability

In this chapter, we propose ACCORDION (Ahmed et al., 2022) (ACCelerating and Optimizing model RecommenDatIONs), a novel methodology and a tool, that can be used to automatically and efficiently assemble the information extracted from literature into models and to recommend models that achieve desired dynamic behavior. In contrast to (Liang et al., 2017), ACCORDION focuses on identifying clusters of strongly connected elements in the newly extracted information, that have a measurable impact when added to the model. Once the interactions extracted from the literature are clustered, ACCORDION scores their performance on a selected set of system properties, using stochastic simulation methods (Sayed, Kuo, et al., 2018) and statistical model checking (Wang et al., 2016) described in Sections 2.4.1 and 2.4.2 respectively. The scoring helps determine which clusters to add to the baseline model. Therefore, ACCORDION takes at most a few hours to execute thousands of experiments in silico, which would take days, or months, or would be impractical to conduct in vivo or in vitro.

We designed ACCORDION to be versatile and to be used to extend many different models. To demonstrate the efficiency and utility of ACCORDION, we have selected nine different case studies using models of three systems, namely, the T cell differentiation model (Miskov-Zivanov, Turner, et al., 2013b), the T cell large granular lymphocyte model (Zhang et al., 2008) and the pancreatic cancer cell model (C. A. Telmer, Sayed, et al., 2019), and seven machine reading outputs with varying features. Our main goal is to show that ACCORDION, automatically, without

human intervention, recommends model improvements to significantly reduce baseline model error and recapitulate desired system behavior.

3.2 Proposed Methodology

The inputs and outputs of ACCORDION, as well as the main methods within the tool are outlined in Figure 6. The first step of our proposed methodology is creating an input for ACCORDION, which includes extracting new event information from literature by machine reading engines, followed by filtering, scoring and classifying these events. Once the new input is created, the three main steps within ACCORDION are performed, and they include (1) clustering of new events, (2) assembly of the clustered event data into models, and (3) selection of the most suitable and useful events. In the following subsections, we discuss each of these steps in detail.

3.2.1 Baseline Model (BM)

The baseline model can be obtained in many different ways, for example, it could be manually created with expert input, or adopted from models published in literature (Zhang et al., 2008), (Miskov-Zivanov, Turner, et al., 2013b), (C. A. Telmer, Sayed, et al., 2019), (Bianconi et al., 2012) and in model databases (Aoki & Kanehisa, 2005), (Fabregat et al., 2018),(Pillich et al., 2017). In general, ACCORDION works with models that have *directed cyclic graph* structure, $G(V,E)$, where each node $v \in V$ corresponds to one model element, representing a protein, gene, chemical, or a biological process, and each directed edge $e(v_i, v_j) \in E$ indicates that element v_j is regulated or influenced, directly or indirectly, by element v_i. We refer to the set of regulators of an

element as its *influence set*, distinguishing between positive and negative regulators. ACCORDION assigns to each element v a discrete variable x representing the element's state, such as a level of its activity or amount. Each model element may have a state transition function, referred to as *element update rule*, which defines its state changes given the states of its regulators, thus enabling the study of system dynamics. While the types of elements and their update rules (see Sections 3.2.3-3.2.7) are not constrained by the main methods implemented within ACCORDION, they are largely affected by the information that is available in new events (see Section 3.2.2) and in the baseline model. Most often, the events described in literature are qualitative, for example, only two element states (e.g., inactive/active, absent/present) may be distinguished or relevant, or only two or three levels of concentration may be considered (e.g., low/high or low/medium/high). Causal or Boolean types of regulations and update rules are most suitable in such cases and ACCORDION is also compatible with such qualitative information.

3.2.2 Candidate Event Set

A set of candidate events (CEs) is the second input to ACCORDION, which can be collected from different sources and created manually or automatically. Since the machine reading of published literature results in large event sets, and therefore, allows for a high throughput processing of available information, we will assume here such automated pipeline, including both machine readers (e.g., the ones described in (Valenzuela-escárcega et al., 2018), (Allen & Teng, 2017)) and INDRA database of interactions extracted from literature (Gyori et al., 2017b). The set of relevant papers can be selected either using search tools such as Google or PubMed (Roberts, 2001) or by providing key search terms to reading engines, which then directly use Medline search tools (e.g., PubMed (Roberts, 2001), Ovid (*Https://Www.Ovid.Com*, n.d.)) to find most relevant

papers. The former approach includes manual user step (using search tools to find papers to input to machine readers), but gives more flexibility to users when selecting relevant papers, while the latter approach allows for full automation, starting with the query entered by a user. In either case, machine readers process the selected papers and output a set of events. Examples of queries, sentences processed by machine readers, and events in the machine reading output are shown in Figure 6. As can be seen in the figure, each event has a direction (source and target of interaction) and sign (positive or negative regulation).

3.2.3 Network Creation and Return Path Definition

The CE set can be represented as a set of edges, E^{ext}, where the source and target nodes of these edges form set V^{ext}. From the baseline model graph $G^{BM}(V^{BM}, E^{BM})$ and the CE set, ACCORDION creates a new graph $G^{new}(V^{new}, E^{new})$, where $V^{new} = V^{BM} \cup V^{ext}$, and $E^{new} = E^{BM} \cup E^{ext}$. The edges $e(v_s,v_t)$ in E^{ext}, where v_s is the source node and v_t is the target node, can be classified into three categories: (i) both source node v_s and target node v_t are found in the baseline model: $\{v_s,v_t\} \in V^{BM}$; (ii) either the source node or the target node is found in the baseline model: ($v_s \in V^{BM}$ and $v_t \notin V^{BM}$) or ($v_s \notin V^{BM}$ and $v_t \in V^{BM}$); (iii) neither the source node nor the target node is found in the baseline model: $\{v_s,v_t\} \notin V^{BM}$. Adding the entire set of CEs to the baseline model all at once usually does not result in a useful and accurate model. Alternatively, we can add one interaction at a time and test each model version, which is time consuming, or even impractical, given that the number of models increases exponentially with the size of the CE set. Moreover, adding individual

interactions does not have an effect on the model when an interaction belongs to category (iii), and most often when it belongs to category (ii).

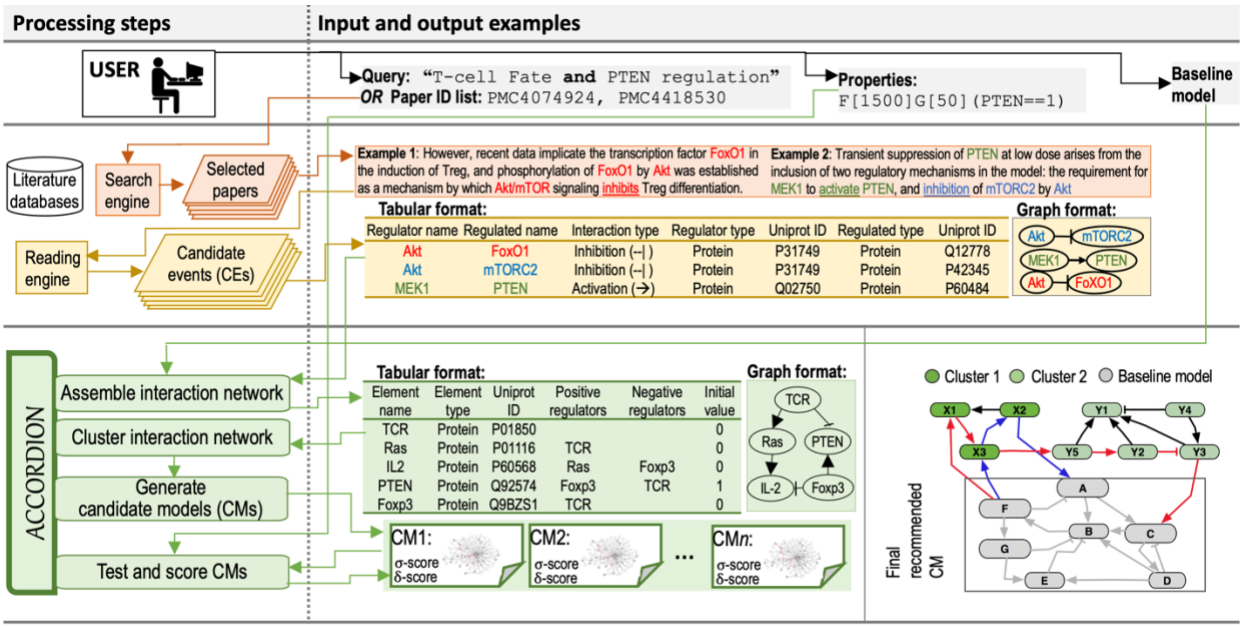


Figure 6 The diagram of the flow that includes a user, machine reading and ACCORDION. Input and output examples column: (Top) *Left*: Example query used to select relevant papers. *Right*: example property written in BLTL format. (Middle) Main components of information extraction from relevant papers. *Top*: Two example sentences with highlighted entities and events that are extracted by machine readers. *Bottom-Left*: Tabular outputs from REACH engine with Example 1 and Example 2 sentences as input. *Bottom-Right*: Graphical representation of REACH outputs. (Bottom) *Left*: Tabular representation of several elements and their influence sets (positive and negative regulators) in BioRECIPES format [45] and the graphical representation of elements and influence sets. *Right*: A toy example graph G^{new} of a baseline model and connected clusters: grey nodes belong to the baseline model, light and dark green nodes belong to the CE set obtained from machine reading, blue edges highlight a return path within one cluster, and red edges show a return path connecting two clusters. The multi-cluster path starts at G^{BM} (baseline model), continues through C_1 (cluster 1), then through C_2 (cluster 2), and ends in G^{BM}.

It proves much more useful to add paths of connected interactions, which are at the same time connected to the baseline model in their first and last nodes. Therefore, our approach for finding the most useful subset of the CE set includes finding connected interactions, that is, a set of edges in the graph G^{new} that form a return path.

We define a path of k connected edges as $e^{path}(v_{s1},v_{tk}) = (e_{i1}(v_{s1},v_{t1}), e_{i2}(v_{s2}=v_{t1},v_{t2}), e_{i3}(v_{s3}=v_{t2},v_{t3}), \ldots,e_{ik}(v_{sk}=v_{tk-1},v_{tk}))$, and we will refer to $e^{path}(v_{s1},v_{tk})$ as a return path, when$\{v_{s1},v_{tk}\}\in V^{BM}$ (Figure 6). ACCORDION searches for such return paths after clustering G^{new}.

3.2.4 Network Clustering

To find clusters in G^{new}, we apply Markov Clustering algorithm (MCL) (Schaeffer, 2007), an unsupervised graph clustering algorithm, commonly used in bioinformatics (e.g., clustering of protein-protein interaction networks (Brohée & van Helden, 2006), (Lei et al., 2016)). In (Vlasblom & Wodak, 2009), the authors showed that the MCL algorithm is tolerant to noise, while identifying meaningful clusters. A number of previous studies have demonstrated that the MCL algorithm outperforms other clustering techniques(Frey & Dueck, 2007), (King et al., 2004), (Blatt et al., 1996),(Bader & Hogue, 2003). The MCL algorithm has been proven to converge with undirected graphs (Vlasblom & Wodak, 2009), and therefore, ACCORDION provides to the MCL algorithm the information about node adjacency in G^{new}. Since we are interested in clustering a graph given its connectivity only, the information about adjacency without directionality is sufficient in this step. The directionality will be used in later steps when exploring dynamic behavior. In other words, the adjacency matrix M created this way is symmetric, mapping nodes in G^{new} to both row and column headers in M. The entries in matrix M are assigned value 1 when an edge between their column and row nodes exists in G^{new} or when an entry is on the main

diagonal of M (i.e., same column and row node), and value 0 otherwise. Next, the updated matrix M is used by the MCL algorithm as an initial version of a stochastic Markov matrix (Gagniuc, 2017), where each entry represents the probability of a transition from the column node to the row node. Since G^{new} is not a weighted graph, all transitions are assumed to be equally likely, and the matrix M is normalized such that the sum of entries in each column is equal 1. As mentioned earlier, graph G^{new} can be cyclic, and although the MCL algorithm has been previously applied to acyclic graphs (Mountasser et al., 2017), we still use the MCL algorithm for its speed, and our results show that it provides useful results when applied in automated model extension recommendation. MCL simulates random walks on an underlying interaction network (in our case, graph G^{new}), by alternating two operations, expansion and inflation. The probability of a random walk of length l between any two nodes can be calculated by raising the matrix M to the exponent l, a process called *expansion*. As the number of paths is likely larger between nodes within the same cluster than between nodes across different clusters, the transition probabilities between nodes in the same cluster will typically be higher in a newly obtained expanded matrix. MCL further amplifies this effect by computing entry-wise exponents of the expanded matrix, a process called *inflation*, which raises each element of the matrix to the power *r*. Clusters are determined by alternating expansion and inflation, until the graph is partitioned into subsets such that there are no paths between these subsets. The final number of generated clusters, $C_1,\ldots,C_n$, depends on the selected inflation parameter *r* (Schaeffer, 2007). As discussed above, ACCORDION clusters the entire G^{new} in order to account for the connectivity with the baseline model, and thus, it likely assigns parts of the baseline model to different clusters. Once the clusters are generated, since we are interested in adding the components of the CE set from the clusters to the entire baseline model,

we will refer to the CE (BM) part of a generated cluster l as C_l^{CE} (C_l^{BM}) and to the nodes and edges in such cluster subsets as $V^{C_l,CE}$ ($V^{C_l,BM}$) and $E^{C_l,CE}$ ($E^{C_l,BM}$), respectively.

3.2.5 Assembly of Candidate Model Networks

From the generated clusters and the baseline model, ACCORDION assembles multiple *candidate models* (*CMs*) as follows. ACCORDION can add clusters one at a time, or in groups. The more clusters or cluster groups are generated, the number of possible cluster combinations grows, and consequently, ACCORDION needs to assemble and test more models. In addition to that, in most cases V^{BM} is smaller than V^{ext}, and E^{BM} is smaller than E^{ext}, and thus, the number of new nodes and edges in a cluster tends to be relatively large compared to the size of the baseline model. Adding a large number of new nodes and edges to the baseline model at once can significantly change the structure and the behavior of the model. Therefore, the default approach in ACCORDION is to evaluate only individual clusters generated as described in Section 3.2.4, as well as clusters $C_{i,j}$, created by merging pairs of clusters C_i and C_j ($i,j = 1..n$, $i \neq j$). ACCORDION determines for each individual and merged cluster whether it forms a return path with the baseline model, and for each such cluster, ACCORDION creates a candidate model by adding the entire baseline model to the cluster. In other words, the number of created candidate models is equal the number of clusters (both individual and merged) that form a return path with the baseline model. As defined above, the clusters formed from the G^{new} graph can contain nodes and edges of the baseline model. Therefore, for those clusters (individual or merged) that were used to create candidate models, ACCORDION computes the *node overlap* (*NO*) value (Ahmed, Telmer, et al., 2021), as a ratio of those nodes in a cluster C_l that are present in the baseline model ($V^{C_l,BM} = V^{BM} \cap V^{C_l}$) and the total number of nodes within a cluster (V^{C_l}).

$$NO_l = \frac{|V^{C_l,BM}|}{|V^{C_l}|} \quad \textbf{(3.1)}$$

3.2.6 Executable Model Creation and Testing

In this section, we discuss an additional input to ACCORDION besides the baseline model and the CE set and how all three inputs are used to evaluate the dynamics of candidate models. The third input to ACCORDION includes a set of properties $\mathcal{T}$ defining desired dynamic behavior that the assembled model should satisfy. ACCORDION uses element update rules in the baseline model and the sign of influences (positive or negative) in the CE set to create new element update rules. For those elements that were already in the baseline model, but their influence set was extended after adding a cluster to the baseline model, ACCORDION modifies their update rules. When new elements with non-empty influence set are added to the baseline model, ACCORDION generates a new update rule for them. As stated previously, event information available in the CE set is often qualitative, for example, "*A positively regulates B*". Furthermore, if an update rule for element *B* in the baseline model already includes two positive regulators *C* and *D*, i.e., $x_B = f(x_C, x_D)$, then the new event from the CE set can be added to the update rule for *B* as $x_B = f(x_C, x_D)$ OR x_A, or $x_B = f(x_C, x_D)$ AND x_A (following the definition from Section 2.3, x_A, x_B, x_C, x_D are variables representing level or amount or activity of elements *A*, *B*, *C*, *D*, respectively). For elements with more than two discrete levels, ACCORDION can use max and min operators to determine the maximum or minimum influence from a given set of regulators. To select the CM that allows for most closely reproducing the experimentally observed or desired behaviors and, given the randomness in time and order of events in modeled systems, ACCORDION uses a combination of stochastic simulation and statistical model checking. The DiSH simulator, described in Section

2.4.1, is used to obtain the dynamic behavior of the baseline model and the CMs. DiSH is a stochastic simulator that can simulate models at different levels of abstraction, information resolution, and uncertainty. This range of simulation schemes is especially valuable when working with diverse information sources and inputs, such as the ones used by ACCORDION. Each simulation run starts with a specified initial model state, where initial values are assigned to all model elements to represent a particular system state (e.g., naïve or not differentiated cell, healthy cancer cell). The initial values for the baseline model elements (nodes in V^{BM}) are usually already known, however, the newly added elements (nodes in V^{ext}) need to be assigned initial values as well. Given that machine reading does not provide this information, we assume that all elements within the same cluster have the same initial value. ACCORDION runs a statistical model checker (Section 2.4.2) to verify whether the CMs satisfy a set of properties describing expected behavior of the modeled system. The model checker reads properties formally written using Bounded Linear Temporal Logic (BLTL) (Kumar-Jha et al., 2009), (Tkachev & Abate, 2013) and, for a given model $\mathcal{M}$ and a property t, it outputs a property probability estimate, $p_{t}^{\mathcal{M}}$, that model $\mathcal{M}$ satisfies property t, under predefined error interval for the estimate. For instance, we can test whether at any point within the first s_1 time steps, model element v_i (i.e., its state variable x_i) reaches value X_1 and element v_j (i.e., its state variable x_j) reaches value X_2, and they both keep those values for at least s_2 time steps. We write this property formally as $F^{s_1}G^{s_2}(x_i = X_1 \wedge x_j = X_2)$, where F^{s_1} stands for "any time in the future s1 steps", and G^{s_2} stands for "globally for s2 steps". An example of a property and its expected value are shown in Figure 6. To avoid a full state space search, the statistical model checker calls the simulator to generate element trajectories for a defined number of steps and then performs statistical analysis on those trajectories with respect to a given property (Miskov-Zivanov, Zuliani, et al., 2013), (Liang et al., 2017).

3.2.7 CM Scoring and Recommendation

Usually, we are interested in a model that can satisfy a property $t_j \in \mathcal{T}$ with high probability. However, in some cases, due to randomness in biological systems, the $p_{t^{\mathcal{M}}}$ value lower than 1 (e.g., $p_{t^{\mathcal{M}}} \geq 0.7$) is expected. In our case studies explored in Section 2.6 (and in Table 1, Table 2, Table 3), we will show examples of such properties. In order to provide the recommendation of top CMs that are closest to expected probability values for properties, we introduce several metrics. The first metric, *model property error*, determines the difference between an estimated probability value for property t_j for CMi, $p_{t_j}^{CM_i}$, and the *goal property probability* value for t_j, P_{t_j}: $\varepsilon_{t_j}^{CM_i} = |p_{t_j}^{CM_i} - P_{t_j}|$. Next, we compute *average model error*, across all tested properties $t_j \in \mathcal{T}$, for each CMi, $\varepsilon_{\mathcal{T},\text{avg}}^{CM_i}$, and *$\sigma$-score* for model *CMi* for the given set of properties as $\sigma_{\mathcal{T}}^{CM_i} = 1 - \varepsilon_{\mathcal{T},\text{avg}}^{CM_i}$. The larger σ-score for a model is, the closer the model is to satisfying all desired properties. We also define model *δ-score*, $N_{\mathcal{T},\delta}^{CM_i}$, as the percent of properties out of all properties in $\mathcal{T}$ for which: $\varepsilon_{t_j}^{CM_i} \leq \delta$. In other words, the parameter δ indicates how close the $p_{t_j}^{CM_i}$ value needs to be to the goal probability P_{t_j} for the property to be considered satisfied. This parameter can be selected by ACCORDION users depending on their modeling goals.

Below, we provide notations summary as well as the definition of all the metrics used in to evaluate the performance of ACCORDION

$\mathcal{T}$ – set of all properties

$t_j \in \mathcal{T}$ – property *j*

$p_{t_j}^{\mathcal{M}}$ - estimated probability that model $\mathcal{M}$ satisfies property t_j

P_{t_j} – goal probability value for property t_j

$\varepsilon_{t_j}^{\mathcal{M}} = \left| p_{t_j}^{\mathcal{M}} - \mathrm{P}_{t_j} \right|$ – model error for $\mathcal{M}$ in property t_j **(3.2)**

$\varepsilon_{\mathcal{T},avg}^{\mathcal{M}} = \frac{1}{|\mathcal{T}|} \sum_{t_j \in \mathcal{T}} \varepsilon_{t_j}^{\mathcal{M}}$ – average model error for $\mathcal{M}$ across all properties $t_j \in \mathcal{T}$ **(3.3)**

$\sigma_{\mathcal{T}}^{\mathcal{M}} = 1 - \varepsilon_{\mathcal{T},avg}^{\mathcal{M}}$ – σ-score for $\mathcal{M}$ with respect to a set of properties $\mathcal{T}$ **(3.4)**

δ – limit for $\varepsilon_{t_j}^{\mathcal{M}}$, that is, if $\varepsilon_{t_j}^{\mathcal{M}} \leq \delta$, then it is considered that $\mathcal{M}$ satisfies property t_j; δ can have different values

$\mathcal{D} = \left\{ t_j | \, t_j \in \mathcal{T} \text{ and } \varepsilon_{t_j}^{\mathcal{M}} \leq \delta \right\}$ – $\mathcal{D}$ is a subset of $\mathcal{T}$

$N_{\mathcal{T},\delta}^{\mathcal{M}} = |\mathcal{D}|$ – δ-score for model $\mathcal{M}$ with respect to a set of properties $\mathcal{T}$ **(3.5)**

$N_{\mathcal{T},\delta,\%}^{\mathcal{M}} = \frac{N_{\mathcal{T},\delta}^{\mathcal{M}} * 100}{|\mathcal{T}|}$ – percent of properties within $\mathcal{T}$ that model $\mathcal{M}$ satisfies, assuming δ

(3.6)

$\mathfrak{C}$ - set of all generated CMs

$\tilde{\sigma}_{\mathcal{T}}^{CEM_l} = \left(\sigma_{\mathcal{T}}^{CEM_l} - \min_{CEM_i \in \mathfrak{C}} \left(\sigma_{\mathcal{T}}^{CEM_i} \right) \right) / \left(\max_{CEM_i \in \mathfrak{C}} \left(\sigma_{\mathcal{T}}^{CEM_i} \right) - \min_{CEM_i \in \mathfrak{C}} \left(\sigma_{\mathcal{T}}^{CEM_i} \right) \right)$

– normalized σ-score **(3.7)**

$\tilde{\varepsilon}_{\mathcal{T},avg}^{CEM_l} = \left(\varepsilon_{\mathcal{T},avg}^{CEM_l} - \min_{CEM_i \in \mathfrak{C}} \left(\varepsilon_{\mathcal{T},avg}^{CEM_i} \right) \right) / \left(\max_{CEM_i \in \mathfrak{C}} \left(\varepsilon_{\mathcal{T},avg}^{CEM_i} \right) - \min_{CEM_i \in \mathfrak{C}} \left(\varepsilon_{\mathcal{T},avg}^{CEM_i} \right) \right)$

– normalized average error Equation **(3.8)**

$\tilde{p}_{\mathcal{T}}^{CEM_l} = \left(\prod_{t_j \in \mathcal{T}} p_{t_j}^{CEM_l} - \min_{CEM_i \in \mathfrak{C}} \left(\prod_{t_j \in \mathcal{T}} p_{t_j}^{CEM_i} \right) \right) / \left(\max_{CEM_i \in \mathfrak{C}} \left(\prod_{t_j \in \mathcal{T}} p_{t_j}^{CEM_i} \right) - \min_{CEM_i \in \mathfrak{C}} \left(\prod_{t_j \in \mathcal{T}} p_{t_j}^{CEM_i} \right) \right)$

– normalized joint probability of satisfying all properties in $\mathcal{T}$ (assuming independence in properties. **(3.9)**

3.3 Results

3.3.1 Benchmarking

In the absence of standardized benchmarks to evaluate ACCORDION, we created nine case studies. In Section 2.6, we provide an overview of the biological background for all studied systems, the details of creating the baseline model and the golden model. In this section, we detail the steps of selecting literature and creating CE set for each conducted case study. In Figure 7, we list the main characteristics of these nine cases, with models of three biological systems and different sets of CEs for each system. The three models include control circuitry of naïve T cell differentiation (T cell) (Miskov-Zivanov, Turner, et al., 2013b), T cell large granular lymphocyte (T-LGL) leukemia model (Zhang et al., 2008), and pancreatic cancer cell model (PCC) (C. Telmer et al., 2019). The studies vary in the size and graph features of baseline models ("BM creation" columns) and the CE sets (CE set creation" columns), and are named Tcell CE^{FA}, Tcell CE^{SA}, Tcell CE^{SM}, T-LGL Q^{Sm}, T-LGL Q^{Med}, T-LGL Q^{Det}, PCC BM^{Au}, PCC BM^{Ap}, and PCC BM^{Pr}. As can be seen in Figure 7, the size of baseline models varies from several tens to several hundreds of nodes or edges, and the number of interactions in the CE set varies from half the number of interactions in the baseline model to six times larger ("BM and CE set relationship" columns). We also list in Table 1, Table 2, Table 3, the sets of desired properties, that are not fully satisfied by baseline models and are used to guide new model assembly for each case study. The properties are provided in both natural language descriptions and machine readable BLTL format, and we also include their goal probability values (P_{t_j}).

For each system, besides a baseline model, we also found a golden model in literature ((Hawse et al., 2015b) for the T cell model, (Zhang et al., 2008) for the T-LGL model, and (C. A.

Telmer, Sayed, et al., 2019) for the PCC model). Figure 7 includes the characteristics of golden models (columns "GM" and "GM and CE set relationship"). With these nine case studies, we evaluate ACCORDION's performance and also demonstrate different research scenarios where it can be used, such as varying size and contents of baseline model and CE set (all nine case studies), varying quality of the CE set (Tcell case studies), varying level of detail in user selection of literature (Tcell CE^{FA} and all three T-LGL case studies) reconstruction of previously published model (all nine case studies).

	CE set creation								BM creation				BM and CE set relationship		GM		GM and CE set relationship		
STUDY	query	pre-selected papers	curation	fixed	read papers	$\|E^{RO}\|$	$\|V^{CE}\|$	$\|E^{CE}\|$	target element	fixed	$\|V^{BM}\|$	$\|E^{BM}\|$	$\|V^{BM} \cap V^{CE}\|$	$\|E^{BM} \cap E^{CE}\|$	$\|V^{GM}\|$	$\|E^{GM}\|$	$\|E^{GM} \cap E^{CE}\|$	$\|E^{GM} \setminus E^{BM}\|$	$\|(E^{GM} \setminus E^{BM}) \cap E^{CE}\|$
Tcell CE^{FA}	✓	-	-	-	11	300	156	239	-		39	60	18	0	43	74	10	14	10
Tcell CE^{SA}	-	✓	✓	-	13	188	82	131	-	✓	39	60	20	2	43	74	13	14	11
Tcell CE^{SM}	-	✓	✓	-	12	118	64	85	-		39	60	15	0	43	74	11	14	11
T-LGL Q^{Sm}	✓	-	-	-	22	78	58	52	-		41	113	16	3	60	193	11	80	8
T-LGL Q^{Med}	✓	-	-	-	38	645	303	448	-	✓	41	113	20	6	60	193	20	80	14
T-LGL Q^{Det}	✓	-	-	-	46	907	424	644	-		41	113	26	5	60	193	18	80	13
PCC BM^{Au}	-	✓	-		19	706	451	631	✓	-	232	323	74	8	257	373	12	50	4
PCC BM^{Ap}	-	✓	-	✓	19	706	451	631	✓	-	195	314	73	6	257	373	12	59	6
PCC BM^{Pr}	-	✓	-		19	706	451	631	✓	-	177	286	73	8	257	373	12	87	4

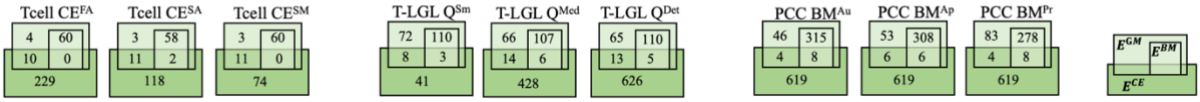


Figure 7 (Top) Description of each use case in terms of 1-how each CE set is acquired (using a query or a preselected set of papers, how many papers are read, the number of edges in the entire machine reading output file, E^{RO}, the number of nodes or entities V^{CE}, the unique number of interactions in each CE set E^{CE}), 2-how each baseline model (BM) is created, whether it is fixed across all three studies for the same system, or it is different for the three studies of the same system, 3-the relationship between each BM and the corresponding CE set in terms of the number of common nodes (entities) and edges (interactions), 4-the golden model (GM) specifications, 5-the relationship between each GM and the corresponding CE set (the number of common edges, the number of edges that are in GM but not in BM, the number of edges that are in GM but not in BM and are found in the CE). (Bottom) Venn diagrams showing the overlap between three sets, E^{CE}, E^{BM} and E^{GM} for the nine case studies.

3.3.1.1 T cell CE Sets

The CE set, which is another input to ACCORDION, is assembled in three different ways for the Tcell case studies. In the *fully automated* (Tcell CE^{FA}) approach, both the PubMed database search for relevant articles and the extraction of event data from the selected articles were done by machines. Specifically, in the FA experiment, we used search query "*T-cell and (PTEN or AKT or FOXO)*" and selected top 11 from the best matched papers, by the PubMed search engine. In the *semi-automated* (Tcell CE^{SA}) approach, we selected papers that are cited by (Hawse et al., 2015a) and used the event information that REACH extracted from those papers. Finally, in the *semi-manual approach* (Tcell CE^{SM}), we rely the most on human intervention, we manually excluded from the SA reading output those interactions that violate any assumptions made by the authors originally in (Miskov-Zivanov, Turner, et al., 2013b). For instance, the authors in (Miskov-Zivanov, Turner, et al., 2013b) consider element TCR to be an input to the network, and therefore, TCR should not have any regulators in the T cell model. Therefore, if REACH retrieves an interaction in which TCR is a regulated element, we manually remove these interactions and keep only the interactions having TCR as a regulator.

3.3.1.2 T-LGL CE Sets

For the T-LGL cases, we came up with three different queries that will be the input to the search engine in order to retrieve the most relevant sets of pa-pers. The first query which corresponds to "T-LGL simple query" or (T-LGL Q^{Sm}) case is "*T-LGL leukemia therapeutic targets and apoptosis*". From the papers that PubMed returned, we then selected 22 papers that PubMed identified as "Best match". The machine reading output obtained by reading those papers contains 52 interactions. For the second and third cases of this study namely "T-LGL medium query" or T-LGL Q^{Med} and "T-LGL detailed query" or T-LGL Q^{Det}, we used "*T cell large granular*

lymphocyte (T-LGL) leukemia proliferation apoptosis" and "*T cell large granular lymphocyte (T-LGL) leukemia therapeutic targets proliferation apoptosis*", respectively. The number of interactions used in these cases are 448 and 644 extracted from 38 and 46 papers, respectively. As can be noticed, the queries were designed so that they include key words related to the T-LGL model and therapeutic targets. For each query and the corresponding set of papers, we used REACH to read these papers and extract the set of CEs as input to ACCORDION.

3.3.1.3 PCC CE Sets

The CE input set for ACCORDION is the same for the three PCC cases, it contains 631 interactions retrieved from 19 papers cited in the PCC model paper (C. A. Telmer, Sayed, et al., 2019).

3.3.2 Recommending New Models with Desired Behavior

In Figure 8, we show the minimum and maximum of the average model error ($\varepsilon_{T,avg}^{CM_i}$) found across all created CMs for each of the nine use cases. Additionally, in Figure 9, we show the δ-score, $N_{\mathcal{T},\delta}^{CM_i}$, values for the top CMs recommended by ACCORDION in all nine use cases. We also explored different δ values (0.1 to 0.5). To highlight the improvements in CMs when compared to the original baseline model, we show all results next to their corresponding baseline model values. As can be seen from the figure, ACCORDION achieved δ-score of 95% when δ = 0.3 (i.e., all but one property satisfied). Furthermore, increasing δ improves the model score, however, we observed that 0.2 or 0.3 value for δ is optimal to obtain useful models with high score. Overall, ACCORDION automatically selected a small fraction (e.g., ~20%, as will be discussed

in Section 3.3.3) of all interactions in the CE set, sufficient to decrease model error by up to 83%, as shown in Figure 10.

As can be concluded from Figure 8, Figure 9, Figure 10, automated reading and model assembly are not able to reduce model errors $\varepsilon_{T,avg}^{CM_i}$ all the way to 0 in our use cases. ACCORDION outputs $p_{t_j}^{CM_i}$ values for all properties and all CMs it creates, and the list of extensions from CEs that are used in each CM. We show in Figure 11 the $p_{t_j}^{CM_i}$ heatmaps that ACCORDION computed for all nine case studies. The heatmaps provide details per each individual property and CM, and this information can be especially useful if users decide to manually inspect and further modify CMs recommended by ACCORDION. Although we show in Figure 11 results for all properties, several of the CE sets did not fulfill the necessary requirement for all properties to be used.

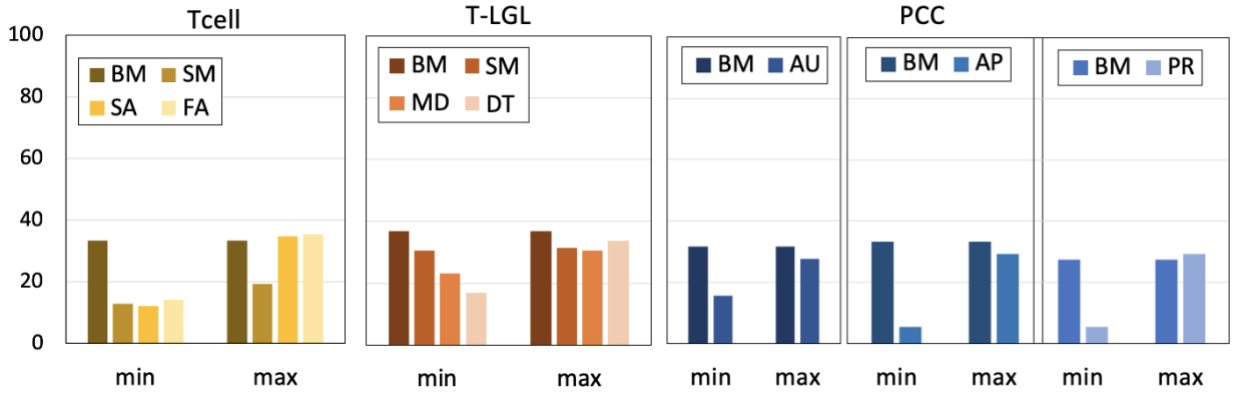


Figure 8 ACCORDION evaluation on nine case studies, three Tcell studies SM, SA, FA, three T-LGL studies SM (Sm), MD (Med), DT (Det), and PCC studies AU, AP, PR: (a) minimum (best) and maximum (worst) average model error $\varepsilon_{T,avg}^{CM_i}$ across all recommended models for each case study, compared to the average error of the baseline model in each study

In other words, all the elements that are listed in properties (Table 1, Table 2, Table 3) need to be present in at least one of the sets V^{BM} and V^{CE}. As shown in Figure 12 ("Properties" columns), in six out of nine studies, these elements are either already in the baseline model or in the CE set. However, in all three T-LGL studies element GAP is not found in either of the two sets, V^{BM} and V^{CE}, and in the T-LGL Q^{Sm} case two elements, Ceramide and SOCS, are also not present. These element omissions occur in ACCORDION's input and are due to machine reading not finding those elements in selected papers. While the properties that correspond to such omitted elements are not suitable for evaluating ACCORDION, we included them in our results to demonstrate realistic cases with imperfect CE sets. As part of our future work on ACCORDION, we plan to include pre-processing methods to automatically exclude such tests before clustering the CE set, or to inform the user at the beginning that property elements are not found in the input.

On the other hand, we were especially interested in ACCORDION's performance in the cases where property elements are not present in V^{BM} but are in V^{CE}. Thus, we defined "criterionA" (Figure 12) to evaluate ACCORDION in such cases. As can be seen from the figure, ACCORDION is able to recover all property elements missing from a baseline model in at least one of the recommended CMs.

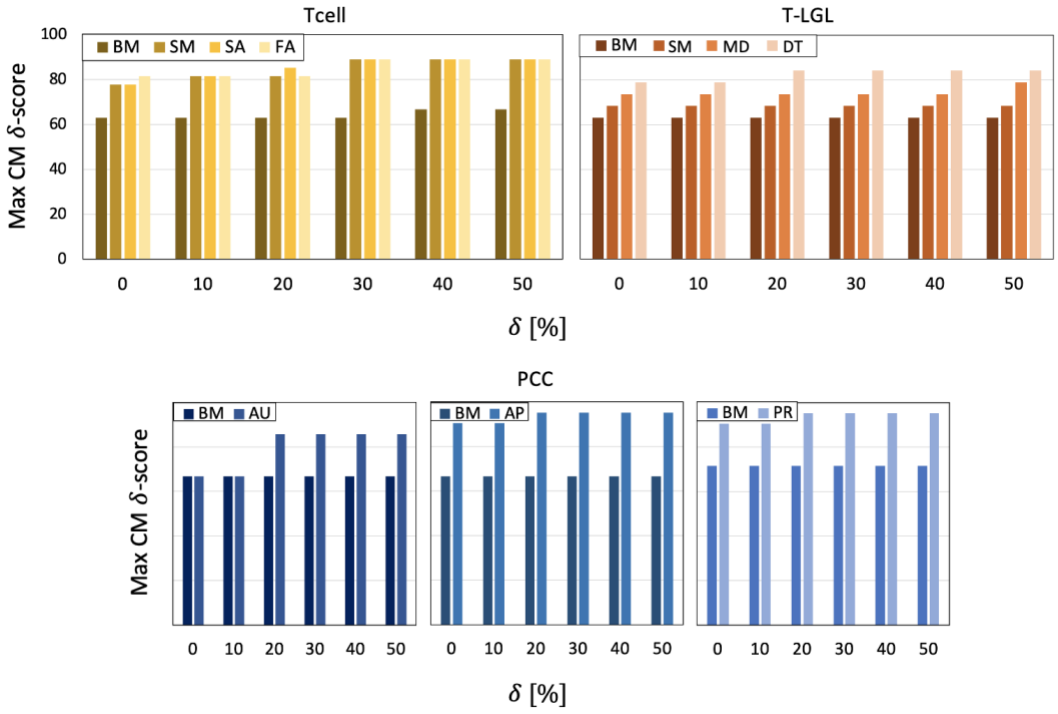


Figure 9 ACCORDION evaluation on nine case studies, three Tcell studies SM, SA, FA, three T-LGL studies SM (Sm), MD (Med), DT (Det), and PCC studies AU, AP, PR: maximum across all CM δ-scores $N_{\mathcal{T},\delta}^{CM_i}$ obtained in each case study expressed in %; the results are compared for different values of δ (10%, 20%, 30%, 40%, 50%)

Finally, when ACCORDION recovers all necessary property elements, most often the reason for non-zero model property errors ($\varepsilon_{t_j}^{CM_i} > 0$) is in update rules. For instance, in the Tcell cases, for the best recommended model per case, ACCORDION was able to recover FOXO1 which was not in V^{BM} but was in V^{CE}. Moreover, ACCORDION recovered the update function of FOXO1 in all three cases and therefore, the properties that correspond to the dynamic behavior of FOXO1($\mathcal{t}9$, $\mathcal{t}18$ and $\mathcal{t}27$) under three different scenarios were all satisfied as shown in Figure 11. However, in the case of update function for AKT, ACCORDION added a number of new AKT regulators to the baseline model which affected the dynamic behavior of AKT. Again, this demonstrates the dependance of ACCORDION output on the CE sets provided by machine

reading. There are two ways in which this could be overcome. First, one could either use other tools to filter or score individual interactions in CE set (Gyori et al., 2017b) before they are used by ACCORDION, which we are planning to incorporate as one of our next steps. Second, ACCORDION can be used to identify cases where human input is necessary, for example, cases where many element regulators appear in literature, not all of which can be used to form regulatory rules.

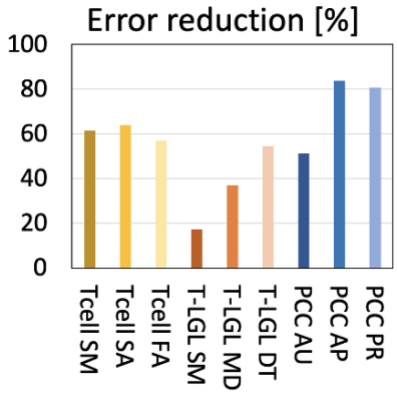


Figure 10 Error reducttion ACCORDION achieves in each case study

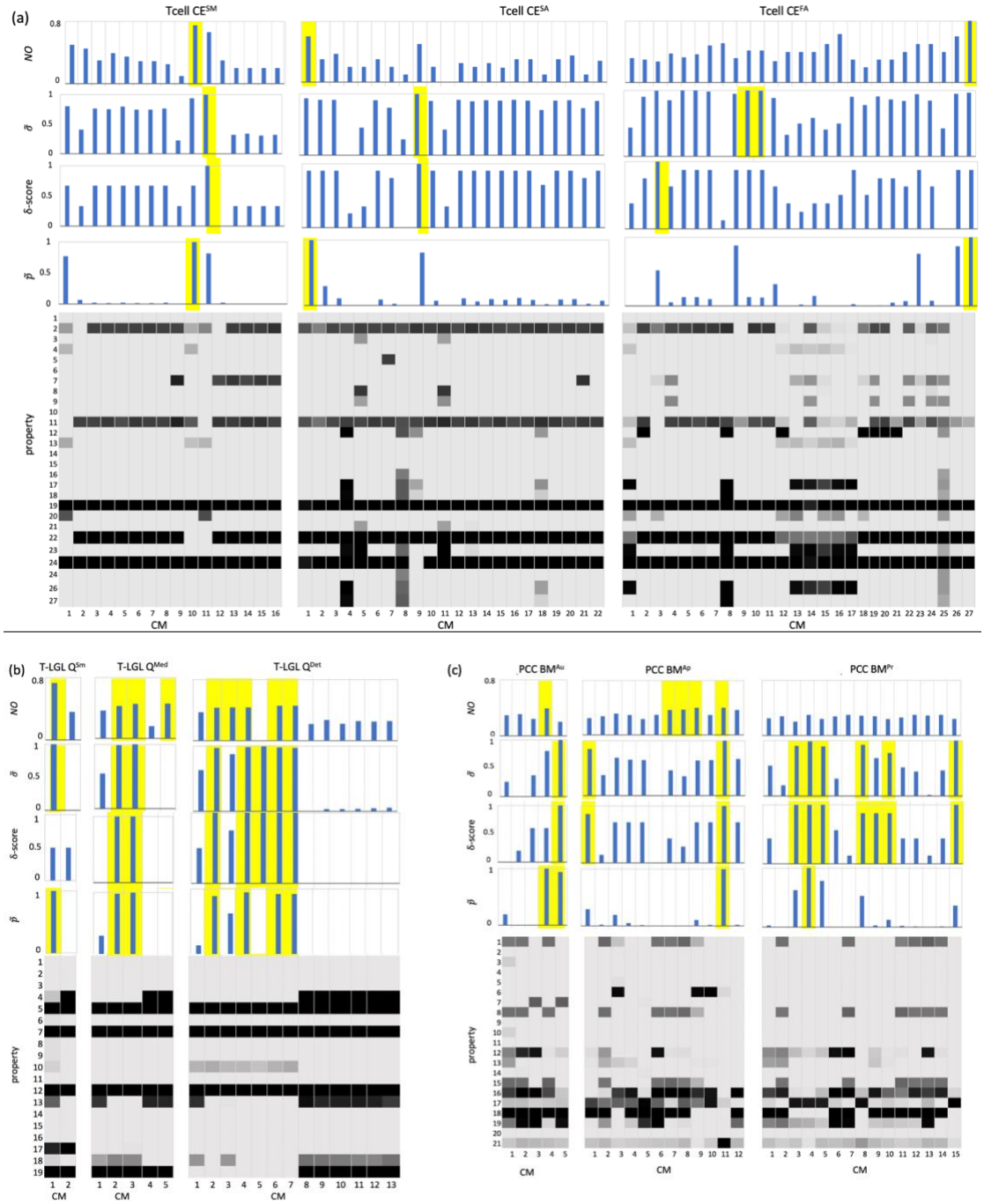


Figure 11 The Tcell, T-LGL and PCC use cases results including 1- the node overlap (*NO*) values for each candidate model (CM), 2- the normalizedsigma-score ($\tilde{\sigma}$) , 3- the normalized delta-score (δ-score), 4- the normalized joint probability ($\tilde{p}$), and 5- heatmaps of the statistical model checking results of 27, 19 and 21 properties of the Tcell, T-LGL and PCC use cases, respectively, listed in Table 1, 2 and 3.

3.3.3 Finding Most Relevant Set of New Interactions

We created the use cases such that the relationship between the number of elements and interactions in baseline models ($|V^{BM}|$, $|E^{BM}|$), and in their corresponding CE sets ($|V^{CE}|$, $|E^{CE}|$) varies, from the CE set being smaller than baseline model in the T-LGL Q^{Sm} case, to being up to six times larger than baseline model in other use cases (Figure 7). We also determined the size of the overlap, $|V^{BM} \cap V^{CE}|$ (see Figure 7), further highlighting that indeed the number of new elements that could be added to the model is much larger than the number of elements in the model. Additionally, we created these nine case studies such that they have baseline models with varying level of network connectivity. As described in Section 2.6.1.1, the baseline model in the T cell studies is a previously published, thus functional model, while the T-LGL and PCC baseline models were created by removing nodes and interactions from a published model. Since by construction the clusters that ACCORDION generates are usually connected only to a part of the baseline model, we used the node overlap metric *NO*, defined in Section 5.2.4, to determine the relationship between the number of new nodes that are added to the baseline model and the part of the model those nodes are connected to. The *NO* numbers in Figure 11, together with the ratios $\frac{|E^{CM} \setminus E^{BM}|}{|E^{CE}|}$ listed in Figure 12, show that ACCORDION is very selective, and it only adds to the baseline model a subset of new interactions that are well connected with the model.

We further investigated the percentage of these interactions selected from the entire CE set that were included in the top recommended CM (Figure 12(a)). For the Tcell cases, ACCORDION recommended on average 14% of the interactions as candidates for model extension, whereas for T-LGL and PCC cases, ACCORDION identified on average 26% and 15% of such interactions, respectively. These numbers emphasize an important characteristic of ACCORDION: while it

provides a comprehensive overview of literature, it significantly reduces the number of selected interactions, such that, if human input is still necessary, the number of interactions to manually review is significantly smaller than the original CE set. Interestingly, when observed together with the model error results, in the T cell and T-LGL studies, the higher *NO* values seem to correlate well with larger reduction in model error. However, in the PCC studies this correlation does not hold, where the CMs with a large number of new interactions compared to the size of the baseline model significantly decrease the baseline model error (~80% reduction). This demonstrates another important characteristic of ACCORDION: when the baseline model is already well-built, a smaller number of extensions can help improve it (e.g., Tcell and T-LGL cases), while for baseline models that are not very well connected and not functional or usable to start with (e.g., when the user starts only with a seed set of interactions and not a complete model), a larger number of interactions needs to be added to improve them (e.g., PCC case).

3.3.4 Identifying Alternative Networks

As described in Section 2.6, we identified golden models for our case studies. Our goal with using golden models was twofold: we were interested in exploring how closely ACCORDION can reproduce previously published models ("criterion B" and "criterion C" in Figure 12) as well as comparing and contrasting them to automatically created models that satisfy the same set of properties. In all three T cell case studies, ACCORDION adds all the interactions from the $E^{GM} \backslash E^{BM}$ set to its top recommended CMs (columns "GM" in Figure 12, dark yellow cells). For example, the merged cluster $C_{1,2}^{SM}$, with *NO*=0.7, restored all the missing interactions that were removed from the golden model. In the T-LGL and PCC studies, ACCORDION adds 30% and 32% of missing golden model interactions to recommended CMs. However, while in all

three T cell studies all missing golden model interactions, i.e., interactions from the $E^{GM} \backslash E^{BM}$ set are present in CE sets, the CE sets in the T-LGL and PCC studies do not contain all the interactions from the $E^{GM} \backslash E^{BM}$ sets, as shown in Figure 12 (columns "GM", dark yellow cells). This is due to either papers that were selected using queries do not include those missing interactions or machine reading does not recognize these interactions in the papers.

An important outcome from this exercise is that ACCORDION recommends new CMs, different from golden models, which have high σ-score and δ-score and contain new interactions that form return paths with the baseline model. Moreover, in the T-LGL studies, a significant portion of interactions (41%) was removed from the golden model to obtain the baseline model. In such cases, ACCORDION selected from the large CE sets many additional interactions that form stronger connections with the baseline model (as part of clusters with high *NO* values and return paths) than the ones that are in the golden model, while also being able to find CMs that have high σ-score and δ-score. For instance, the regulators of AKT in the golden model are PIP3 and mTORC2, while the models recommended by ACCORDION also include regulations by TGFB, IFNgamma, CK2, CTLA4, SHIP1, all of which are suggested in literature. This highlights another possible use of ACCORDION, when examining redundancies in signaling networks or discovering alternative pathways regulating the same target element.

3.3.5 Assistance in Query Answering

We also explored the relationship between the design of queries and ACCORDION's effectiveness, that is, whether the selection of search terms to mine literature affects the usefulness of extensions selected by ACCORDION. As described in Section 3.3.1.1, for the Tcell CE^{FA} case, we used a search query as an input to PubMed to identify the most relevant papers. We investigated

the influence of this query on the percentage of interaction in clusters used to create CMs with top scores. In Figure 12, we show the average and the maximum percentage of selected interactions, i.e., $\left(\frac{|E^{CM} \setminus E^{BM}|}{|E^{CE}|}\right)_{avg}$ and $\left(\frac{|E^{CM} \setminus E^{BM}|}{|E^{CE}|}\right)_{max}$, which are 10% and 33%, respectively. For the best recommended model of this particular case study, ACCORDION was able to recover all the missing elements that are in V^{GM} and not in V^{BM}, namely, FOXO1, NEDD4, CK2 and MEK1. Furthermore, as can be seen in Figure 11, ACCORDION recapitulated the dynamic behavior of FOXO1, an element that was in the search query used to collect interactions for the CE set (Section 3.3.1.1), in all three scenarios (properties $t9$, $t18$and $t27$). However, the dynamic behavior of AKT (also in the search query), IL2 and STAT5 was not recovered in one out of three scenarios, (high TCR scenario, properties $t19$, $t22$ and $t24$). This is due to potentially erroneous interactions in the CE set extracted by machine readers, e.g., CD8 → AKT, proliferation → AKT, differentiation -| AKT, differentiation -| IL2 and differentiation -|STAT5 ("→" represents positive regulation, "-|" represents negative regulation, also used in Figure 1, Figure 4). As mentioned above, we plan to add pre-processing of CE sets (e.g., using interaction filtering (Holtzapple et al., 2020)). For the T-LGL model study, we used three different queries as described in Section 3.3.1.2. The most elaborate query, in the T-LGL Q^{Det} case study, introduced more descriptive search terms, led to selecting more relevant papers, and consequently, extraction of relevant events and element regulators resulting in recommendation of a CM with high σ-score (0.76) and δ-score (0.75). Additionally, the update rules of most of the elements were retrieved except three elements, S1P, GAP and IL2RB. The properties that correspond to these three elements are properties $t5$, $t7$ and $t12$. In contrast, for T-LGL Q^{Sm} and T-LGL Q^{Med} cases, less properties have been satisfied.

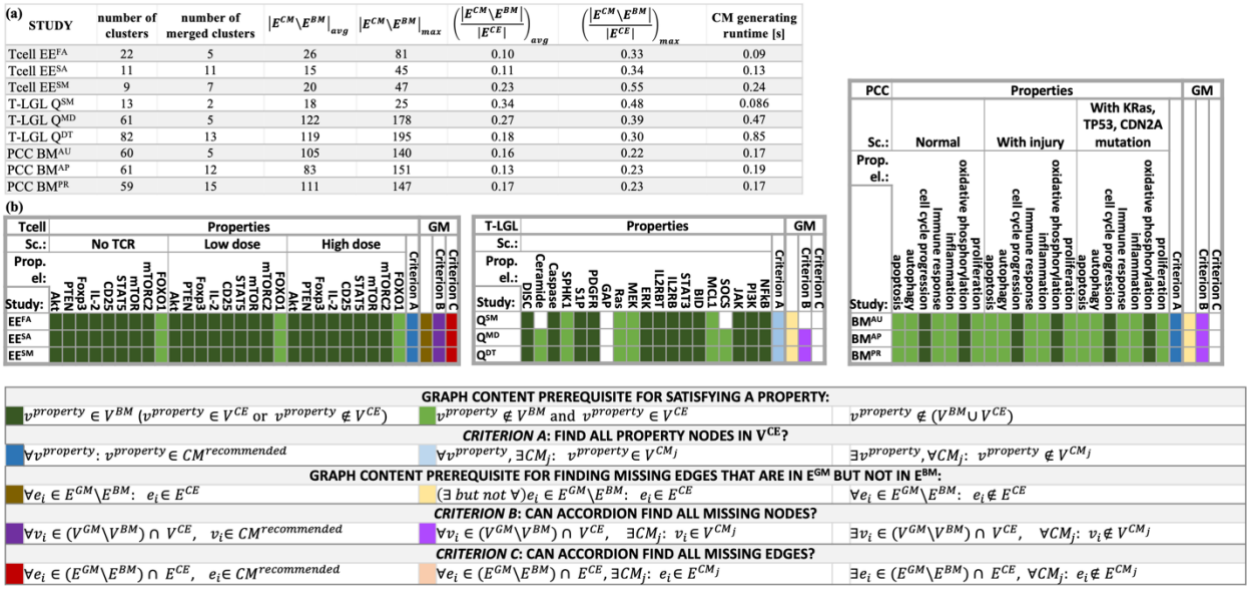


Figure 12 (a) Characterization of CMs created by ACCORDION for the nine case studies. (b) Three criteria definitions and ACCORDION's criteria outcomes in the nine case studies ($v^{property}$ is the element included in the property as listed in Table 1, 2 and 3; $CM^{recommended}$ is the top recommended model).

For example, the baseline model error in property $t17$, related to the behavior of element JAK, is not corrected in the T-LGL Q^{Sm} case, while property $t19$, related to element NFκB, is not corrected in both T-LGL Q^{Sm} and T-LGL Q^{Med} cases. This is mainly due to the key regulatory interactions for these elements not being extracted from literature, or due to the interactions that are recovered not forming proper update functions. Overall, by comparing the results for the three queries in the T-LGL case studies, we have confirmed that a better query design leads to more useful and relevant information in the input CE sets.

3.3.6 Role of Static Network Characteristics

In this section, we will explore some of the static network characteristics of the network to be clustered or G^{new}, for the three cases of the Tcell benchmark. In Figure 13, we show three

networks (undirected interaction maps, for easier visualization) that were formed by combining each of the CE sets (Tcell CE^{FA}, Tcell CE^{SA}, and Tcell CE^{SM}) with the baseline model.

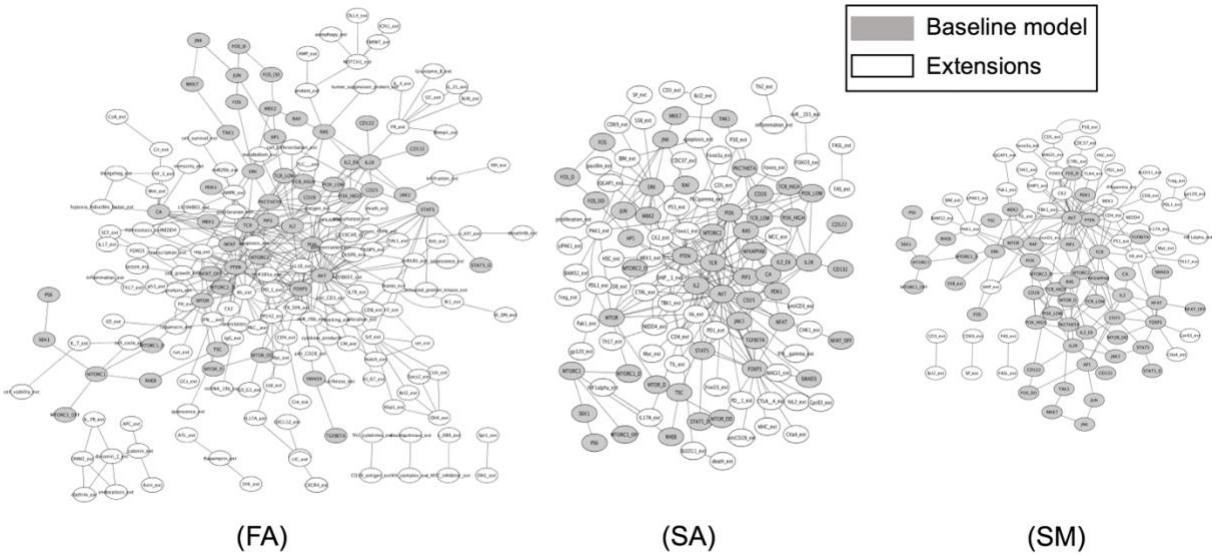


Figure 13 Networks obtained when combining baseline model with the CEI set for each of the three cases, FA, SA, and SM ($G^{new,FA}$, $G^{new,SA}$, and $G^{new,SM}$). Gray nodes are the baseline model nodes and white nodes are the new nodes that belong to the CEIs.

We explored static characteristics of these three graphs and their correlation with the selection of the best extended model. Given a directed graph $G^{new,*}(V^{new,*}, E^{new,*})$, where $*\in\{FA,SA,SM\}$ and the definition of a path in Section 2.2, we computed average path length (APL), clustering coefficient (Coeff), and graph density (Section 2.4). In Figure 14, we highlight the difference between the three graphs: $G^{new,FA}$ is shown in green, $G^{new,SA}$ in blue, and $G^{new,SM}$, which is a subgraph of $G^{new,SA}$, in orange. In addition, we show the overlapping nodes between the three networks in cyan.

Interestingly, it was observed that despite network diversity, $G^{new,FA}$, $G^{new,SA}$, and $G^{new,SM}$ share prominent structural features: they have small *APL*, small average *Coeff*, and small graph density, and thus, large average degree values are unlikely. This similarity is even better illustrated in (Figure 15(a)), showing the Degree histogram for the nodes in each network that follows a

power law, and in (Figure 15(b)) showing the distribution of node distance (*d*) centered approximately around value 4. As can be noticed, both graph parameters, Degree (*D*) and *d*, have similar patterns but with different count numbers for each $G^{new,*}$ in proportion to the size of its network.

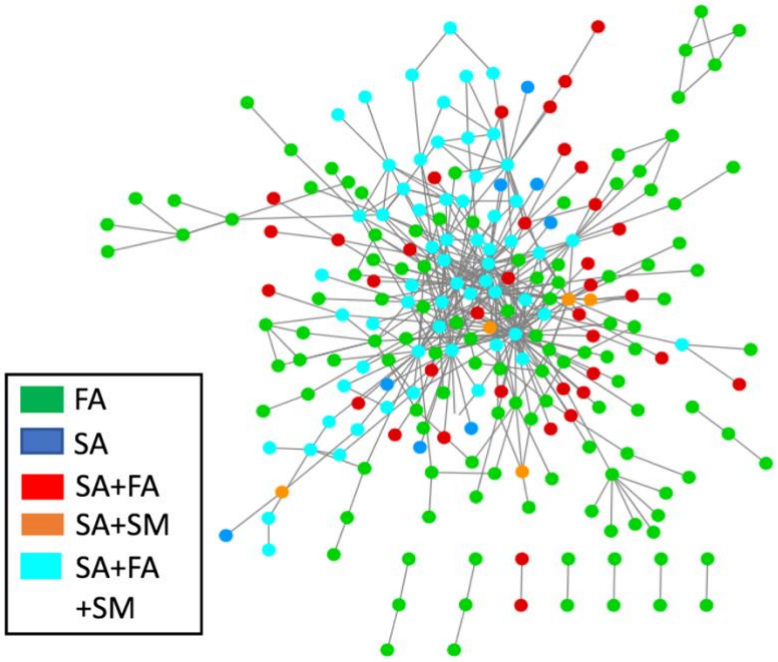


Figure 14 $G^{new,FA}$, $G^{new,SA}$, and $G^{new,SM}$ drawn together, highlighting the common nodes

Moreover, the values of graph density of G^{new} for Tcell CE^{FA}, Tcell CE^{SA}, and Tcell CE^{SM} cases are 0.017, 0.032 and 0.032, respectively, which suggest that the graphs assembled from the information extracted by machine readers are less dense, even with varying network size. These results also suggest that the difference in literature sources and the size of the CE sets did not affect the characteristics of $G^{new,*}$ graphs in this case study.

The inspection of obtained clusters shows that they are less dense and star-like networks (two examples shown in Figure 16, which agrees with the conclusions of the above studies of graph characteristics. Thus, computing the graph parameters can guide our proposed extension method by providing an early estimate of whether the CE sets can lead to desired models. For

instance, if the *APL* is large, we will expect to extract a fewer number of return paths from the $G^{new,*}$ graphs, and therefore, in our analysis we will lack the connectivity of the CEs to the baseline model. Additionally, the graphs with smaller graph density will reduce the computation time, and computing this parameter helps determine in advance the expected execution time of our algorithm.

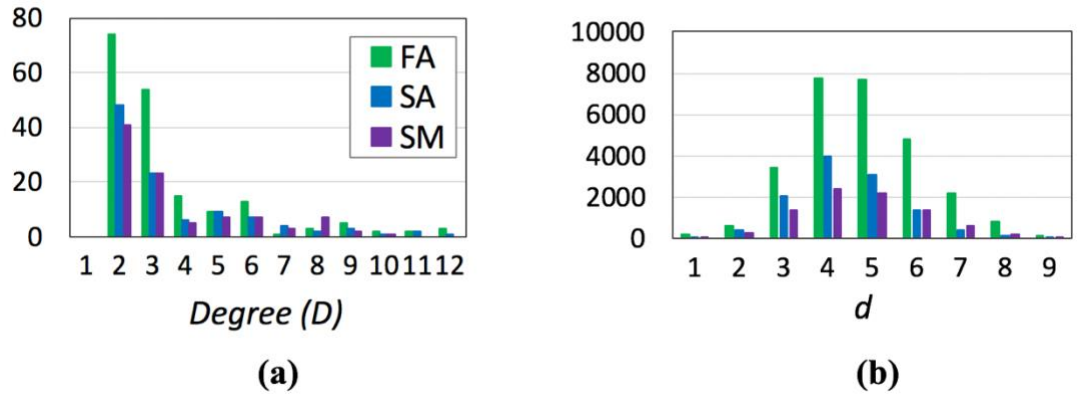


Figure 15 (a) Different degree *D* values and corresponding number of nodes; (b) Different shortest path length *d* values and the corresponding number of paths

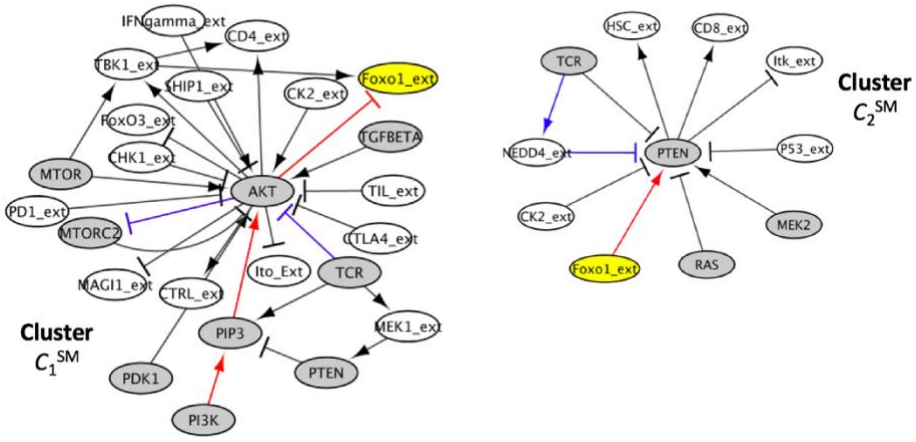


Figure 16 Two clusters that form a return path with the baseline T cell model, shown as directed graphs (yellow node is a common node for both clusters).

3.3.7 Runtime

In Figure 12, we list the time that ACCORDION takes to generate clusters when run on a 3.3 GHz Intel Core i5 processor. The time required by ACCORDION to generate clusters increases with larger CE sets. For the PCC case studies, the runtime same across studies since the same CE set has been used. However, for the T cell and T-LGL case studies, the CE sets have different sizes, and thus, result in different runtime. The runtime of the overall extension algorithm is proportional to the number of properties that we need to test against. In other words, if we have N_C clusters and N_P properties, the time required for the extension algorithm is at the order of $O(N_C * N_P)$. However, the runtime can be significantly reduced if testing for all properties and clusters is carried out in parallel, which is part of our immediate future work.

3.4 Conclusion

In this chapter, we have described a novel methodology and a tool, ACCORDION, that can be used to automatically assemble the information extracted from literature into models and to recommend models that achieve desired dynamic behavior. Our proposed approach combines machine reading with clustering, simulation, and model checking, into an automated framework for rapid model assembly and testing to address biological questions. Furthermore, by automatically extending models with the information published in literature, our methodology allows for efficient collection of the existing information in a consistent and comprehensive way, while also facilitating information reuse and data reproducibility, and often helping replace tedious trial-and-error manual experimentation, thereby increasing the pace of knowledge advancement.

The results we presented here demonstrate different research scenarios where ACCORDION can be used. As our next steps, we are planning to improve the input pre-processing in order to provide more useful candidate event sets, to make ACCORDION compatible with other model representation formats (e.g., SBML), as well as to work on parallelizing the tool implementation to improve the runtime when testing large number of properties.

4.0 Guided Assembly of Network Models from Knowledge in Literature using Collaboration Graph

4.1 Objective and Applicability

We propose a novel methodology, MELOGRAPH (Model Extension using CoLlabOration GRAPH) (Ahmed & Miskov-Zivanov, 2021), to automatically assemble network mechanistic models, by selecting most relevant and useful information from published literature. This is achieved by identifying the most influential events in the newly extracted information, and then scoring these events using the occurrence frequency of events and graph centrality metrics. The proposed methodology examines events extracted from literature in the context of a collaboration graph and the measure of the occurrence frequency in literature. MELOGRAPH also explores the role of several graph centrality metrics in identifying the most influential events. Using MELOGRAPH, we propose a heuristic to determine which centrality metrics are crucial for finding those influential events. Our methodology takes at most a few seconds to execute thousands of in silico experiments, which would otherwise take months, or would be impractical, to conduct in a wet lab. We evaluate our method of model assembly using three benchmark models, namely, Tcell CE^{SM}, TLGL Q^{Med} and PCC BM^{Au} (Section 3.3.1). The main contributions of MEOGRAPH are: 1) application of the concept of a collaboration graph in guiding model extension; 2) a metric for event ranking based on their occurrence frequency in literature; 3) a method to evaluate the importance of graph centrality characteristics in finding influential events; 4) application of the proposed methods on several case studies in biology.

4.2 Proposed Methodology

In this section, we describe the main steps of our proposed methodology, which are also outlined in Figure 17.

4.2.1 ECG Creation

Following the notion of a collaboration graph that is often used to model social networks (Grossman & Ion, 1995), we introduce the *Event Collaboration Graph* (*ECLG*). In the social network domain, nodes represent participants and edges connect two nodes whenever there is a collaborative relationship between them. Similarly, we define the ECLG as an undirected graph $\mathcal{G}(\mathcal{E}, \mathcal{C})$, where $\mathcal{E}$ is a set of graph nodes, each representing a distinct event e in CE set, $\mathcal{C}$ is a set of undirected graph edges, each edge $c(e_i,e_j)$ indicating a co-occurrence in the same paper of its adjacent nodes, e_i and e_j (i.e., the two events represented by these nodes).

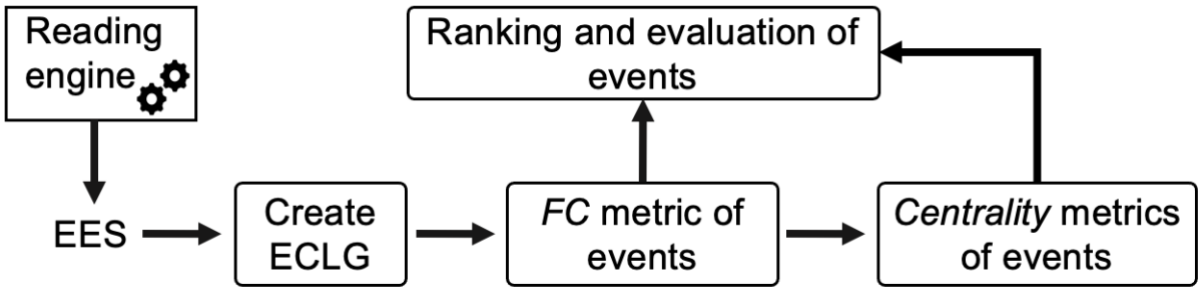


Figure 17 MELOGRAPH methodology workflow

4.2.2 Frequency Class Metric and Centrality Metrics

To measure the frequency of occurrence within CE set of individual distinct events that belong to an ECLG, we propose to use a computational linguistic concept for calculating word frequency, called Häufigkeitsklasse or *frequency class* (*FC*) (P.Brown et al., 1992; Weeber et al., 2000) Here, given the CE set (with *n* total events and *m* distinct events), we compute the frequency class value, FC_i, for each extracted distinct event e_i, where i=1,..,m and $\lfloor .. \rfloor$ is the floor function:

$$\boldsymbol{FC_i = \left\lfloor 0.5 - log_2 \frac{f_i}{f_{max}} \right\rfloor} \quad \textbf{(4.1)}$$

We denote the frequency of each distinct event e_i, that is, the overall number of occurrences of event e_i within CE set, as f_i. We also identify all distinct events for which $f_{max} = \max(\{f_i: i=1,..,m\})$. As can be concluded from the equation above, the most frequent event, that is, any event e_i for which $f_i=f_{max}$, will have $FC_i = 0$, while any event half as frequent as the most frequent event will have $FC_i = 1$ (due to logarithm with base 2).

It is worth mentioning that unlike simple naïve event count, the frequency class-based metric helps group the events within a CE set into several categories. This will allow modelers to consider the events within or across these categories. Additionally, setting a threshold based on a simple event count sounds arbitrary and does not account for the occurrence frequency of the other events. In contrast, the frequency class-based metric is computed for each event with respect to the most frequent event, which helps modelers to choose a threshold (as will be discussed later in this Chapter) and discard the less frequent events.

A number of centrality metrics have been introduced to identify and rank most influential or central nodes in large networks (Das et al., 2018; Koschützki & Schreiber, 2008). For any given network, the selection of most suitable centrality metrics is affected by the network's topology.

Here, we are interested in exploring the correlation between several centrality metrics (defined in Section 2.5), namely, Degree (*D*), Neighborhood connectivity (*NC*), Betweenness centrality (*BC*), Closeness centrality (*CC*) and Radiality (*R*), and our proposed frequency class metric.

4.2.3 Relationship between Centrality and FC metrics

To determine in an automated way which centrality metrics are most correlated with the FC metric for a given network, we use the *permutation feature importance* (PFI), a machine learning technique described in (A. Fisher et al., 2018). A typical supervised machine learning problem is composed of: (*i*) a data set $\mathfrak{D}$, (*ii*) a set of features Φ, and (*iii*) a corresponding target class $\mathcal{T}$. For the application of the PFI algorithm in this work, the nodes of the ECLG form the data set $\mathfrak{D}$, the centrality metric values D, NC, BC, CC, and R form the feature set Φ, and the FC metric values previously determined for the ECLG nodes are used as the target class value set $\mathcal{T}$. Next, we use the PFI algorithm to determine which feature (i.e., which centrality metric) in Φ contributed the most to the target class value (FC) in $\mathcal{T}$ of each data point in $\mathfrak{D}$ (node in the ECLG). The class of each data point in $\mathfrak{D}$ in a supervised machine learning problem is obtained using a trained classifier, and we use here the *k*-nearest neighbor (KNN) classifier (Tan et al., 2007). KNN is considered one of the top ten most effective data mining algorithms for their ability to generate simple but powerful classifiers (Wu et al., 2008).

The details of the PFI algorithm are the following. For the given data $\mathfrak{D}$ (i.e., the ECLG nodes) and each feature (i.e., centrality metric), we determine the corresponding feature vector, i.e., a vector of values for the given feature in each data point. The PFI algorithm then conducts multiple iterations, in each iteration randomly shuffling one feature vector to obtain a corrupted version of

data $\mathfrak{D}$. For a given feature $\varphi \in \Phi$, and an iteration $l = 1..L$, the algorithm computes a score $s_{\varphi,l}$, which is used to indicate the accuracy of the classifier (how closely it matches the target class). The *importance score* p_{φ} is then computed for each feature φ using the following equation:

$$\boldsymbol{p_{\varphi} = s_{\varphi,0} - \frac{1}{L}\sum_{l=1}^{L} s_{\varphi,l}} \quad \textbf{(4.2)}$$

where $s_{\varphi,0}$ is computed at the beginning of the algorithm, before any shuffling. The PFI algorithm provides as output the importance score of each feature (centrality metric), thus quantifying the contribution of these features to the given classification of the ECLG.

4.2.4 Selection of Candidate Extension Events

The events selected either using the FC metric or the centrality metrics are considered potential candidates for model extension or assembly and are selected as follows.

We rank all the events in the ECLG (i.e., in set $\mathcal{E}$) in ascending order of FC values, i.e., from the most to the least frequent event. Next, we determine a threshold FC value. This threshold can be determined in different ways, for example, it can be provided as a fixed input parameter, or it can be determined based on the used CE set. As we will discuss later in this chapter, for our case studies we consider the threshold to be an average FC value, FC_{avg}, computed across all nodes (events) in set $\mathcal{E}$. We then create a new set $\mathcal{E}^{FC}$, a subset of $\mathcal{E}$, including all events from $\mathcal{E}$ with $FC \leq FC_{avg}$. We refer to the events in $\mathcal{E}^{FC}$ as *FC candidate events*. In other words, we remove less frequent events from the original ECLG to form a smaller graph $\mathcal{G}^{FC}$ ($\mathcal{E}^{FC}$, $\mathcal{C}^{FC}$). This step will effectively remove edges from the original set $\mathcal{C}$, thus making $\mathcal{C}^{FC}$ a subset of $\mathcal{C}$.

We also rank all events in the original set $\mathcal{E}$, based on the values of the node centrality metric with the highest p_φ, as selected by the PFI algorithm. Next, we choose the cut-off threshold for the centrality metric in order to select the most central nodes. We apply the threshold to determine a subset of $\mathcal{E}$, a new set $\mathcal{E}^{Central}$, that includes the most influential nodes, i.e., top ranked nodes according to the selected centrality metric. We refer to events in $\mathcal{E}^{Central}$, as *centrality candidate events*.

4.3 Results

We conducted several experiments using the three benchmark models described in Section 2.6. We explored how well MELOGRAPH performs in various scenarios, small vs. large model and controlled vs. query-based CE set.

4.3.1 *FC* Candidate Events in three Case Studies

For the Tcell CE^{SM}, TLGL Q^{Med} and PCC BM^{Au} cases (Section 3.3.1), we create an ECLG for each case. To identify the FC candidate events, we compute the FC value for all nodes (events) in the ECLG, according to FC equation. As described earlier in this chapter, events with $FC = 0$ are the most frequent ones and are considered to be strongly supported by literature, where multiple statements include them. We found that in all three case studies $FC_{avg} = 2$, and therefore, we will use this value as a threshold for removing less frequent events (i.e., all events with $FC > 2$) from the ECLG in each case study. We list in Table 4 the number of nodes and edges in the ECLG of each case study, both before ($ECLG^{original}$) and after ($ECLG^{FC}$) the removal of less frequent nodes,

that is, the size of sets $\mathcal{E}$ and $\mathcal{C}$, and sets $\mathcal{E}^{FC}$ and $\mathcal{C}^{FC}$, respectively. We also show in Table 4 the centrality metric values for all ECLGs. To further compare and contrast the two versions of ECLG in each case study, $ECLG^{original}$ and $ECLG^{FC}$, we also show in Table 4values for other commonly used graph metrics. As a reminder, the nodes in $ECLG^{FC}$ represent FC candidate events. As can be noticed in Table 4, not only there is a difference in size between sets $\mathcal{E}$ and $\mathcal{E}^{FC}$, and sets $\mathcal{C}$ and $\mathcal{C}^{FC}$, but also other graph parameters changed. For instance, the change in the average neighborhood connectivity value NC_{avg} ranges from 2% for the T cell use case to 14% in the PCC use case. The distribution of FC values within the CE set of each case study is illustrated with pie charts in Figure 18(Top). When the percentages in each FC category are averaged across the three case studies, the distribution of events with FC=(0, 1, 2, 3, 4, 5) is (7%, 31.3%, 33.6%, 16.3%, 9.6%, 2.3%), respectively. As expected, consistent across all three case studies, the number of distinct events that occur most frequently in literature (FC=0) is small compared to other categories. Interestingly, in all three case studies, the number of distinct events that do not have many occurrences in literature (FC=4 or FC=5) is also relatively small, while more than half of the total number of distinct events is in the higher occurrence frequency categories (FC = 2 or FC = 1).

4.3.2 Evaluating Centrality of FC Candidate Events

We investigated the relationship between graph centrality metrics (described in Section 2.5) and our proposed FC metric. To compare these metrics, we used the golden model and the corresponding CE set from each case study. As the values in Table 4 suggest, for each case study, the removal of less frequent nodes leads to a denser graph, with strongly connected components, which is in agreement with both the increased NC_{avg} value of the nodes and the high clustering

coefficient. Moreover, the average node degree D_{avg} in ECLG increased after the removal of less frequent nodes. This is due to the high (inverse) correlation between the D value and the event FC value, as shown in Figure 18(Middle). Furthermore, the less frequent events (with $FC > 2$), have the lowest D values. The most frequent events in literature, which are at the same time the nodes with higher D values, tend to have a greater ability to influence other ECLG nodes. The correlation coefficients of different metrics are illustrated in Figure 18(Bottom). The nodes having high D values also have FC=0. The strong correlation between FC and D values is also highlighted in the D distribution histogram for $ECLG^{original}$ and $ECLG^{FC}$ in Figure 19(Top).

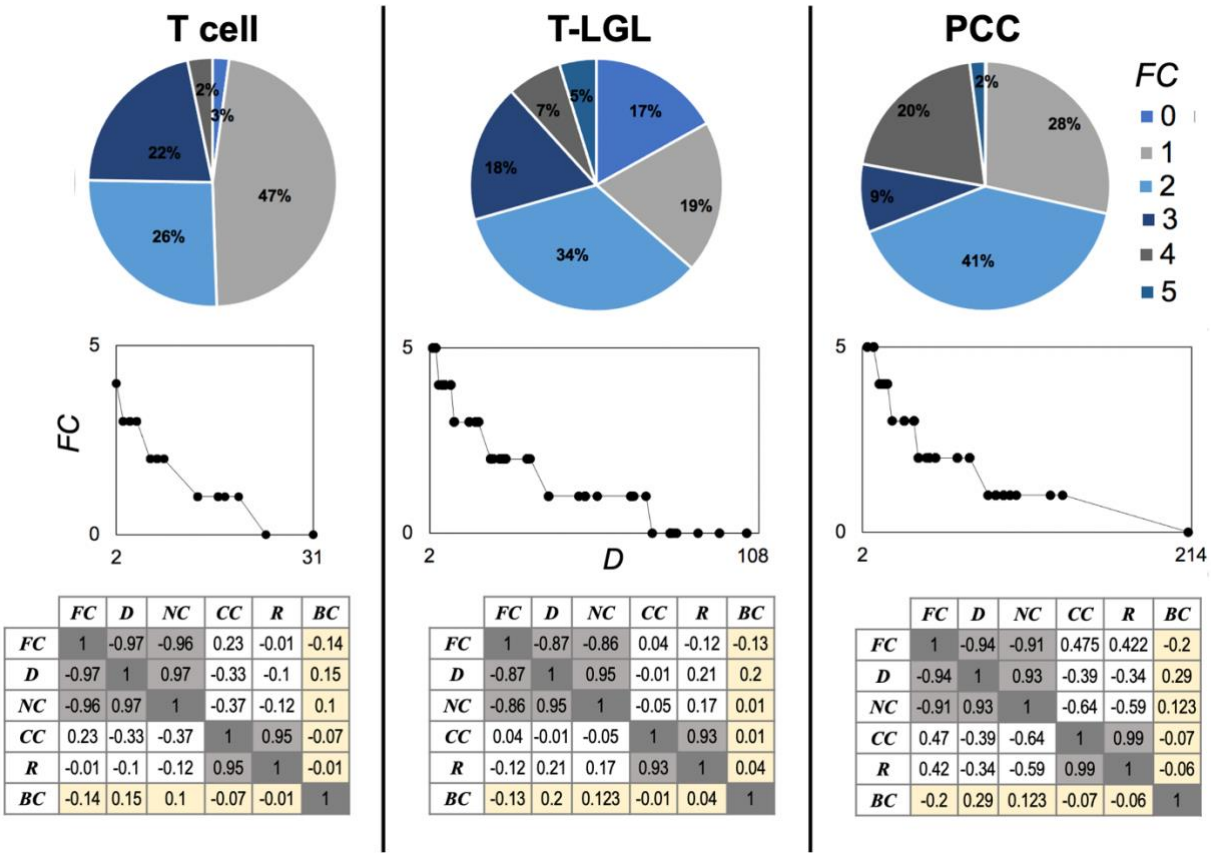
On the other hand, the closeness centrality CC values and the radiality R values of nodes do not seem to correlate with FC values. The R and CC values are highly correlated (Scardoni & Lau, 2012), that is, larger R and CC values indicate central position of a node in a graph, and this is also clearly seen in Figure 18(Bottom). Interestingly, when we plotted R vs. CC of $ECLG^{original}$ and $ECLG^{FC}$, we found that in $ECLG^{FC}$ the relationship between the R and CC values takes almost a linear shape. The removal of the less frequent nodes took out the main outliers that existed in $ECLG^{original}$, while the R and CC values of the remaining nodes did not change much. The reason behind small changes in the remaining nodes is that some of the removed nodes were already separated from the main connected component of the $ECLG^{original}$ graph. Another centrality measure that is not correlated with the literature occurrence frequency is betweenness centrality BC. As listed in Table 4, the average BC value is approximately 0.01, both before and after the removal of less frequent nodes (i.e., in $ECLG^{original}$ and $ECLG^{FC}$, respectively).

As can be seen in the table in Figure 18(Bottom), there is high correlation between NC and FC values in ECLG nodes, which is also confirmed by the strong correlation between D and NC values, on one side, and the D and FC values, on the other. This is further confirmed using PFI to

identify the centrality metric that contributed the most to classifying the CE set – Figure 20 shows the importance score p_{φ} of each centrality metric for all three case studies. We note that degree metric has the highest importance score, for all the case studies, the neighborhood connectivity centrality metric has the second highest and a non-zero importance score, whereas all the other centrality metrics have zero importance score.

Table 4 Summary of the baseline model graph measures and ECLG metric values before and after the removal of less frequent events (ECLG^{original} and ECLG^{FC}, respectively) for T cell, TLGL, and PCC use cases.

Study	T cell		T-LGL		PCC	
Model measures	Baseline	Gold	Baseline	Gold	Baseline	Gold
number of nodes	39	43	41	60	241	257
number of edges	60	74	113	193	280	373
EES measures	ECLG^{original}	ECLG^{FC}	ECLG^{original}	ECLG^{FC}	ECLG^{original}	ECLG^{FC}
number of papers	12	6	38	18	19	10
number of edges	512	465	8898	7963	18094	15866
number of nodes	95	72	496	346	658	453
mean papers per interaction (MPI)	10.7	12.9	36	46	54.9	70
mean interactions per paper (MIP)	9.75	12	17	38.4	37	64.7
path length (APL_{avg})	1.55	1.58	2.8	2.6	2.2	1.85
maximum path length	3	3	6	6	5	4
minimum path length	0	0	0	0	0	0
clustering coefficient ($Coeff_{avg}$)	0.977	0.979	0.98	0.99	0.99	0.99
degree (D_{avg})	10.7	12.9	36	46	54.9	70
number of clusters	9	6	20	9	17	7
neighborhood connectivity (NC_{avg})	11.29	13.3	37.8	46.6	56.8	70.6
betweenness centrality (BC_{avg})	0.012	0.01	0.004	0.005	0.003	0.003
closeness centrality (CC_{avg})	0.77	0.77	0.62	0.6	0.66	0.67
radiality (R_{avg})	0.85	0.86	0.81	0.8	0.82	0.8
edge betweenness (EB_{avg})	6.7	6.3	27.7	17.6	15.2	8
frequency class (FC_{avg})	1.75	1.3	1.9	1.2	2	2



	FC	D	NC	CC	R	BC
FC	1	-0.97	-0.96	0.23	-0.01	-0.14
D	-0.97	1	0.97	-0.33	-0.1	0.15
NC	-0.96	0.97	1	-0.37	-0.12	0.1
CC	0.23	-0.33	-0.37	1	0.95	-0.07
R	-0.01	-0.1	-0.12	0.95	1	-0.01
BC	-0.14	0.15	0.1	-0.07	-0.01	1

	FC	D	NC	CC	R	BC
FC	1	-0.87	-0.86	0.04	-0.12	-0.13
D	-0.87	1	0.95	-0.01	0.21	0.2
NC	-0.86	0.95	1	-0.05	0.17	0.01
CC	0.04	-0.01	-0.05	1	0.93	0.01
R	-0.12	0.21	0.17	0.93	1	0.04
BC	-0.13	0.2	0.123	-0.01	0.04	1

	FC	D	NC	CC	R	BC
FC	1	-0.94	-0.91	0.475	0.422	-0.2
D	-0.94	1	0.93	-0.39	-0.34	0.29
NC	-0.91	0.93	1	-0.64	-0.59	0.123
CC	0.47	-0.39	-0.64	1	0.99	-0.07
R	0.42	-0.34	-0.59	0.99	1	-0.06
BC	-0.2	0.29	0.123	-0.07	-0.06	1

Figure 18 Comparison of metric values for the three case studies, T cell, T-LGL, and PCC: (Top) Distribution of frequency class values within each CE set. (Middle) FC vs. D values of all nodes in ECLG^{original}. (Bottom) Correlation coefficients between different metrics.

4.3.3 Evaluation of the Proposed FC Metric

For each case study, we compute the precision and recall of the FC metric and the degree centrality metric. To determine the precision and recall values, we consider the candidate events that are also present in the golden model as true positives or *true events*, and the remaining candidate events as false positives or *false events*. Similarly, the events that are in the golden model and were not selected as candidate events are false negatives and the events that are not in the golden model and were not selected as candidate events are true negatives. We will refer to the

golden model events as *correct events*. Precision is the ratio between the number of true events and the sum of the number of true events and the number of false events, whereas recall is the ratio between the number of true events and the total number of correct events found in the CE set (i.e., the sum of the number of true positive and the number of false negative events).

We show in Figure 21 the precision and recall results for the FC candidate events of the three case studies. For the T cell case, we achieved a precision of 0.44. This means that 56% of the FC candidate events are false positives (i.e., they are not in the golden model). On the other hand, for the T-LGL case, the event precision is 0.3, and in the PCC case, it is 0.25. While in the T-LGL and PCC studies more than half of the events and entities are false positives, it is important to note that these two studies have much larger CE set, compared to the T cell study, and thus, have more candidate events. Moreover, the events that are in the golden models are not necessarily the only valid events, as there could be other events in literature that are also useful and important, and therefore, should be considered in model assembly. Our proposed methodology is able to uncover such events and suggest them as model extension candidates. For instance, in the PCC study, the events IL-6 → MMP, STAT3 → Twist, NF-κB → Bcl-2, NF-κB → Bcl-X L, STAT3 → Bcl-2 and STAT3 → Bcl-X L, AID —| P53 (where "→" represents positive regulation, and "—|" represents negative regulation) were identified as FC candidate events, and their correctness was approved by domain expert although they are not in the model. Examples of the evidence statements for those events, found in the REACH output, are: "IL-6 promotes MMP expression", "STAT3 mediated induction of Twist transcription", "the expression of the anti-apoptotic proteins Bcl-2 and Bcl-X L are promoted by both NF-κB and STAT3 and a novel mouse model of hepatocarcinogenesis triggered by AID causing deleterious p53 mutations". Therefore, the precision that we report in Figure 21 is likely smaller than the actual precision of our proposed

method due to these additional important events that the method is able to uncover, and which are not in the golden model. To elaborate more on this, we conducted the following exercise for the PCC study. We used human judgement of candidate events, that is, based on domain expert's opinion, we labeled the candidate events as true or false positives. Interestingly, the domain expert identified additional 144 events as true positives (valid FC candidate events, but not in the golden model), besides the 151 true events (FC candidate events that are also in the golden model). When we changed the status of these 144 events from false positives to true positives, the precision increased to 0.65.

We used INDRA to compute a belief score for each selected event in the T-LGL study, where we used a query to search for papers instead of preselected list of papers. INDRA generated a belief score with value greater than or equal 0.7 (out of 1) for 22 additional events. When we changed the status of these events from false positives to true positives, this has increased the event precision from 0.3 to 0.4.

The recall values for our proposed method are much higher than the precision values (Figure 21). This demonstrates the ability of our methodology to identify useful and relevant events in a given CE set. In particular, for the T cell study, the recall value is 1, as none of the correct events are missed, that is, there are zero false negatives. For PCC case, the recall value is 0.76. Finally, for the T-LGL case study, the recall is 0.70, i.e., our method missed approximately 30% of correct events. The lower recall in this study is due to removing a large number of events (41%) from the golden model to create the baseline model (Table 4), as well as using a large CE set.

It is worth noting that the values of precision and recall are highly affected by the accuracy of machine readers. There are several errors that arise from machine reading output when extracting the events from published literature. For instance, a common error that we noticed in

the PCC case study is related to the EGFR (epidermal growth factor receptor) protein. When the machine reader finds EGFR in a paper, it translates it into EGFrna which is not true and makes any event that contains the protein EGFR a false positive.

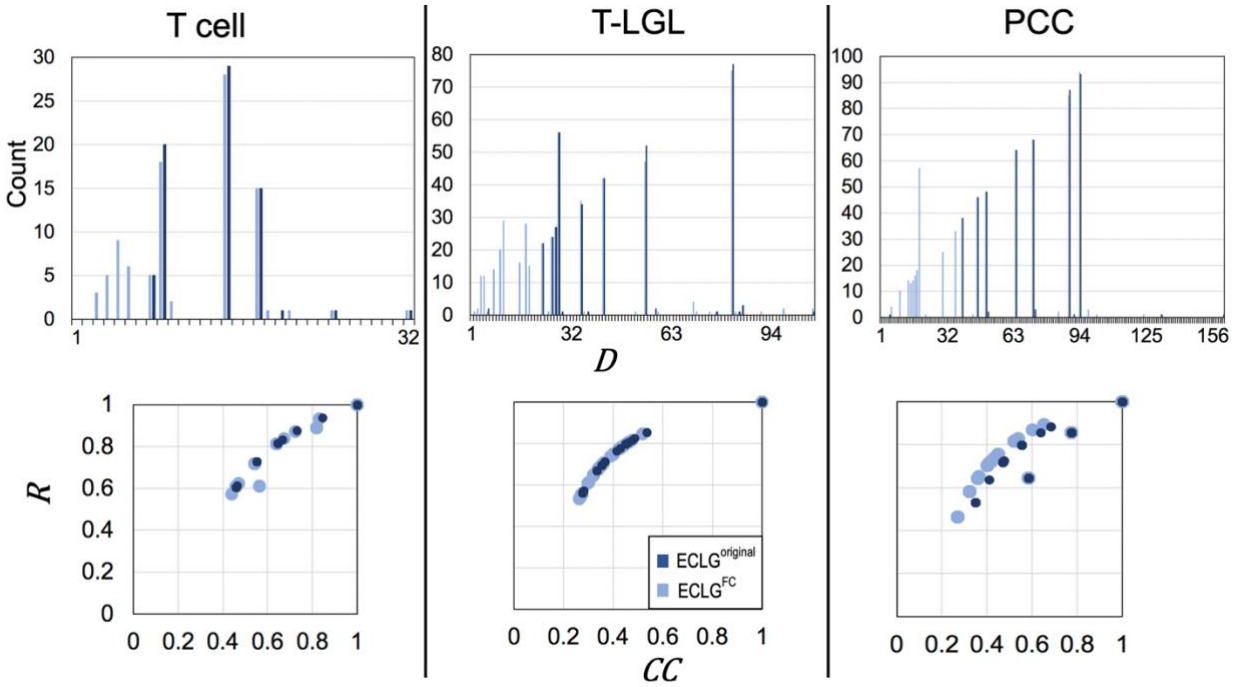


Figure 19 . Exploration of graph characteristics for the three case studies, T cell model, T-LGL model, and PCC model: (Top) distribution of degree (*D*) values, (Bottom) Radiality (*R*) vs. closeness centrality (*CC*) of the ECLG network before and after the removal of less frequent events (ECLG^{original} and ECLG^{FC}, respectively).

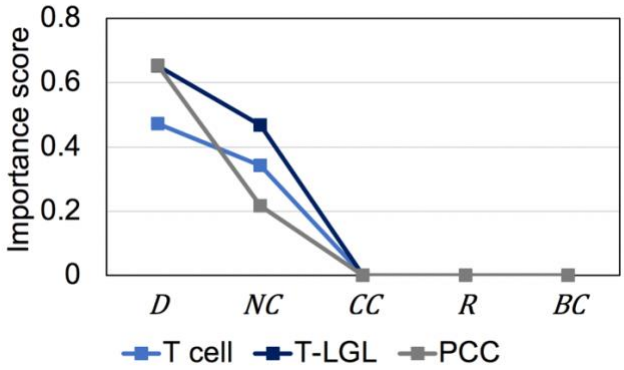


Figure 20 The importance score p_{φ}, for the centrality metrics *D*, *NC*, *CC*, *R*, and *BC* in the three case studies

4.3.4 Evaluation of the Degree Centrality Metric

Since we found that the degree centrality metric highly correlates with our proposed FC metric, we were interested in exploring the difference between the set of centrality candidate events found using the degree metric and the set of FC candidate events. We ranked the nodes (events) in the $ECLG^{original}$ in descending order of D values, i.e., from the event with highest D value to the event with lowest D value. Similar to FC, we can set a threshold and remove the events with low D values. Therefore, the top ranked nodes are the centrality candidate events to be evaluated.

For each case study, we computed the threshold as the average D value, and we removed all the events that are below this value. The generated set is the set of centrality candidate events. As shown in Table 4, the average D values are 10.7, 36 and 54.9 for T cell, T-LGL and PCC case studies, respectively. The number of events in the set $\mathcal{E}^{Central}$, after the removal of the events with small D values is 47 for the T cell case study, 215 for the T-LGL case study and 405 for the PCC case study. We show in Figure 21 the precision and recall of the centrality candidate events for the three case studies. For T cell study, the precision is 0.4 and the recall is 0.9. For the PCC study, the precision is very small, it is 0.16, and the recall is 0.6. Similarly, for the T-LGL study, the precision value is very small, it is 0.11 and the recall is 0.36. In all case studies, these values are smaller than the values reported for the FC metric. This is due to the centrality metric D removing a subset of true events that were in the FC candidate event set. These results emphasize the fact that the FC metric is more accurate in extracting true events from the CE set, compared to centrality metrics.

Finally, we conducted a two-step exercise, by first selecting the nodes (events) with $FC \leq 2$ from ECLG to form $ECLG^{FC}$. We then computed the average D value of the nodes in $ECLG^{FC}$

as our new threshold and we removed all the nodes having D value below this threshold. The number of events that we obtained are 48, 137 and 255 for the T cell, T-LGL and PCC cases, respectively. For those events, we compute the precision and recall as shown in Figure 21. For all use cases, the recall values are smaller than FC values of 0.63 for T cell, of 0.25 for T-LGL and of 0.51 for PCC. This means that more correct events were removed which reduces the recall values. However, for the PCC case, the precision value significantly increased to 0.5 since additional events that are likely false positives were removed. This suggests that using the two-step selection could be beneficial when the CE set is at the order of tens of thousands of interactions or larger.

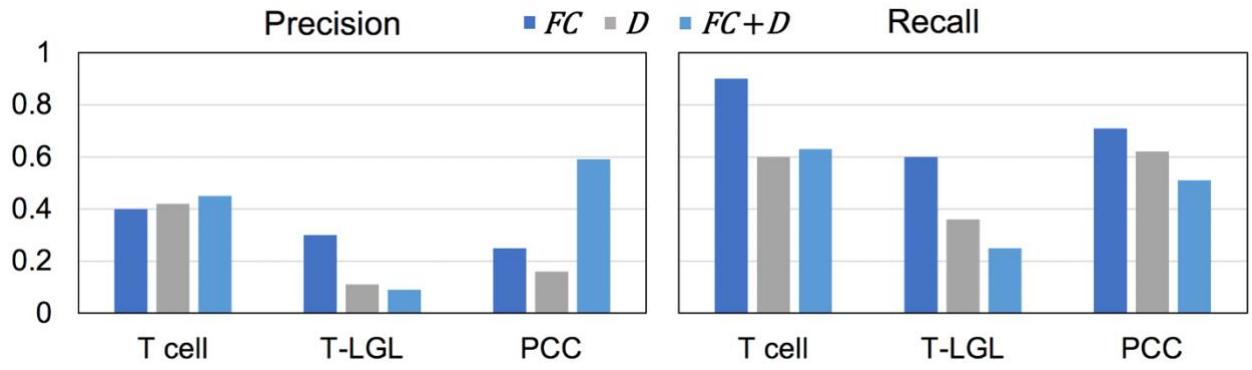


Figure 21 Precision and recall values when compared to golden models for T cell, T-LGL and PCC use cases

4.4 Conclusion

In this chapter, we presented MELOGRAPH, an automated framework for rapid model assembly that combines machine reading, the frequency class-based metric, and graph analysis. We compared the performance of our proposed frequency class-based metric to four common centrality metrics, and we also evaluated the usefulness of the centrality metrics when identifying the events that are highly supported in literature. Our results suggest that our proposed frequency class-based metric is most useful when the machine reading output has hundreds or thousands of

events, while in the case of larger extracted events sets, the event selection can be further improved when the frequency class metric is used together with the degree centrality metric. Furthermore, our methodology automatically assembles models using the information published in literature within several seconds. As such, it facilitates information reuse and data reproducibility, and it could replace hundreds or thousands of manual experiments

5.0 An efficient Approach for Informing Network Models with Knowledge from Literature using CLARINET

5.1 Objective and Applicability

CLARINET (CLARIfying NETworks) is a novel method and a tool that automatically and efficiently extends existing models, by selecting most relevant and useful information.

CLARINET explores the vast knowledge in published literature and quickly processes the information from hundreds or thousands of papers. CLARINET is a tool that explores the knowledge published in literature and suggests top candidate components and relationships to be included into models. CLARINET is novel – it combines several approaches into a novel model extension methodology: CLARINET compares, evaluates, and integrates the newly extracted information into models, utilizing graph-based approaches, return path finding, and scoring based on frequency of occurrence and co-occurrence in published literature. CLARINET is generalizable: We demonstrate in this chapter the application of CLARINET in modeling cellular signaling pathways in immune system and cancer. However, the domain information is not hard-coded, and CLARINET can be applied in many different domains and to models at different scales, from small scale intracellular signaling networks to large scale networks in political or socio-economic domains.

The main contributions of CLARINET include:

1. An automated, fast methodology and a tool that utilizes the knowledge published in literature and suggests model extensions.

2. A novel approach to study events extracted from literature as a collaboration graph, including several metrics that rely on the event occurrence and co-occurrence frequency in literature.

3. A parametrizable tool that allows users to explore different selection criteria, when automatically identifying best extensions for their models.

5.2 Proposed Methodology

The main steps and components of the CLARINET methodology are outlined in Figure 22. Similar to ACCORDION and MEOGRAPH, CLARINET has two inputs: a baseline model and the candidate event set (CE set). We evaluate CARINET ability to automatically assemble models using three benchmark models, namely, Tcell CE^{SM}, TLGL Q^{Med} and PCC BM^{Au} (Section 3.3.1).

5.2.1 ECLG Creation and Individual Assessment (IA)

Following the notion of collaboration graphs that is often used to model social networks (Grossman & Ion, 1995), we introduce the *Event Collaboration Graph* (*ECLG*). In social networks, nodes represent participants and edges connect pairs of nodes that have collaborative relationships. Similarly, we define an ECLG as a weighted undirected graph $\mathcal{G}$ ($\mathcal{V}$, $\mathcal{E}$, $\mathcal{W}v$, $\mathcal{W}e$), where $\mathcal{V}$ is a set of graph nodes, each representing a distinct event extracted from literature, $\mathcal{E}$ is a set of graph edges, each edge indicating a co-occurrence in the same paper of the two events corresponding to its adjacent nodes. $\mathcal{W}v$ and $\mathcal{W}e$ are sets of node and edge weights, respectively.

We will refer to the ECLG created from the CE set as an original ECLG. We compute the weights $\mathcal{Wv}$ and $\mathcal{We}$ based on the frequency of the event occurrence and co-occurrence in the CE set using several metrics proposed in this chapter.

In this chapter, we will use the frequency class metric introduced in MELOGRAPH chapter (Section 4.2.2) to identify the most and the least frequent events. We will show later that CLARINET is able to select the most relevant events in an accurate way using the frequency class metric. Given the CE set (with *N* total events and *M* distinct events), we compute frequency class FC_i^{IA}, for each extracted distinct event, *i*=1,..,*M*:

$$\boldsymbol{FC_i^{IA} = \left\lfloor 0.5 - log_2 \frac{f_i}{f_{max}} \right\rfloor} \quad \textbf{(5.1)}$$

where ⌊..⌋ is the floor function. We denote the frequency of each distinct event *i*, that is, the overall number of occurrences of event *i* within CE set, as f_i. We also identify all distinct events for which $f_{max} = \max(\{f_i: i=1,..,M\})$. As can be concluded from the equation above, the FC^{IA} value of the most frequent event is 0, while any event half as frequent as the most frequent event will have FC^{IA} value equal 1 (due to logarithm with base 2).

For each node that belongs to the ECLG, we find its FC^{IA} and we rank all the events in ascending order, i.e., from the most to the least frequent event. By setting a threshold for FC^{IA}, we can remove the least frequent events from the ECLG, i.e., all events with FC^{IA} larger than this threshold. This allows for extending models with the high confidence, and likely more relevant, events. Additionally, we can keep only the events that co-occur in literature with the most frequent event(s), by removing nodes in ECLG that are not connected to the nodes with FC^{IA} =0.

Thus, using the FC^{IA} metric, we automatically select a subset of EES events to be considered for model extension, called *candidate extensions*. We will refer to the ECLG obtained automatically after individual assessments and the removal of selected nodes as a *candidate ECLG*.

We note here that, unlike simple naïve event count, the FC metric helps classify events within an EES into several classes, thus allowing modelers to examine events within or across these classes. Moreover, setting a threshold based on a simple event count is arbitrary and does not account for the occurrence frequency of the other events. On the other hand, FC^{IA} is computed for each event with respect to the most frequent event, allowing modelers to use a threshold and discard the less frequent events.

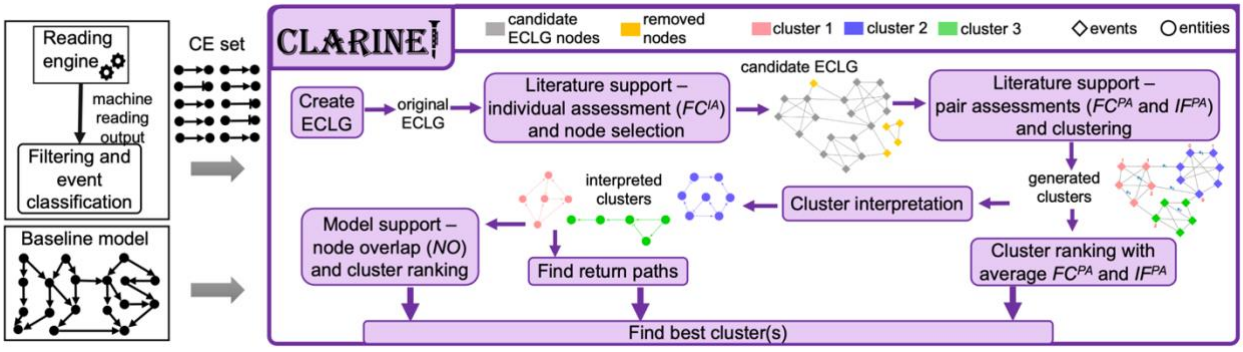


Figure 22 Illustration of CLARINET framework: (Left) CLARINET inputs: Candidate event set (CE set) and Baseline model. (Right) Flow diagram of the CLARINET processing steps and outputs.

5.2.2 Pair Assessment (PA)

To identify groups of events that would be most useful when added to the model together, we cluster the candidate ECLG with respect to the weights on its edges (literature co-occurrence-based links between events).

We measure the co-occurrence of pairs of events within the CE set, by computing a frequency class of pairs, FC^{PA}, and a weighted inverse frequency of pairs, IF^{PA}. We define the frequency class of a pair of events *i* and *j* within the EES, $FC_{i,j}^{PA}$, as:

$$\boldsymbol{FC_{i,j}^{PA} = \left\lfloor 0.5 - log_2 \frac{f_{i,j}}{f_{max,pair}} \right\rfloor} \quad \textbf{(5.2)}$$

where the co-occurrence frequency of events *i* and *j*, that is, the number of different papers in which both events *i* and *j* occur, is denoted as $f_{i,j}$, while $f_{\max,pair}$ = max({$f_{i,j}$: *i*=1,..,*M*, *j*=1,..,*M*, $i \neq j$}).

We also propose an additional pair assessment metric, $IF_{i,j}^{PA}$, that combines the inverse relative frequency of events *i* and *j*, N/f_i and N/f_j, respectively, where *N* is the total number of events in the CE set, with a co-occurrence frequency of this pair of events, $f_{i,j}$:

$$\boldsymbol{IF_{i,j}^{PA} = f_{i,j} * \left(ln\left(\frac{N}{f_i}\right) + ln\left(\frac{N}{f_j}\right)\right)} \quad \textbf{(5.3)}$$

As can be noticed, the IF^{PA} value increases proportionally to the number of times a pair of events occurs, and it is offset by the sum of the logarithms of the inverse occurrence frequencies of individual events. This inverse factor in the IF^{PA} metric provides several benefits over the FC^{PA} metric, especially in the case of rare but important extracted events. Specifically, using the inverse relative frequency of an interaction, N/f_i, increases the likelihood of selecting rare events, and therefore, their impact on the model. The logarithm is used to dampen the effect of the fraction. On the other hand, for frequent events, this fraction is low but still positive.

5.2.3 Clustering

In order to identify groups of events that would be most useful when added to the model together, we cluster the ECLG using the community detection algorithm proposed by Blondel et al. in (Blondel et al., 2008), which has been shown to generate communities of very good quality, outperforming other community detection methods. In the context of the ECLG definition from Section 3.1, given the graph $\mathcal{G}$ ($\mathcal{V}$, $\mathcal{E}$, $\mathcal{W}v$, $\mathcal{W}e$), with the node weight set $\mathcal{W}v$ being a set of FC^{IA} values, and the edge weight set $\mathcal{W}e$ a set of either FC^{PA} or IF^{PA} values, we provide here a brief

overview of the community detection algorithm. As defined in (Blondel et al., 2008), modularity *Q* is a measure of the quality of network partitioning into communities (referred to as clusters in our work) computed as the density of edges inside communities relative to the edges between communities:

$$\boldsymbol{Q} = \frac{\mathbf{1}}{\mathbf{2}\boldsymbol{m}}\sum_{\boldsymbol{u},\boldsymbol{v}}\left(\boldsymbol{w}_{\boldsymbol{u},\boldsymbol{v}} - \frac{\boldsymbol{\sigma_u}\boldsymbol{\sigma_v}}{\mathbf{2}\boldsymbol{m}}\right) * \boldsymbol{\delta}(\boldsymbol{c_u}, \boldsymbol{c_v}) \quad \textbf{(5.4)}$$

where $w_{u,v}$ represents the weight of an edge between nodes *u* and *v*, *m* is the sum of all edge weights in the network, c_u and c_v are communities of nodes *u* and *v*, σ_u and σ_v are sums of the weights of the edges connected to nodes *u* and *v*, respectively; $\delta(c_u,c_v)$=1, if *u* and *v* belong to the same community, otherwise, it is 0. In order to maximize *Q*, the algorithm has two phases that are repeated iteratively. The first phase starts by assigning each node in the network to its own community, and then, for each node *u*, we compute the change in modularity, $\Delta Q_{u,v}$, that would occur if node *u* were to be moved from its current community to the community of each of its neighbor nodes in the network:

$$\boldsymbol{\Delta Q}_{\boldsymbol{u},\boldsymbol{v}} = \left[\frac{\boldsymbol{S_v}+\mathbf{2}\boldsymbol{\sigma}_{\boldsymbol{u},\boldsymbol{v}}}{\mathbf{2}\boldsymbol{m}} - \left(\frac{\boldsymbol{S}_{\boldsymbol{v},\boldsymbol{tot}}+\boldsymbol{\sigma_u}}{\mathbf{2}\boldsymbol{m}}\right)^{\mathbf{2}}\right] - \left[\frac{\boldsymbol{S_v}}{\mathbf{2}\boldsymbol{m}} - \left(\frac{\boldsymbol{S}_{\boldsymbol{v},\boldsymbol{tot}}}{\mathbf{2}\boldsymbol{m}}\right)^{\mathbf{2}} - \left(\frac{\boldsymbol{\sigma_u}}{\mathbf{2}\boldsymbol{m}}\right)^{\mathbf{2}}\right] \quad \textbf{(5.5)}$$

where S_v is the sum of weights of the edges inside community c_v that node *u* is moving into, $S_{v,tot}$ is the sum of the weights of the edges incident to nodes in c_v, σ_u is the sum of the weights of the edges incident to node *u*, $\sigma_{u,v}$ is the sum of the weights of the edges from node *u* to nodes in c_v, and *m* is the sum of all the weights of all the edges in the network. Once this value is calculated for all communities that *u* is connected to, *u* is placed in the community that resulted in the greatest modularity increase. If no increase is possible, *u* remains in its original community. This process is applied repeatedly for all nodes as long as there is increase in *Q*.

After Q reaches a local maximum, the second phase of the algorithm creates a new network where nodes are the communities from the previous phase, the first phase proceeds with this new network, and the iterations are repeated until there is no more increase in Q. We will refer to the communities in the undirected candidate ECLG that result from applying this algorithm as *generated clusters*. We show examples of candidate ECLG and the corresponding ECLG with generated clusters in Figure 23.

Next, from the total N_C generated clusters, we are interested in selecting those clusters that would be most useful for extending the model and answer the questions that initiated literature search. To rank the clusters, we will use the two *PA* literature support metrics. For each cluster C_l, we find the average values of $FC_{i,j}^{PA}$, and $IF_{i,j}^{PA}$, across all pairs (i,j) of connected events i and j within the cluster C_l, $FC_{l_{avg}}^{PA}$ and , $IF_{l_{avg}}^{PA}$, respectively:

$$\boldsymbol{FC_{l_{avg}}^{PA} = \frac{1}{P_l}\sum_{(i,j)}\left(FC_{i,j}^{PA}\right)} \quad \textbf{(5.6)}$$

$$\boldsymbol{IF_{l_{avg}}^{PA} = \frac{1}{P_l}\sum_{(i,j)}\left(IF_{i,j}^{PA}\right)} \quad \textbf{(5.7)}$$

where $P_l = |\mathcal{E}_l|$ is the total number of edges in cluster C_l.

To add events from generated clusters to a baseline model, we convert the generated clusters, where nodes are events, and edges are literature-based co-occurrences between events, into *interpreted clusters* (weighted directed graphs), with nodes/edges being entities/events.

5.2.4 Model Support Metrics

In addition to ranking the information extracted from literature based on the literature support metrics, we also introduce a model support metric, *Node Overlap* (*NO*), which measures the connectivity of the clusters to the baseline model. More formally, we denote the baseline model

graph as $G^{BM}(V^{BM},E^{BM})$, and the graph formed by the CE set as $G^{CE}(V^{CE},E^{CE})$. For each interpreted cluster C_l we can also define $G^{C_l}(V^{C_l}, E^{C_l})$, and it is clear from the clustering algorithm that $V^{CE} = \bigcup_{l=1}^{N_c} V^{C_l}$. We define the set of *overlapping nodes* between cluster C_l and the baseline model as $V^{C_l,ON} = V^{BM} \cap V^{C_l}$ and the set of *new nodes* in C_l as $V^{C_l,new} = V^{C_l} \setminus (V^{BM} \cap V^{C_l})$. *NO* is then computed for every interpreted cluster C_l to determine the ratio between the overlapping nodes and the total number of nodes:

$$\boldsymbol{NO_l = \frac{|V^{C_l,ON}|}{|V^{C_l}|} \times 100} \quad \textbf{(5.8)}$$

We also determine whether there are any *return paths* between clusters and the baseline model. If there exists a path of connected edges $e^{path}(v_{s1},v_{tp}) = (e_{i1}(v_{s1},v_{t1}), e_{i2}(v_{s2}=v_{t1},v_{t2}), e_{i3}(v_{s3}=v_{t2},v_{t3}), \ldots, e_{ip}(v_{sp}=v_{tp-1},v_{tp}))$, we say that $e^{path}(v_{s1},v_{tp})$ is a return path, if $\{v_{s1},v_{tp}\} \in V^{BM}$, and all edges $e_{i1}, \ldots, e_{ip}$ belong to clusters in the set of interpreted clusters. The baseline model and the clusters on such return path form a *candidate extended model.*

5.2.5 Selection of Best Cluster

We rank all generated clusters with respect to each, average FC^{PA} and average IF^{PA}, and their corresponding interpreted clusters with respect to the *NO* value. We also determine which clusters belong to return paths. Finally, we say that cluster C_l is assumed to be the best candidate for model extension if it satisfies the following rule:

$$(\boldsymbol{FC^{PA}_{l_{avg}} = min}\ (\{\boldsymbol{FC^{PA}_{i_{avg}} : i = 1..N_C}\}) \quad \text{AND} \quad \boldsymbol{IF^{PA}_{l_{avg}} = max}\ (\{\boldsymbol{IF^{PA}_{i_{avg}} : i = 1..N_C}\})) \quad \text{OR}$$

$(\boldsymbol{NO_l > 50}\%$ AND $\boldsymbol{C_l}$ belongs to at least one return path) **(5.9)**

As can be seen from the equation above, a cluster C_l is considered for model extension if it satisfies either both of the literature support criteria or both of the model support criteria. For the

literature support criteria, C_l must have the lowest $FC_{l_{avg}}^{PA}$ value and the highest $IF_{l_{avg}}^{PA}$. This means that the events belonging to C_l are the most supported in literature, among all the events of the CE set. On the other hand, for the model support criteria, if the cluster C_l has more than a 50% node overlap with the baseline model and it belongs to at least one return path, then C_l will be highly connected to the baseline model. Consequently, the cluster C_l should be considered for model extension.

We introduced this equation in order to provide a guided and comprehensive way to expand dynamic network models, by adding the events that are not only just frequent in literature but are also connected to the baseline model through return paths.

5.3 Results and Discussion

To evaluate CLARINET and demonstrate its features, we conducted several experiments using the three models, Tcell CE^{SM}, TLGL Q^{Med} and PCC BM^{Au} described in Section 3.3.1. We explored how well CLARINET performs in various scenarios, small vs. large model extension, controlled vs. query-based extension, and extension of a smaller published model vs. reconstruction of a truncated model.

5.3.1 Model Extension with CLARINET

While CLARINET is fully automated, and parametrizable, to demonstrate its flexibility and the outcomes of parametrizations, we also show here results for intermediate steps. For each baseline model and CE set, CLARINET creates an ECLG, similar to the one shown in Figure 23

(T cell). For all nodes (events) in the original ECLG, CLARINET then computes FC^{IA}. As stated previously, events with $FC^{IA} = 0$ are the most frequent ones, thus strongly supported by literature, with multiple evidence statements. The users can enter a value for the FC^{IA} as a threshold for removing less frequent events (i.e., events found less often in the selected set of papers), otherwise CLARINET assumes the average FC^{IA} value within the CE set as a default threshold. We found that an average value in all three case studies is FC^{IA}=2, and using this default threshold, we removed events with FC^{IA}>2 from the ECLG. Using FC^{IA}=2, CLARINET removed 20, 150, and 205 less frequent events from the CE set in the T cell, T-LGL and PCC studies, respectively.

In Figure 23, we highlight with black color the nodes that are being removed from the ECLG for our T cell study. The number of nodes and edges in the ECLG before and after this step are shown in Table 5. As can be noticed, after the removal of the less frequent nodes, not only the size of the ECLG changed, but also other graph parameters changed. For instance, the mean number of papers per interaction, which maps to the average degree of nodes (events), increased after the removal of the less frequent nodes. The removal led to a denser graph, with strongly connected components, which is in agreement with both the increased neighborhood connectivity of the nodes and the high clustering coefficient.

CLARINET can use an additional selection criterion, to keep only the nodes of the reduced ECLG that are neighbors of the nodes representing most frequent (FC^{IA}=0) events, and it removes the rest of the nodes from the reduced ECLG. In other words, CLARINET can remove events that do not co-occur with any of the most frequent events. In Figure 23, the sub-graph enclosed in a green box is an ECLG that we would obtain for the T cell study if we applied this additional selection criterion.

Table 5 Description of each use case in terms of the size of both baseline and gold standard models, followed by the values of several graph metrics for the ECLG before and after the removal of less frequent events, for T cell, T-LGL and PCC case studies.

Study	T cell		T-LGL		PCC	
Model Measures	Baseline	Gold	Baseline	Gold	Baseline	Gold
Number of nodes	39	43	41	60	241	257
Number of edges	60	74	113	193	280	373
CE set Measures	Before	After	Before	After	Before	After
Total papers	12	6	38	18	19	10
Total edges	512	465	8898	7963	18094	15866
Total nodes	95	72	496	346	658	453
Mean Papers per Interaction (MPI)	10.7	12.9	36	46	54.9	70
Mean Interactions per Paper (MIP)	9.75	12	17	38.4	37	64.7
Clustering coefficient (Coeff)	0.977	0.979	0.98	0.99	0.99	0.99
Number of clusters	9	6	20	9	17	7
Average Neighborhood Connectivity (NC)	11.29	13.3	37.8	46.6	56.8	70.6
Average FC^{IA}	1.75	1.3	1.9	1.2	2	2

Next, after applying the literature support metrics and obtaining the candidate ECLG, CLARINET assigns weights to all edges in the candidate ECLG, using two different sets of weights, FC^{PA} and IF^{PA}, for two separate clustering procedures. CLARINET partitioned the candidate ECLG into six, nine, and seven edge weighted generated clusters, for T cell, T-LGL and PCC, respectively. These clusters include interactions from 6, 18 and 10 out of the 12, 38 and 19 papers that were selected at the beginning (Table 5). Using the two different metrics (FC^{PA} and IF^{PA}) to weigh edges did not affect the number of generated clusters and the edges within each cluster for our three studies; however, while the IF^{PA} values had a larger discrepancy between clusters, the FC^{PA} values seem to be much closer to one other.

To select the best generated cluster(s) that would be most useful when added to the baseline model, CLARINET computes for each cluster the average FC^{PA} and IF^{PA} values and ranks the

clusters according to these values. Since the FC^{PA} and IF^{PA} values computed for any given edge are usually different, the clusters' average FC^{PA} and IF^{PA} values are also different, and therefore, the ranking of clusters with respect to these values can differ as well. For each case study, we show the average FC^{PA} and IF^{PA} values for generated clusters in Figure 24. As can be noticed, for the T cell case, the ranking of clusters from lowest to highest average FC^{PA} value is C_2, C_6, C_4, C_1, C_5, C_3, and the ranking of clusters from highest to lowest average IF^{PA} value is C_2, C_6, C_5, C_4, C_1, C_3. From these rankings, we see that cluster C_2 is suggested as the best cluster in both cases, that is, it has the lowest average FC^{PA} value, and the highest average IF^{PA} value among all six clusters. On the other hand, in the T-LGL case study (Figure 24), the FC^{PA}-based ranking is C_3, C_1, C_9, C_7, C_5, C_4, C_6, C_2, C_8, whereas the IF^{PA}-based ranking is C_3, C_1, C_9, C_7, C_5, C_6, C_4, C_2, C_8. For the PCC model, the corresponding cluster rankings are C_2, C_7, C_1, C_4, C_5, C_6, C_3 and C_2, C_7, C_5, C_6, C_3, C_1, C_4, respectively.

Next, CLARINET transforms these generated clusters into interpreted clusters, and explores the connection between the interpreted clusters and the baseline model. Figure 24 shows the *NO* values for the clusters of each case study. In the T cell case study, clusters C_3 and C_5 have the highest *NO* value, i.e., the highest percentage overlap with the baseline model. For T-LGL, clusters C_1 and C_2 are the ones with highest *NO*, whereas for PCC, clusters C_2, C_4 and C_7 all have high *NO* values. We can conclude from these results that the *NO* measure identifies different clusters, compared to the ones with highest FC^{PA} and IF^{PA} weights. This demonstrates the versatility of CLARINET and the flexibility it provides to users in choosing different strategies for automated model extension.

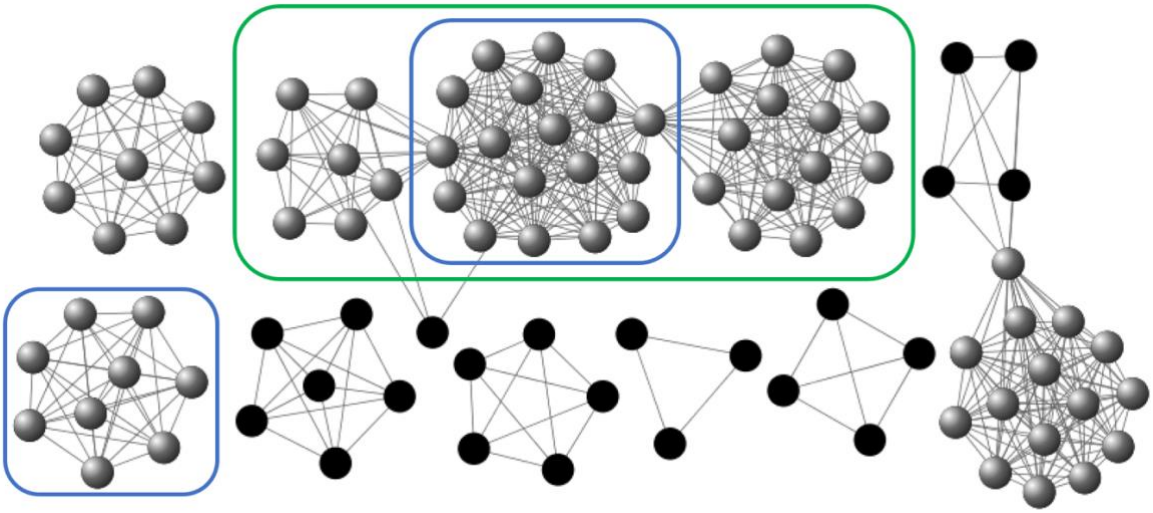


Figure 23 Candidate ECLG for the T cell case study

In addition to the metrics discussed above, the user may also be interested in extending a baseline model to include a particular element and to study its effects on the model. In such cases, if there are two or more clusters in the set of interpreted clusters that we obtained, all containing regulators and regulated elements of the element of interest, CLARINET can instead consider those clusters for extension. We are especially interested in combining these clusters if they can be connected through a return path, which starts and ends in the model. If the user selects this option, CLARINET can find return paths, thus enabling users to add key regulatory pathways that are not in the baseline model.

To illustrate the return paths, the set of events that is included in the top ECLG cluster enclosed by blue box in Figure 23, is also shown as interactions in interpreted cluster C_3 in Figure 25(a). It can be seen from Figure 25(a) that Foxo1_ext, which is a new element in the EES, is activating PTEN, which is also an element in the baseline model, $Tcell^{baseline}$. If we add only the cluster from Figure 25(a) to $Tcell^{baseline}$, we will be able to study the effect of Foxo1_ext, as it will become an input to $Tcell^{baseline}$. However, with such extension, we will not be able to study the effect of the other parts of the model on Foxo1_ext, given that Foxo1_ext is not regulated by any

other element in Tcell[baseline]. Therefore, CLARINET can search for other clusters that include Foxo1_ext regulators. One such cluster is C_2 (Figure 25(b)).

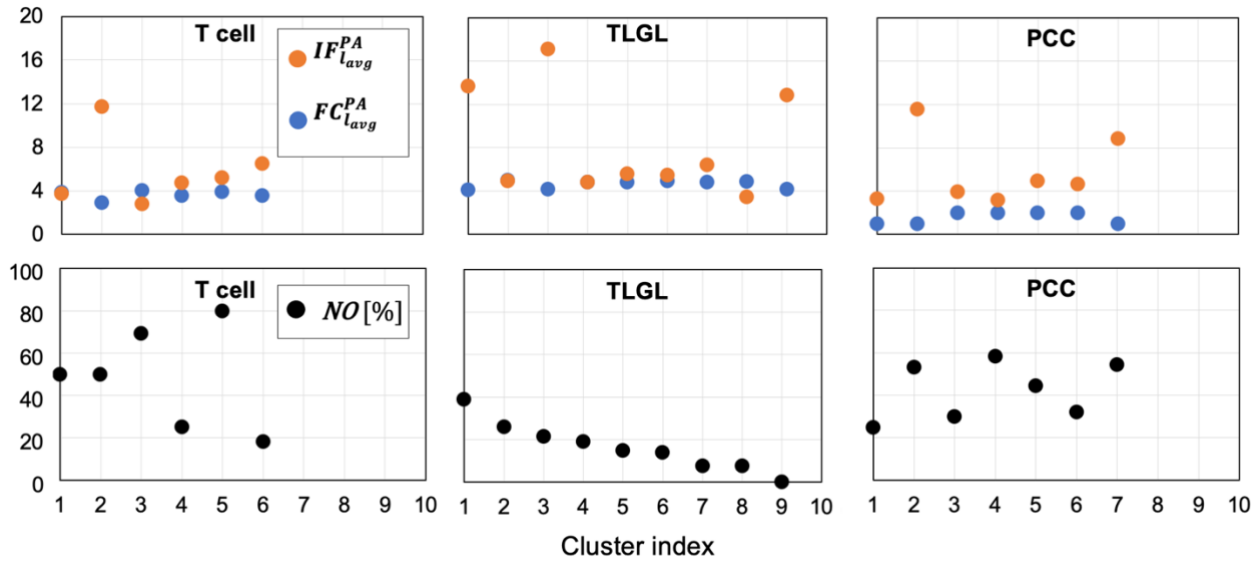


Figure 24 (Top) Average literature support metrics FC^{PA}_{avg} and IF^{PA}_{avg} for generated clusters. (Bottom) Node overlap (*NO*) between the clusters and the baseline model for the three case studies.

Cluster C_2 also corresponds to the bottom cluster enclosed by blue circle in Figure 23. In the set of the six interpreted clusters that CLARINET obtained for our T cell model case study, clusters C_2 and C_3 form a return path with Tcell[baseline], as shown in Figure 25. Thus, the final set of events that CLARINET formed by merging the two clusters in Figure 25, contains all the elements of the full model Tcell[gold] from (Hawse et al., 2015b) (FOXO1, NEDD4, MEK1, CK2), which were missing in Tcell[baseline] from(Miskov-Zivanov, Turner, et al., 2013a).

Similarly, for T-LGL case study, in the set of the nine interpreted clusters that CLARINET obtained, clusters C_3 and C_9 form a return path with the baseline model, TLGL[baseline]. Therefore, CLARINET provides the set of events formed by merging clusters C_3 and C_9 as our finally selected set of TLGL[baseline] extensions. Finally, for the PCC case study, in addition to the return path that CLARINET found between clusters C_2, C_7, and the baseline model, PCC[baseline], making the union of these two clusters a good candidate for extension, CLARINET also found return paths between

an individual cluster C_2 and $PCC^{baseline}$. Similar to the T cell study, CLARINET was also able to closely reproduce $TLGL^{gold}$ and PCC^{gold} models published in (C. A. Telmer, Sayed, et al., 2019; Zhang et al., 2008), as further detailed in the following.

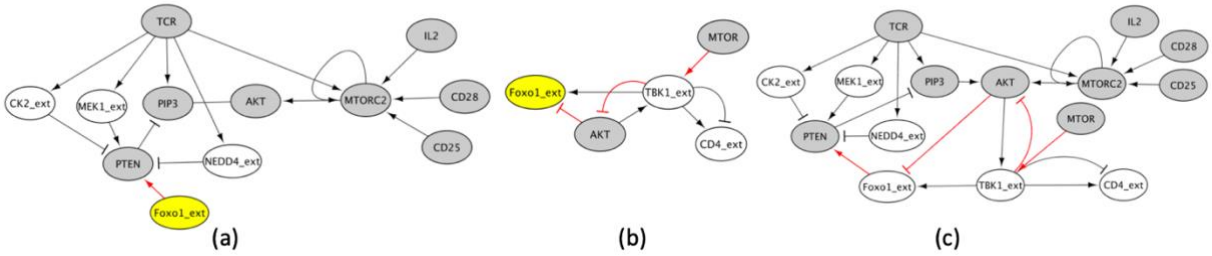


Figure 25 Cluster interpretation and preparation for extension: (a), (b) Interpreted clusters' influence graphs, C_3, C_2, respectively, for T cell case study, nodes are biological entities, pointed arrows represent activation, blunt arrows represent inhibition. Baseline model nodes are in grey and the new nodes with suffix "_ext" are in white. (a) cluster C_3 with upstream element Foxo1_ext highlighted in yellow, (b) cluster C_2 with downstream element Foxo1_ext. The return path from C_2 to C_3 is ("MTOR – TBK1_ext – AKT– Foxo1_ext – PTEN"), highlighted in red, the first node and last node of the path are MTOR and PTEN, respectively, (c) the result of merging C_3 and C_2.

5.3.2 Precision and Recall

To evaluate the relevance and the completeness of the entities and events that CLARINET selects, we computed its precision and recall. This was done by comparing the final models, $Tcell^{final}$, $TLGL^{final}$, and PCC^{final}, that were obtained using CLARINET, with the gold standard models, $Tcell^{gold}$ (Hawse et al., 2015c), $TLGL^{gold}$ (Zhang et al., 2008) and PCC^{gold} (C. Telmer et al., 2019), respectively (see Figure 26). The precision value indicates the relevance by determining the percent of events (or entities) that are selected by CLARINET, and which are at the same time a part of the gold standard model. These are usually called true positives, and therefore, we will refer to these entities and events as *true* entities and events. In Figure 27, we show precision results

for all three final models (FC^{IA}=2), for both entities and events. For the T cell case, CLARINET achieves high precision for both events and entities, 0.86 and 0.87, respectively. This means that just 14% of the events and 13% of the entities that CLARINET selected are false positives (i.e., they are not in the gold standard model). On the other hand, for the T-LGL case, the event precision is 0.45 and the entity precision is 0.5, and in the PCC case, it is 0.61 and 0.5, respectively. While in the T-LGL and PCC studies almost half of the events and entities are false positives, it is important to note that these two studies have much larger CE set, compared to the T cell study, and also compared to their baseline models, and thus, have more candidates to add to the model. Additionally, the events that are in the gold standard models are not necessarily the only valid events. In other words, there could be other events in literature, and which CLARINET suggested in its output, that are also useful and important, and should be included in the model. Therefore, the precision of CLARINET that we report here is likely smaller than its actual precision due to these additional important events that CLARINET finds but are not in the gold standard model.

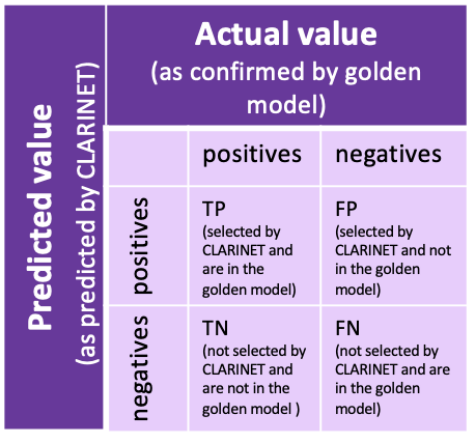


Figure 26 The matrix used to compute precision and recall of CLARINET.

To investigate this further, we conducted the following exercise for the T-LGL study. We used INDRA(Gyori et al., 2017a) to compute a belief score for each event that CLARINET selected. Interestingly, we found that INDRA generated a belief score with value greater than or

equal 0.7 (out of 1) for 27 events and 21 entities, not all of which were in the CLARINET's true event and true entity sets. When we changed the status of these additional entities and events from false positives to true positives, this has increased the entity precision to 0.7 and event precision to 0.64. Moreover, these events form more than one return path with the baseline model, i.e., they are highly connected to the baseline model. Additionally, if preferable, one can reduce the number of false positives by increasing the threshold for the *NO* value, as will be discussed in the following subsection.

To evaluate the completeness of CLARINET results we computed its recall with respect to the gold standard models. We will refer to all entities and events in gold standard models as *correct* entities and events. We compute recall as the ratio between true events (or entities) selected by CLARINET and the total number of correct events (or entities) found in the CE set. We note here that we only account for those events from the gold standard model that are in the CE set, as it is possible that there are events in the gold standard model that are not in the reading output and in the baseline model, and therefore, not present in CLARINET' s input. This is due to reading engine not recognizing in papers all the events, while the human reader who cited the papers was able to find the events and manually include them in the model.

For the T cell study, a recall value of 1 has been reported for both events and entities. This means that none of the correct events or entities are missed by CLARINET, that is, there are zero false negatives. For PCC case, the event recall value is 0.8 and the entity recall value is 0.7. Here, CLARINET achieved better values for recall than for precision. This again demonstrates the ability of CLARINET in identifying the useful and relevant entities and events in a given CE set. Similarly, in the T-LGL case recall values are higher than precision values. As shown in the figure, the entity recall is 0.74 and the event recall of 0.63, i.e., CLARINET missed approximately 26%

of correct entities and 37% of correct events. The slightly lower recall in this study, when compared to the T cell and PCC studies, is not surprising. Given the significant portion of events (41%) removed from the $TLGL^{gold}$ model to obtain $TLGL^{baseline}$, and the literature co-occurrence criteria, it is not surprising that CLARINET found in the large CE set many additional entities and events that have stronger connections with the baseline model than the ones that are in the gold model.

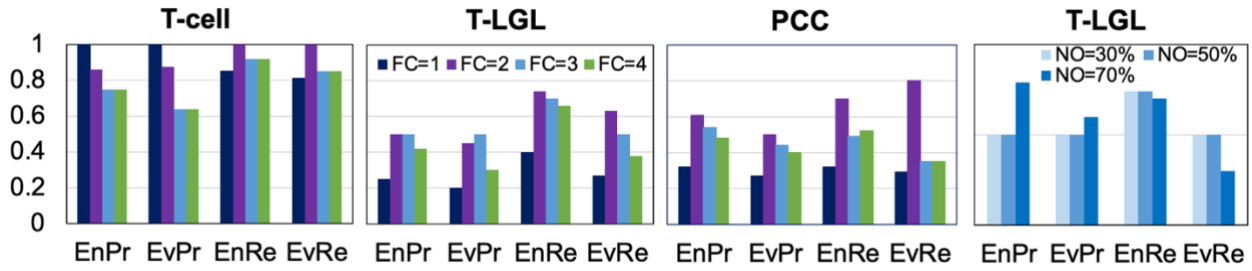


Figure 27 Precision and recall of CLARINET when compared to gold standard model for T cell, T-LGL and PCC use cases. EnPr, EvPr, EnRe and EvRe denote entity precision, event precision, entity recall and event recall, respectively.

5.3.3 Parameter Selection

We explored the effect on precision and recall when varying the two key parameters from the equation in Section 5.2.5, the thresholds for FC^{IA} and *NO* values. Varying the FC^{IA} threshold, affected the size of candidate ECLG. Consequently, this affected the number and the size of generated clusters. Increasing the FC^{IA} threshold, i.e., including more of the less frequent events in the analysis, increases the size of the candidate ECLG and will increase the number of generated clusters (Table 6). As can be seen from Figure 29, the best results were obtained for FC^{IA} threshold of 2 for T cell and PCC cases. For T-LGL, the FC^{IA} threshold of 2 or 3 resulted in similar precision and recall, while the threshold equal 2 had a better event recall value.

Table 6 Number of generated clusters for T cell, T-LGL and PCC use cases using different frequency class values

	# generated clusters			
FC^{IA}	1	2	3	4
T cell	3	6	9	10
T-LGL	3	9	13	16
PCC	2	7	9	16

Overall, the FC^{IA} threshold of 2 achieved the best results for all cases. As a reminder, this value is the average FC^{IA} value that we obtained for the EES of each case, and the user can choose to specify this threshold value as an input to CLARINET. It is also important to keep in mind that the low FC^{IA} threshold results in selecting the most frequent nodes, with the expense of ignoring any infrequent nodes that may be of interest, and the high FC^{IA} threshold leads to larger number of clusters and longer runtimes, without much benefit.

In Figure 29, we also show the effect of three different thresholds for *NO* on precision and recall. As can be noticed, any *NO* threshold below 50%, will not affect the precision and recall values. Increasing this *NO* parameter to a value higher than 50% will ensure a more connected cluster to the model, and thus, fewer false positives. However, this may not be a desirable solution in the cases when we are interested in identifying other entities and events that are not necessarily in the model. Our analysis suggests that an *NO* value of 50% or more, along with finding return paths, is sufficient in determining how well a cluster is connected to the model, while not missing potentially useful new information from literature.

5.3.4 CLARINET Scalability

We also investigated how scalable CLARINET is when applied on models and EES with different sizes. We have run and tested CLARINET on a 3.3 GHz Intel Core i5 processor. We have found that, for the T cell case, having both small model and small CE set (see Table 5), CLARINET took 2.5 seconds to run and generate clusters. For the T-LGL study, with only a slightly bigger model, but a large CE set, CLARINET took 10.1 seconds. And finally, for the PCC case, when we applied CLARINET on a large model and a much larger CE set compared to the previous two cases, the runtime was 25.4 seconds. Therefore, CLARINET can very efficiently extend baseline models that already have several hundred nodes, while exploring candidates from a CE set with tens of thousands of events.

5.4 Conclusion

In this chapter, we presented CLARINET and its underlying methodology that integrates information from published literature and expert-built models to rapidly assemble or extend models. CLARINET is parametrizable, it allows users to select different extension criteria, depending on the context, focus and goals of their models. By automatically extending models with the information published in literature, our methodology allows for rapid collection of the existing information in a consistent and comprehensive way, while facilitating information reuse and data reproducibility, and replacing hundreds or thousands of manual experiments, thereby reducing the time needed for the advancement of knowledge. We tested CLARINET on three previously published biological networks of different sizes with different machine reading outputs

that varied in size from hundreds to tens of thousands. CLARINET was able to reproduce these manually built networks with an average recall of 0.8, while also identifying new interactions with high confidence, all within several seconds.

6.0 Comparison with Previously Proposed Model Extension Methods

6.1 ACCORDION vs. the Layered-based method

We tested the effectiveness of the layered-based method from (Liang et al., 2017) when applied to the T cell case study systems. This is achieved by replacing ACCORDION with the method introduced in (Liang et al., 2017), using the same baseline T cell model (described in Section 2.6.1.1) and the three CE sets (Tcell CE^{FA}, Tcell CE^{SA} and Tcell CE^{SM}). In the layered-based method, the authors described an automated extension method that considers only the extensions that are connected to the baseline model. They first identify a set of baseline model elements of interest, and then add the extensions based on the proximity to these elements. The proximity is measured as a number of edges on a path connecting baseline model elements and new elements in extensions, and the extension is conducted in layers, starting from the baseline model. Several extension configurations are proposed in (Liang et al., 2017), depending on the extension approach that the user could be interested in. For example, the focus of model extension can be including the regulation of a certain element or a set of elements, regardless of the number of extension layers this would require. Another approach discussed in (Liang et al., 2017) focuses on reducing the number of layers while tracking the effect of adding new extensions to the baseline model. Since we focus on studying the effect of adding new extensions to the baseline model, therefore, we used the latter approach from (Liang et al., 2017) in the comparisons.

Figure 28 highlights the differences between the results of our method and the method from (Liang et al., 2017), when tested using statistical model checking. We compared the goal probability values (P_{t_j}) for each property and each CM for the two methods. As can be observed,

ACCORDION outperforms the method from (Liang et al., 2017) in the case of Tcell CE^{FA} and Tcell CE^{SA}. However, in the Tcell CE^{SM} case, the method from (Liang et al., 2017) shows slightly better results. These results indicate that the layer-based approach is less effective when used on a large set of CEIs and without any human intervention.

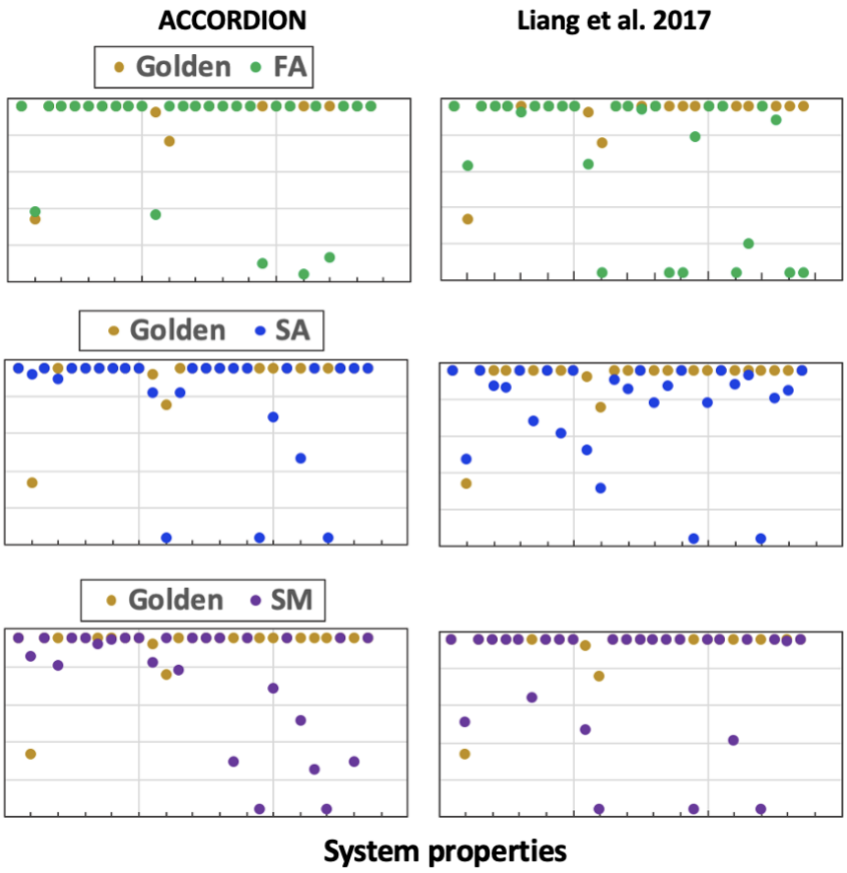


Figure 28 Comparison of the goal probability estimate P_{t_j} for all 27 properties, for the golden model, and for the best model obtained from each of the three CE sets (Tcell CE^{FA}, Tcell CE^{SA} and Tcell CE^{SM}) using ACCORDION and the method from (Liang et al., 2017)

The visualization of the topology of the sets of extensions extracted by each method is shown in Figure 29. ACCORDION provides concise groups of connected CEs, that are at the same time connected to the baseline model through return paths. On the other hand, the networks generated by the method from (Liang et al., 2017), show several nodes that are extensions and are

downstream from the baseline model, which means they are not affecting the baseline model. Thus, the comparisons we conducted suggest that the Liang et al. method has two major limitations that ACCORDION overcomes: it is subjective and prone to human judgment variation in selecting the number of elements of interest and the number of layers, and it becomes impractical with the large number of layers.

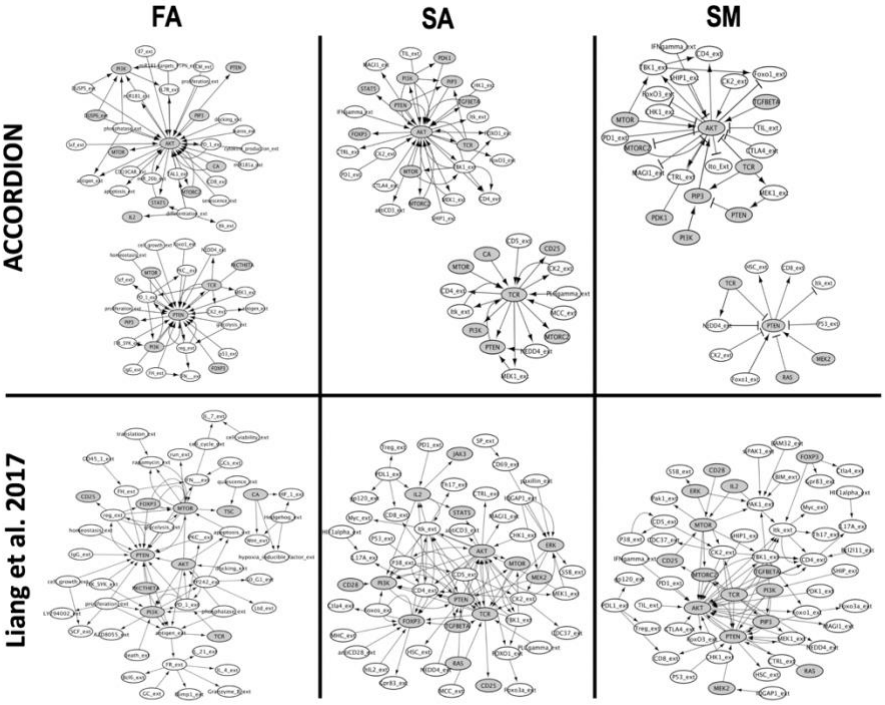


Figure 29 Clusters that were included in the final CMs for the Tcell CE^{FA}, Tcell CE^{SA} and Tcell CE^{SM} cases, for ACCORDION and (Liang et al., 2017) (white nodes are new nodes from the CEIs and gray nodes are the nodes in the CEIs that also exist in the baseline model).

6.2 CLARINET vs. GA-based Method vs. Layered-based Method

To evaluate CLARINET against the GA-based method and the Layered-based method, we applied each method on the CE set that we obtained in the T cell model case study, and we compared the selected groups of extensions. We show the selected candidate extensions obtained with the GA-method (Section 1.2) and the Layered-based method (Section 1.2) in Figure 30. The layered-based method proposed in (Liang et al., 2017) adds candidate extensions to the baseline model in layers. For example, all candidate extensions with both nodes in the baseline model belong to layer 0, those with one node in the baseline model are in layer 1, and so on. Therefore, the output of the layered-based method includes elements that do not regulate other model elements (thus, called "hanging"), and they can be seen in Figure 30. This makes their methodology less practical, especially if it is applied on a large-scale model and large CE set. Compared to the work in (Liang et al., 2017), CLARINET approaches the model extension challenge in a more inclusive way, by combining several metrics, based on occurrence and co-occurrence frequencies in published literature, and the connectivity to the baseline model. This way, CLARINET provides groups of connected events that are also well connected with the baseline model through return paths Figure 25.

We also show in Figure 30, several groups of extensions selected by GA-method (Section 1.2) (Sayed, Bocan, et al., 2018). Due to the non-deterministic behavior of the genetic algorithm, there is more than one set of extensions selected with this method. However, all the selected subsets share the same characteristics, they contain several disconnected components, and they lack the main regulations for PTEN, which is one of the key elements in Tcell gold model (Hawse et al., 2015b). Interestingly, the extensions are connected through a return path, AKT→Foxo1_ext→PTEN, which means the interactions are connected to Tcell baseline model.

However, when compared to the manually extended model, there are still some missing interactions such as CK2→PTEN, MEK1→PTEN, and NEDD4→PTEN. Moreover, similar to the results obtained for the method from the layered-based method, there are several candidate extensions that include hanging nodes, and therefore, do not affect the model.

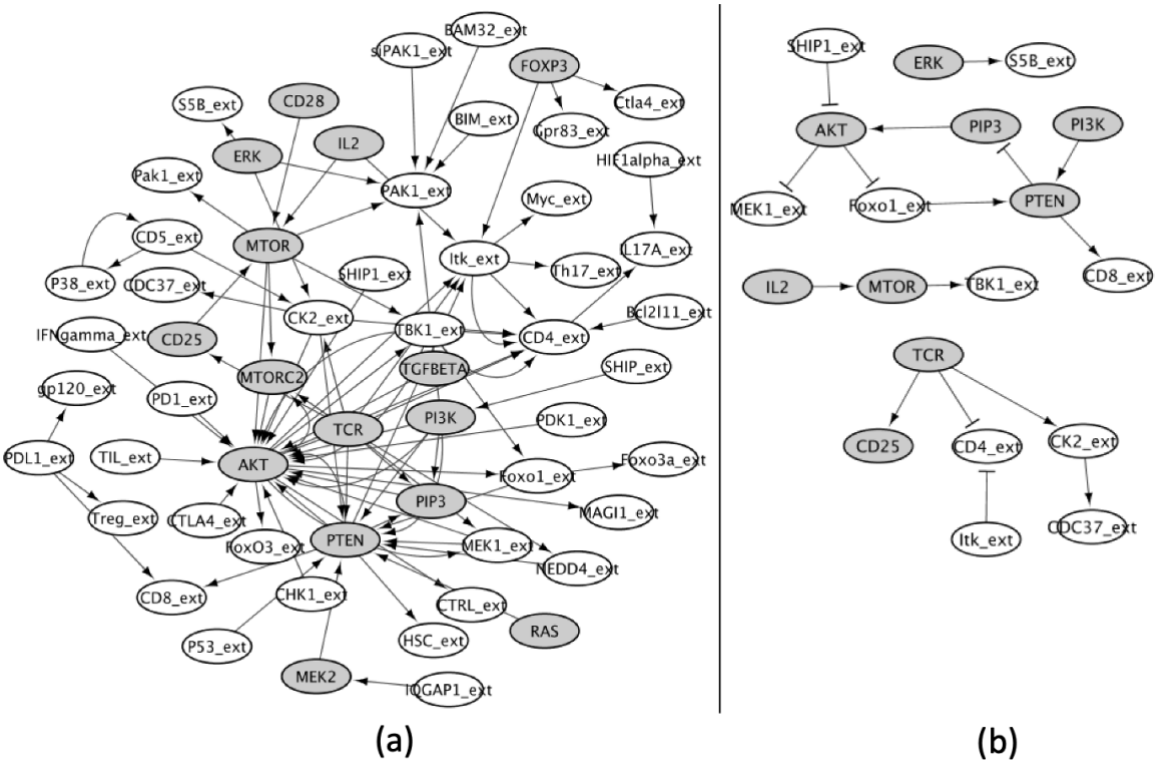


Figure 30 Comparison between the finally selected interactions by the layered-based method [4], (b) GA-based method

As shown in Figure 31, the GA-based method achieves a better entity and event precision than the method in the layered-based method. The low event precision 0.3, and low entity precision 0.21 in the layered-based method, is due to the large number of false positives. On the other hand, the entity and event precision of the GA-based method are 0.57 and 0.53, respectively. Both methods have similar entity recall values of approximately 0.8 in layered-based and 0.7 in GA-based, however, the GA-based method missed a number of events, resulting in a low event recall of 0.4, whereas the event recall value of the layered-based is 0.8. On the other hand, as shown in

Figure 31 and Figure 27, CLARINET outperformed the results in the Layered-based and the GA-based methods. For the T cell case study, entity and event precision values are 0.87 and 0.86, respectively. Moreover, the recall value is 1 for both entities and events.

Thus, from the comparisons we conducted for this case study, using the literature and model support metrics, CLARINET outperforms both Layered-based and the GA-based methods in selecting the best set of model extensions.

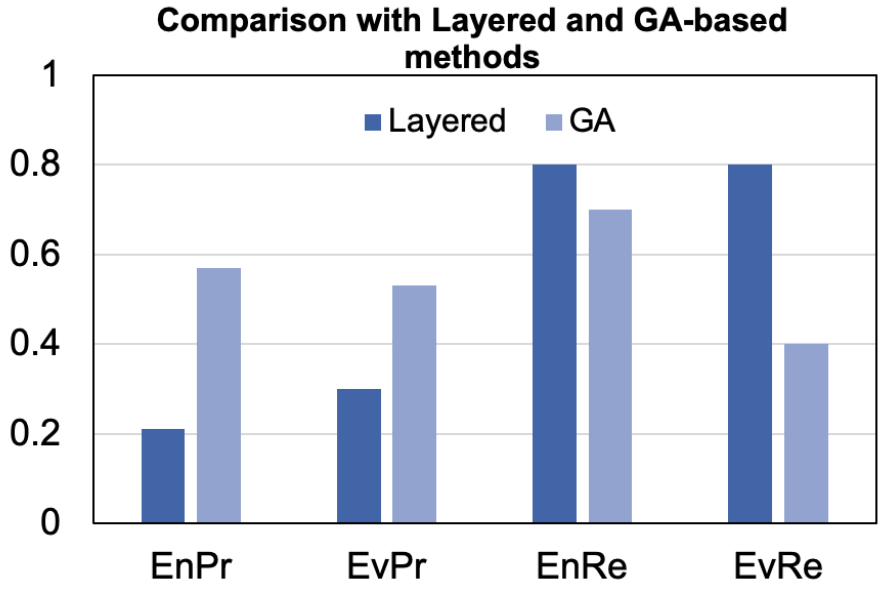


Figure 31 Precision and recall when compared to gold standard model for T cell use case, for Layered-based methods and GA-based method. EnPr, EvPr, EnRe and EvRe denote entity precision, event precision, entity recall and event recall, respectively.

6.3 The Baseline Model (BM) Error vs. CLARINET Error vs. ACCORDION Error vs. Layered-based Method Error

The goal of this section is to explore how each of the proposed methods recapitulates the desired system behavior when compared to the baseline model (BM). This is achieved by

computing the average model error $\varepsilon_{\mathcal{T},avg}^{CM_{recommended}}$ introduced in Section 3.2.7, for the best recommended candidate model (CM) by CLARINET, ACCORDION and the Layered-based method as well as the baseline model (BM). Figure 32 shows that ACCORDION obtains the lowest $\varepsilon_{\mathcal{T},avg}^{CM_{recommended}}$. We applied the layered-based approach from (Liang et al., 2017) only on the T cell case study, since it has been shown to mainly work on smaller models, and we applied the approach from (Ahmed, Telmer, et al., 2021) or CLARINET on all three baseline models. CLARINET relies only on the event occurrences and cooccurrences in literature, without accounting for dynamic behavior, and therefore, ACCORDION outperforms it, as it is guided by the desired system behavior (i.e., the set of properties $\mathcal{T}$ and their corresponding goal property probabilities P$\mathcal{t}j$).

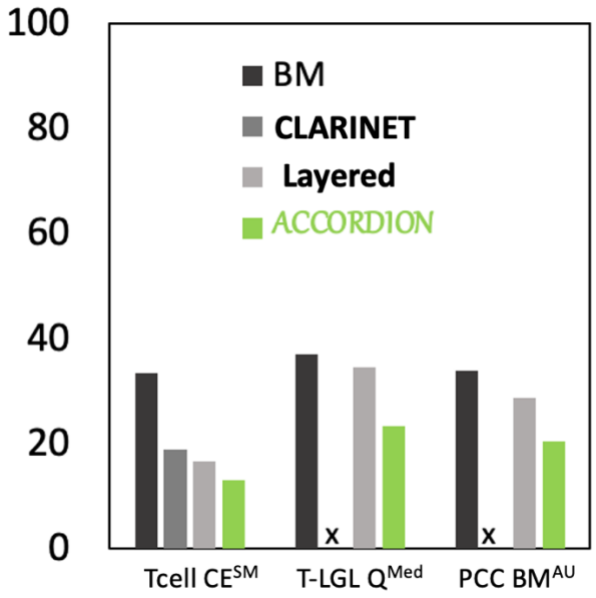


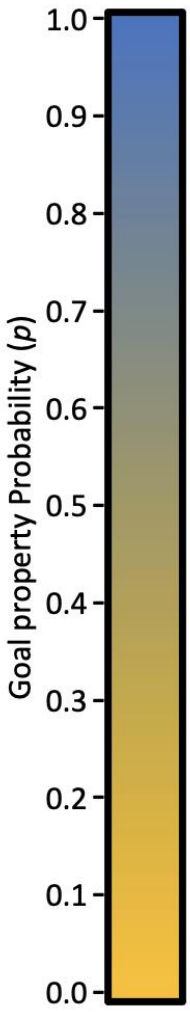
Figure 32 Comparison between BM error and the top recommended model by ACCORDION, CLARINET and Layered-based method

6.4 The New FIDDLE tool vs. ACCORDION vs. CLARINET vs. Layered-based Method tool

FIDDLE (Finding Interactions using Diagram Driven modeL Extension). A tool described in (Butchy & Telmer, n.d.) (Ahmed, Butchy, et al., 2021)that employs two methods based on network search algorithms—Breadth First Addition (BFA) and Depth First Addition (DFA)—to automatically assemble or extend models with the knowledge extracted from published literature. FIDDLE is able to refine models by systematically adding known biological interactions into intermediate models, measuring changes in model performance, and then adding or discarding interactions based on whether they improve the model performance metric. Both BFA and DFA scale linearly with the size of the model they are tasked to extend, and the number of potential interactions with which to extend the model.

To demonstrate the accuracy, efficiency, and utility of each tool, we applied each model extension tool on the Tcell CE^{SM} case (see Section 2.6.1and Section 3.3.1.1). Our main goal with this case study is to show that each tool is able to automatically assemble and extend an existing published model into another published and manually built model using new elements and new interactions automatically extracted from published literature. Figure 33 highlights the differences between the results obtained for each tool when tested using statistical model checking. The GA-based method features the best performance as scored through statistical model checking. Due to its iterative nature, the time required to perform GA-based extension increases with the number of possible extensions and can be prohibitively long when applied to large scale models (Sayed, Bocan, et al., 2018). Both ACCORDION and CLARINET balance performance with scalability and can be applied to large scale models, as well as large scale machine reading output, as demonstrated in Chapter 2 and Chapter 4. CLARINET scores the newly extracted events based on

both the evidence from literature and the connectivity to the baseline model. If the user is interested in collecting new interactions that are strongly connected to each other and strongly connected to the baseline model, then, ACCORDION would be a better choice; since it adds paths of connected interactions, which are at the same time connected to the baseline model. The layered-based and BFA methods perform similarly, despite adding different number of extensions to the baseline model. The layer-based method is meant to be applied when the user is interested in collecting new, relevant interactions that are directly connected to the baseline model. The DFA method performed the worst, scoring below the baseline model. This can be attributed to optimizing a scoring metric different than statistical model checking. In fact, both FIDDLE methods attempt to optimize the same metric with the fewest number of extensions to the baseline model. Their poor performance points to their metric being a poor stand in for statistical model checking, and the stipulation to minimize the number of additional extensions as an unnecessary restraint.



Scenario	Property	Baseline	Layer-Based	GA-Based	ACCORDION	CLARINET	BFA	DFA	Golden
1	1	0.94	0.96	0.94	0.96	0.94	0.94	0.29	0.96
	2	0.06	0.51	0.94	0.73	0.94	0.06	0.06	0.34
	3	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	4	0.94	0.96	0.94	0.75	0.94	0.94	0.69	0.96
	5	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	6	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	7	0.94	0.64	0.94	0.96	0.94	0.94	0.94	0.96
	8	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	9	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
2	10	0.94	0.96	0.94	0.96	0.94	0.87	0.36	0.96
	11	0.06	0.47	0.94	0.40	0.39	0.06	0.06	0.93
	12	0.06	0.04	0.16	0.96	0.94	0.06	0.06	0.76
	13	0.94	0.96	0.94	0.81	0.82	0.93	0.68	0.96
	14	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	15	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	16	0.06	0.96	0.94	0.96	0.94	0.06	0.06	0.96
	17	0.06	0.96	0.94	0.96	0.94	0.06	0.06	0.96
	18	0.06	0.96	0.94	0.96	0.94	0.06	0.06	0.96
3	19	0.06	0.04	0.94	0.04	0.14	0.94	0.94	0.96
	20	0.94	0.96	0.72	0.96	0.94	0.94	0.94	0.96
	21	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	22	0.06	0.42	0.94	0.96	0.94	0.94	0.94	0.96
	23	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
	24	0.06	0.04	0.94	0.04	0.11	0.94	0.94	0.96
	25	0.06	0.96	0.94	0.96	0.94	0.06	0.06	0.96
	26	0.94	0.95	0.94	0.96	0.94	0.94	0.94	0.96
	27	0.94	0.96	0.94	0.96	0.94	0.94	0.94	0.96
>0.85		17	20	25	21	23	20	16	25
Time		-	hrs	days	secs	secs	hrs	hrs	-
# Ext		-	70	16	40	23	4	4	-

Figure 33 Comparison of the model checking goal property probability *p* for the baseline model, golden model, and the best candidate model (CM) obtained from each of the five tools: Layer-based, GA-based, ACCORDION, CLARINET and FIDDLE (BFA and DFA), when run on a 3.3 GHz Intel Core i5 processor. In the last three rows, we show the number of properties with probability estimates >0.85, the length of time for each method, and the number of extensions added to the baseline model.

7.0 Conclusions and Future Work

Automatically extending models with the information published in literature allows for rapid collection of the existing information in a consistent and comprehensive way. It also facilitates information reuse and data reproducibility. In this dissertation, we introduced three novel efforts in this direction. We demonstrated the respective benefits and drawbacks of each tool and we tested them on a previously published biological models. Moreover, we compared them with existing related methods. These methods and software tools represent a novel effort to replace hundreds or thousands of manual experiments and have a potential to significantly accelerate the advancement of scientific knowledge.

In this section we explore the possible future directions.

7.1 Exploring Different Dynamic Network Model Versions and System Property Testing Results in order to develop a Heuristic to Modify Model Update Rules

As discussed before in Chapters 3, 4 and 5, after choosing the extension method, we create a set of model candidate extensions. These sets extend the static interaction map of the baseline model. Logical rules, on the other hand, allow for dynamic analysis of the model, as variable states change in time according to their update functions. Therefore, the set of logic update rules represents executable model. Incorporating new components into executable model rules can be done in several different ways. For instance, if the original rule is A =BorC, and the extension interaction states that D positively regulates A, then the new rule will be either A = (B or C) and

D, or A = BorCorD. Other logic functions could be derived as well, but this mainly depends on the information available in reading output about these interactions. Given that individual reading outputs only provide information of type 'Entity A regulates entity B' (in our example, D positively regulates A), and no additional information about interactions with other regulators is given, we add new elements to update rules using either OR or AND operation.

To automatically find the best set of update rules from thousands of combinations generated by incorporating AND or OR operators, we may use the Genetic algorithm (Whitley, 1994) as a search technique for model selection of update rules (model extension).

Genetic algorithms (GA) are optimization and search algorithms that are based on the mechanisms of natural selection of genetics. Unlike many other search algorithms, GA's search from a population of candidate solutions simultaneously, thus reducing the probability of getting stuck in a local optimum. GA has the ability to solve problems with large search spaces, high order, multimodality, discontinuity and noise disturbance.

The basic construction of a simple GA is to consider a population of individuals that each represents a potential solution to the given problem. The relative success of each individual that each represents a potential solution to the given problem. Each individual has a relative success to solve this problem that is considered to be its *fitness*. The latter is then used to reproduce fitter individuals to produce offspring of the next generation that is similar but not identical. Iterating this process will efficiently sample the space of potential individuals and converges to the most fit. In specific, if we have a population of N individuals, each represents a chromosomal string of L values. An initial population g_0 is constructed at random, and each individual is evaluated by some objective function which is the *fitness function*.

The algorithm then performs two operations: 1- selection algorithm that uses the population's *N* fitness measures to determine how many offspring from g_0 will contribute in g_1, and 2- a set of genetic operators are applied to the offspring to make their genetic information different from their parents. g_1 is the resulting population that will be evaluated again, and the cycle is repeated until a notification is made by some measure suggesting that the population has converged.

The principal genetic operators are mutation, crossover and reproduction. Mutation is the sudden inheritable change of a gene from one form to another, while Crossover is the exchange of genetic material between two chromosomes. Reproduction recombines the genetic information of two individuals to produce offspring using the genetic operators: mutation and crossover. The individuals that are more fit have greater probability of being selected for reproduction than less fit individuals.

In terms of computational efficiency, we will work on parallelization of the GA algorithm as well as the model checking algorithm to further increase its execution efficiency.

7.2 Discovering Patterns in the Machine Reading Output to identify the Most Influential Events

As the amount of biological data in the public domain grows, the need for building better modeling and analysis techniques becomes crucial as more cooperation between biology and computer algorithms is considered of great importance. Modeling facilitates explaining systems, helps in describing the dynamics of systems, and sheds the light on new questions or challenges (Epstein, 2008b). However, there are several challenges stemming from limited data quality and

standardization coupled with a dramatic increase in data size, in addition to, creating models is most often a human dependent task. This leads to the slow development of models. As a consequence, there is a pressing need for automating the process of extracting useful information from literature and assembling them into models, to enable researchers to understand and reason about systems described in the literature.

We are particularly interested in developing innovative approaches that utilize new or new combinations of extracted data to address previously intractable questions. Therefore, we may focus on cutting-edge methods aimed at pattern discovery in published literature. Our main goal is to make most of this information useful for the analysis of the dynamics of signaling pathways. To efficiently achieve that, we construct a network of interactions which gives rise to a signaling pathway in a biologically consistent and meaningful manner. This can help identify the properties of the network and highlight the information that are mostly supported in literature, among all the interactions of the machine reading output. Those interactions would be most useful for extending the model in order to answer the questions that initiated literature search.

Therefore, the main objectives of this study include:

1. Finding patterns of how the interactions are connected.

2. Finding patterns in how the interactions are connected through publications.

3. Finding patterns in terms of how much of the information is relevant to existing numerical data.

7.3 Expanding Dynamic Network Models using a Vector Space Representation of the Events Extracted from Literature

In Natural language processing (NLP), in particular, while modeling a language, a data sparseness problem is caused by the insufficiency of training data (Katz, 1987), which in turn, makes the infrequent words have unreliable probability. In order to give the infrequent words more significance, several efforts ((Ney & Vi, 1995); (Naptali et al., 2009); (P.Brown et al., 1992)) were done to map words into classes. This allowed these infrequent words to rely on other more frequent words in the same class. Latent semantic analysis (LSA) (Dumais, 2004), is an NLP technique for analyzing relationships between a set of documents and the terms they contain by producing a set of concepts (classes) related to the documents and terms. In a typical LSA framework, a word-document matrix is commonly used to represent a collection of text. This matrix determines how many times a word occurs in a certain document without taking the word order in the sentence into account. Therefore, a word co-occurrence matrix (Demeester et al., 2013) has been proposed in order to keep the word order.

As a possible future direction, we propose to inherit some of these NLP concepts to analyze machine reading output events. Our proposed pipeline is outlined in Figure 34. We first represent the machine reading output set of events as a mathematical entity, a matrix. We then apply Singular Value Decomposition (SVD)(GOLUB & REINSCH, 1970) on the matrix. Using the matrices obtained from the SVD, every event and paper is projected to the continuous space. Eventually, a clustering is applied to get classes of events.

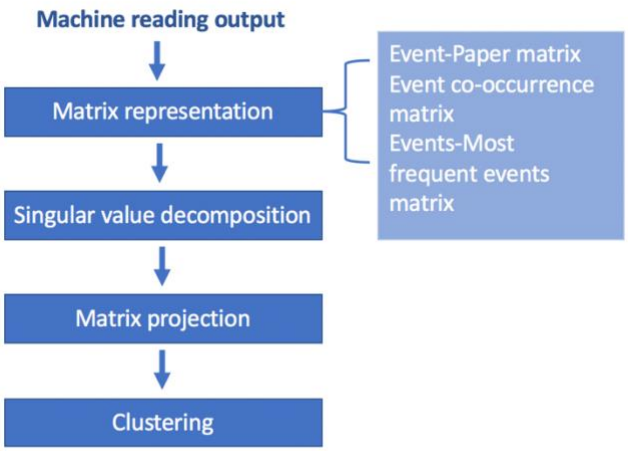


Figure 34 Illustration of the proposed methodology

Bibliography

Ahmed, Y., Butchy, A. A., Sayed, K., Telmer, C., & Miskov-Zivanov, N. (2021). *New advances in the automation of context-aware information selection and guided model assembly*.

Ahmed, Y., & Miskov-Zivanov, N. (2021). Guided assembly of cellular network models from knowledge in literature. *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, 4458–4464. https://doi.org/10.1109/EMBC46164.2021.9630181

Ahmed, Y., Telmer, C., & Miskov-Zivanov, N. (2021). CLARINET: Efficient learning of dynamic network models from literature. *Bioinformatics Advances*. https://doi.org/10.1093/bioadv/vbab006

Ahmed, Y., Telmer, C., & Miskov-Zivanov, N. (2022). *Context-aware information selection and model recommendation with ACCORDION*.

Albert, I., Thakar, J., Li, S., Zhang, R., & Albert, R. (2008). Boolean network simulations for life scientists. *Source Code for Biology and Medicine*, *3*, 1–8. https://doi.org/10.1186/1751-0473-3-16

Albert, R., & Thakar, J. (2014). Boolean modeling: a logic-based dynamic approach for understanding signaling and regulatory networks and for making useful predictions. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, *6*(5), 353–369. https://doi.org/10.1002/wsbm.1273

Allen, J. F., & Teng, C. M. (2017). Broad coverage, domain-generic deep semantic parsing. *AAAI Spring Symposium - Technical Report*, *SS-17-01-*, 108–115.

Aoki, K. F., & Kanehisa, M. (2005). Using the KEGG Database Resource. *Current Protocols in Bioinformatics*, *11*(1), 1–54. https://doi.org/10.1002/0471250953.bi0112s11

Bader, G. D., & Hogue, C. W. V. (2003). An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, *27*, 1–27.

Bardeesy, N., & DePinho, R. A. (2002). Pancreatic cancer biology and genetics. *Nature Reviews. Cancer*, *2*(12), 897–909. https://doi.org/10.1038/nrc949

Bateman, A., Martin, M. J., O'Donovan, C., Magrane, M., Alpi, E., Antunes, R., Bely, B., Bingley, M., Bonilla, C., Britto, R., Bursteinas, B., Bye-AJee, H., Cowley, A., da Silva, A., de Giorgi, M., Dogan, T., Fazzini, F., Castro, L. G., Figueira, L., … Zhang, J. (2017). UniProt: The universal protein knowledgebase. *Nucleic Acids Research*, *45*(D1), D158–D169. https://doi.org/10.1093/nar/gkw1099

Bianconi, F., Baldelli, E., Ludovini, V., Crinò, L., Flacco, A., & Valigi, P. (2012). Computational model of EGFR and IGF1R pathways in lung cancer: A Systems Biology approach for Translational Oncology. *Biotechnology Advances*, *30*(1), 142–153. https://doi.org/10.1016/j.biotechadv.2011.05.010

Blatt, M., Wiseman, S., & Domany, E. (1996). Superparamagnetic clustering of data. *Physical Review Letters*, *76*(18), 3251–3254. https://doi.org/10.1103/PhysRevLett.76.3251

Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, *2008*(10). https://doi.org/10.1088/1742-5468/2008/10/P10008

Brohée, S., & van Helden, J. (2006). Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, *7*. https://doi.org/10.1186/1471-2105-7-488

Büchel, F., Rodriguez, N., Swainston, N., Wrzodek, C., Czauderna, T., Keller, R., Mittag, F., Schubert, M., Glont, M., Golebiewski, M., van Iersel, M., Keating, S., Rall, M., Wybrow, M., Hermjakob, H., Hucka, M., Kell, D. B., Müller, W., Mendes, P., … le Novère, N. (2013). Path2Models: Large-scale generation of computational models from biochemical pathway maps. *BMC Systems Biology*, *7*. https://doi.org/10.1186/1752-0509-7-116

Burns, G. A. P. C., Dasigi, P., de Waard, A., & Hovy, E. H. (2016). Automated detection of discourse segment and experimental types from the text of cancer pathway results sections. *Database : The Journal of Biological Databases and Curation*, *2016*, 1–12. https://doi.org/10.1093/database/baw122

Butchy, A. A., & Telmer, C. A. (n.d.). *FIDDLE : E cient Assembly of Networks That Satisfy Desired Behavior*.

Clarke, M. A., & Fisher, J. (2020). Executable cancer models: successes and challenges. *Nature Reviews Cancer*, *20*(6), 343–354. https://doi.org/10.1038/s41568-020-0258-x

Das, K., Samanta, S., & Pal, M. (2018). Study on centrality measures in social networks: a survey. *Social Network Analysis and Mining*, *8*(1), 13. https://doi.org/10.1007/s13278-018-0493-2

Demeester, T., Sutskever, I., Chen, K., Corrado, G., & Ean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. *Proceedings of the 26th International Conference on Neural Information Processing Systems*, *2*, 3111–3119. https://doi.org/10.18653/v1/d16-1146

Dumais, S. T. (2004). Latent Semantic Analysis. *Annual Review of Information Science and Technology (ARIST)*, *38*. https://doi.org/10.1177/053331647000300202

Epstein, J. (2008a). Why Model? *Cybernetics and Systems*, *35*(2–3), 117–128. https://doi.org/10.1080/01969720490426803

Epstein, J. (2008b). Why model? *Journal of Artificial Societies and Social …*, *11*(4), 6. https://doi.org/10.1080/01969720490426803

Etzioni, O., Banko, M., & Cafarella, M. J. (2006). Machine Reading. *Ameri*, 1517–1519. https://doi.org/10.1145/1298406.1298407

Fabregat, A., Jupe, S., Matthews, L., Sidiropoulos, K., Gillespie, M., Garapati, P., Haw, R., Jassal, B., Korninger, F., May, B., Milacic, M., Roca, C. D., Rothfels, K., Sevilla, C., Shamovsky, V., Shorser, S., Varusai, T., Viteri, G., Weiser, J., … D'Eustachio, P. (2018). The Reactome Pathway Knowledgebase. *Nucleic Acids Research*, *46*(D1), D649–D655. https://doi.org/10.1093/nar/gkx1132

Faeder, J. R., Blinov, M. L., & Hlavacek, W. S. (2009). *Rule-Based Modeling of Biochemical Systems with BioNetGen* (pp. 113–167). https://doi.org/10.1007/978-1-59745-525-1_5

Fisher, A., Rudin, C., & Dominici, F. (2018). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *ArXiv*, *Vi*.

Fisher, J., & Henzinger, T. A. (2007). Executable cell biology. *Nature Biotechnology*, *25*(11), 1239–1249. https://doi.org/10.1038/nbt1356

Frey, B. J., & Dueck, D. (2007). Clustering by passing messages between data points. *Science*, *315*(5814), 972–976. https://doi.org/10.1126/science.1136800

Frisoni, G., Moro, G., & Carbonaro, A. (2021). A Survey on Event Extraction for Natural Language Understanding: Riding the Biomedical Literature Wave. *IEEE Access*, *9*, 160721–160757. https://doi.org/10.1109/ACCESS.2021.3130956

Gagniuc, P. A. (2017). *Markov Chains: From Theory to Implementation and Experimentation*.

GOLUB, G. H., & REINSCH, C. (1970). Singular Value Decomposition and Least Squares Solutions. *Part of the Handbook for Automatic Computation Book Series*.

Gómez Tejeda Zañudo, J., Scaltriti, M., & Albert, R. (2017). A network modeling approach to elucidate drug resistance mechanisms and predict combinatorial drug treatments in breast cancer. *Cancer Convergence*, *1*(1), 1–25. https://doi.org/10.1186/s41236-017-0007-6

Grossman, J. W., & Ion, P. D. F. (1995). On a Portion of the Well-Known Collaboration Graph. *Congressus Numerantium*, *108*, 129–131. https://doi.org/10.1016/S0304-5412(12)70429-8

G.T. Zañudo, J., Steinway, S. N., & Albert, R. (2018). Discrete dynamic network modeling of oncogenic signaling: Mechanistic insights for personalized treatment of cancer. *Current Opinion in Systems Biology*, *9*, 1–10. https://doi.org/10.1016/j.coisb.2018.02.002

Gyori, B. M., Bachman, J. A., Subramanian, K., Muhlich, J. L., Galescu, L., & Sorger, P. K. (2017a). From word models to executable models of signaling networks using automated assembly. *Molecular Systems Biology*, *13*(11), 954. https://doi.org/10.15252/msb.20177651

Gyori, B. M., Bachman, J. A., Subramanian, K., Muhlich, J. L., Galescu, L., & Sorger, P. K. (2017b). From word models to executable models of signaling networks using automated assembly. *Molecular Systems Biology*, *13*(11), 954. https://doi.org/10.15252/msb.20177651

Hawse, W. F., Sheehan, R. P., Miskov-Zivanov, N., Menk, A. V., Kane, L. P., Faeder, J. R., & Morel, P. A. (2015a). Cutting Edge: Differential Regulation of PTEN by TCR, Akt, and FoxO1 Controls CD4 + T Cell Fate Decisions . *The Journal of Immunology*, *194*(10), 4615–4619. https://doi.org/10.4049/jimmunol.1402554

Hawse, W. F., Sheehan, R. P., Miskov-Zivanov, N., Menk, A. v., Kane, L. P., Faeder, J. R., & Morel, P. A. (2015b). Cutting Edge: Differential Regulation of PTEN by TCR, Akt, and FoxO1 Controls CD4 + T Cell Fate Decisions . *The Journal of Immunology*, *194*(10), 4615–4619. https://doi.org/10.4049/jimmunol.1402554

Hawse, W. F., Sheehan, R. P., Miskov-Zivanov, N., Menk, A. v., Kane, L. P., Faeder, J. R., & Morel, P. A. (2015c). Cutting Edge: Differential Regulation of PTEN by TCR, Akt, and FoxO1 Controls CD4 + T Cell Fate Decisions . *The Journal of Immunology*, *194*(10), 4615–4619. https://doi.org/10.4049/jimmunol.1402554

Helikar, T., Kowal, B., McClenathan, S., Bruckner, M., Rowley, T., Madrahimov, A., Wicks, B., Shrestha, M., Limbu, K., & Rogers, J. A. (2012). The Cell Collective: Toward an open and collaborative approach to systems biology. *BMC Systems Biology*, *6*. https://doi.org/10.1186/1752-0509-6-96

Holtzapple, E., Telmer, C. A., & Miskov-Zivanov, N. (2020). FLUTE: Fast and reliable knowledge retrieval from biomedical literature. *Database : The Journal of Biological Databases and Curation*, *2020*, 1–17. https://doi.org/10.1093/database/baaa056

https://www.ovid.com. (n.d.).

Katz, S. M. (1987). Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, *35*(3), 400–401. https://doi.org/10.1109/TASSP.1987.1165125

King, A. D., Pržulj, N., & Jurisica, I. (2004). Protein complex prediction via cost-based clustering. *Bioinformatics*, *20*(17), 3013–3020. https://doi.org/10.1093/bioinformatics/bth351

Koschützki, D., & Schreiber, F. (2008). Centrality analysis methods for biological networks and their application to gene regulatory networks. *Gene Regulation and Systems Biology*, *2008*(2), 193–201. https://doi.org/10.4137/grsb.s702

Kumar-Jha, S., Clarke, E. M., Langmead, C. J., Legay, A., Platzer, A., & Zuliani, P. (2009). A Bayesian Approach to Model Checking Biological Systems. *Cmsb*, *2005*, 218–234.

Kvorjak, M., Ahmed, Y., Miller, M. L., Sriram, R., Coronnello, C., Hashash, J. G., Hartman, D. J., Telmer, C. A., Miskov-Zivanov, N., Finn, O. J., & Cascio, S. (2019). Crosstalk between colon cells and macrophages increases ST6GALNAC1 and MUC1-sTn expression in ulcerative colitis and colitis-associated colon cancer. *Cancer Immunology Research*. https://doi.org/10.1158/2326-6066.CIR-19-0514

Lei, X., Wang, F., Wu, F., Zhang, A., & Pedrycz, W. (2016). Protein complex identification through Markov clustering with firefly algorithm on dynamic protein – protein interaction networks. *Information Sciences*, *329*, 303–316. https://doi.org/10.1016/j.ins.2015.09.028

Liang, K.-W., Wang, Q., Telmer, C., Ravichandran, D., Spirtes, P., & Miskov-Zivanov, N. (2017). *Methods to Expand Cell signaling Models using Automated Reading and Model Checking* (pp. 145–159). Springer, Cham.

Miskov-Zivanov, N. (2015). Automation of Biological Model Learning, Design and Analysis. *Proceedings of the 25th Edition on Great Lakes Symposium on VLSI - GLSVLSI '15*, 327–329. https://doi.org/10.1145/2742060.2743765

Miskov-Zivanov, N., Turner, M. S., Kane, L. P., Morel, P. A., & Faeder, J. R. (2013a). The duration of T cell stimulation is a critical determinant of cell fate and plasticity. *Science Signaling*, *6*(300). https://doi.org/10.1126/scisignal.2004217

Miskov-Zivanov, N., Turner, M. S., Kane, L. P., Morel, P. A., & Faeder, J. R. (2013b). The duration of T cell stimulation is a critical determinant of cell fate and plasticity. *Science Signaling*, *6*(300), 1–16. https://doi.org/10.1126/scisignal.2004217

Miskov-Zivanov, N., Zuliani, P., Clarke, E. M., & Faeder, J. R. (2013). Studies of biological networks with statistical model checking: Application to immune system cells. *2013 ACM Conference on Bioinformatics, Computational Biology and Biomedical Informatics, ACM-BCB 2013*, 728–729. https://doi.org/10.1145/2506583.2512390

Mountasser, I., Ouhbi, B., & Frikh, B. (2017). Parallel Markov-based clustering strategy for large-scale ontology partitioning. *IC3K 2017 - Proceedings of the 9th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, *2*(Keod), 195–202. https://doi.org/10.5220/0006504001950202

Muilenburg, D., Parsons, C., Coates, J., Virudachalam, S., & J. Bold, R. (2014). Role of Autophagy in Apoptotic Regulation by Akt in Pancreatic Cancer. *Anticancer Research*, *34*(2), 631–637. https://doi.org/10.1016/j.physbeh.2017.03.040

Naldi, A., Hernandez, C., Levy, N., Stoll, G., Monteiro, P. T., Chaouiya, C., Helikar, T., Zinovyev, A., Calzone, L., Cohen-Boulakia, S., Thieffry, D., & Paulevé, L. (2018). The CoLoMoTo interactive notebook: Accessible and reproducible computational analyses for qualitative

biological networks. *Frontiers in Physiology*, *9*(JUN), 1–13. https://doi.org/10.3389/fphys.2018.00680

Naptali, W., Tsuchiya, M., & Nakagawa, S. (2009). Word co-occurrence matrix and context dependent class in lsa based language model for speech recognition. *International Journal of Computers*, *3*(1), 1–11.

Newman, M. E. J. (2003a). The structure and function of complex networks. *SIAM Review*, *45*(2), 167–256. https://doi.org/10.1137/S003614450342480

Newman, M. E. J. (2003b). The structure and function of complex networks. *SIAM Review*, *45*(2), 167–256. https://doi.org/10.1137/S003614450342480

Ney, H., & Vi, I. (1995). *Improved backing-off for M-gram language modeling*. 181–184.

P.Brown, V.Della Pietra, de Souza, P., J.Lai, & R.Mercer. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, *18*(1950), 467–479.

Pillich, R. T., Chen, J., Rynkov, V., Welker, D., & Pratt, D. (2017). NDEx: A community resource for sharing and publishing of biological networks. In *Methods in Molecular Biology* (Vol. 1558). https://doi.org/10.1007/978-1-4939-6783-4_13

Roberts, R. J. (2001). PubMed Central: The GenBank of the published literature. *Proceedings of the National Academy of Sciences of the United States of America*, *98*(2), 381–382. https://doi.org/10.1073/pnas.98.2.381

Saadatpour, A., Wang, R.-S., Liao, A., Liu, X., Loughran, T., Albert, P. I., & Albert, R. (2011). Dynamical and Structural Analysis of a T Cell Survival Network Identifies Novel Candidate Therapeutic Targets for Large Granular Lymphocyte Leukemia. *PLoS Computational Biology*. https://doi.org/10.1371/journal.pcbi.1002267

Sayed, K., Bocan, K. N., & Miskov-zivanov, N. (2018). *Automated Extension of Cell Signaling Models with Genetic Algorithm*. *1*, 5030–5033.

Sayed, K., Kuo, Y. H., Kulkarni, A., & Miskov-Zivanov, N. (2018). DiSH simulator: Capturing dynamics of cellular signaling with heterogeneous knowledge. *Proceedings - Winter Simulation Conference*, 896–907. https://doi.org/10.1109/WSC.2017.8247841

Sayed, K., Telmer, C. A., Butchy, A. A., & Miskov-Zivanov, N. (2018). Recipes for translating big data machine reading to executable cellular signaling models. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10710 LNCS*, 1–15. https://doi.org/10.1007/978-3-319-72926-8_1

Scardoni, G., & Lau, C. (2012). Centralities Based Analysis of Complex Networks. *New Frontiers in Graph Theory*, *2008*. https://doi.org/10.5772/35846

Schaeffer, S. E. (2007). Graph clustering by flow simulation. *Computer Science Review*, *1*(september 1969), 27–64. https://doi.org/10.1016/j.cosrev.2007.05.001

Schwab, J. D., Kühlwein, S. D., Ikonomi, N., Kühl, M., & Kestler, H. A. (2020a). Concepts in Boolean network modeling: What do they all mean? *Computational and Structural Biotechnology Journal*, *18*, 571–582. https://doi.org/10.1016/j.csbj.2020.03.001

Schwab, J. D., Kühlwein, S. D., Ikonomi, N., Kühl, M., & Kestler, H. A. (2020b). Concepts in Boolean network modeling: What do they all mean? *Computational and Structural Biotechnology Journal*, *18*, 571–582. https://doi.org/10.1016/j.csbj.2020.03.001

Siegel, P., & Massagué, J. (2003). Cytostatic and apoptotic actions of TGF-beta in homeostasis and cancer. *Nature Reviews. Cancer Cancer*, *3*(11). https://doi.org/10.1038/nrc1208

Smith, A. M., Xu, W., Sun, Y., Faeder, J. R., & Marai, G. E. (2012). RuleBender: integrated modeling, simulation and visualization for rule-based intracellular biochemistry. *BMC Bioinformatics*, *13 Suppl 8*. https://doi.org/10.1186/1471-2105-13-S8-S3

Tan, P.-N., Steinbach, M., & Kumar, V. (2007). *Introduction to Data Mining* (12(3):279–).

Telmer, C. A., Bocan, K. N., Holtzapple, E., Ahmed, Y., Butchy, A., & Hansen, C. E. (2019). Dynamic system explanation: DySE, a framework that evolves to reason about complex systems-lessons learned. *Proceedings of the Conference on Artificial Intelligence for Data Discovery and Reuse*, *Aidr*, 1–10.

Telmer, C. A., Sayed, K., Butchy, A. A., Bocan, K., Kaltenmeier, C., Lotze, M., & Miskov-zivanov, N. (2019). Computational modeling of cell signaling and mutations in pancreatic cancer. *Proc. of Artificial Intelligence for Synthetic Biology (AI4SynBio), November 2019*.

Telmer, C., Sayed, K., Butchy, A. A., Bocan, K. N., Kaltenmeier, C., Lotze, M., & Miskov-zivanov, N. (2019). Computational modeling of cell signaling and mutations in pancreatic cancer. *AI for Synthetic Biology*.

Tkachev, I., & Abate, A. (2013). Formula-free Finite Abstractions for Linear Temporal. *Proceedings of the 16th International Conference on Hybrid Systems: Computation and Control*, 283–292.

Valenzuela-escárcega, M. A., Babur, Ö., Hahn-powell, G., Hicks, T., Noriega-atala, E., Wang, X., Surdeanu, M., Demir, E., & Morrison, C. T. (2018). Large-scale Automated Reading of Scientific Cancer Literature Discovers New Cancer Driving Mechanisms. *Database*.

Valenzuela-Escárcega, M. A., Hahn-Powell, G., Hicks, T., & Surdeanu, M. (2015). A Domain-independent Rule-based Framework for Event Extraction. *Proceedings of ACL-IJCNLP 2015 System Demonstrations*, 127–132. https://doi.org/10.3115/v1/P15-4022

Vlasblom, J., & Wodak, S. J. (2009). Markov clustering versus affinity propagation for the partitioning of protein interaction graphs. *BMC Bioinformatics*, *10*, 1–14. https://doi.org/10.1186/1471-2105-10-99

von Mering, C., Jensen, L. J., Snel, B., Hooper, S. D., Krupp, M., Foglierini, M., Jouffre, N., Huynen, M. A., & Bork, P. (2005). STRING: Known and predicted protein-protein associations, integrated and transferred across organisms. *Nucleic Acids Research*, *33*(DATABASE ISS.), 433–437. https://doi.org/10.1093/nar/gki005

Wang, Q. (2016). *Model Checking for Biological Systems : Languages , Algorithms , and Applications*.

Wang, Q., Miskov-Zivanov, N., Liu, B., Faeder, J. R., Lotze, M., & Clarke, E. M. (2016). Formal modeling and analysis of pancreatic cancer microenvironment. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9859 LNCS*, 289–305. https://doi.org/10.1007/978-3-319-45177-0_18

Weeber, M., Baayen, R. H., & Vos, R. (2000). Extracting the lowest-frequency words: Pitfalls and possibilities. *Computational Linguistics*, *26*(3). https://doi.org/10.1162/089120100561719

Whitley, D. (1994). A genetic algorithm tutorial. *Statistics and Computing*, *4*(2), 65–85. https://doi.org/10.1007/BF00175354

Wu, X., Kumar, V., Ross, Q. J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Yu, P. S., Zhou, Z. H., Steinbach, M., Hand, D. J., & Steinberg, D. (2008). Top 10 algorithms in data mining. In *Knowledge and Information Systems* (Vol. 14, Issue 1). https://doi.org/10.1007/s10115-007-0114-2

Zhang, R., Vinod Shah, M., Yang, J., Liu, X., Yun, J. K., Nyland, S. B., Albert, R., & Loughran, T. P. (2008). Network model of survival signaling in T-cell large granular lymphocyte leukemia. *Proceedings of the National Academy of Sciences*, *105*(42), 16308–16313. https://doi.org/10.1201/9781439811863