Biostatistics Research
https://ojs.wiserpub.com/index.php/BSR/

Review

Biological Network Mining

Zongliang Yue[1*], **Da Yan**[2*], **Guimu Guo**[3], **Jake Y. Chen**[4]

[1]Harrison College of Pharmacy, Auburn University, Auburn, AL 36849, United States of America
[2]Department of Computer Science, College of Arts and Sciences, the University of Alabama at Birmingham, Birmingham, AL 35233, United States of America
[3]Department of Computer Science, College of Arts and Sciences, Rowan University, Glassboro, NJ 08028, United States of America
[4]Informatics Institute, School of Medicine, the University of Alabama at Birmingham, Birmingham, AL 35233, United States of America
*Indicates the authors contribute equally
E-mail: zongyue@auburn.edu

Received: 9 September 2022; **Revised:** 24 February 2023; **Accepted:** 6 March 2023

Abstract: In this survey, we explore the latest methods and trends in constructing and mining biological networks. We delve into cutting-edge techniques such as weighted gene co-expression network analysis (WGCNA), step-level differential response (SLDR), Biomedical Entity Expansion, Ranking and Explorations (BEERE), Weighted In-Network Node Expansion and Ranking (WINNER), and Weighted In-Path Edge Ranking (WIPER) from the Bioinformatics community, as well as breakthroughs in graph mining methods like parallel subgraph mining systems, temporal graph algorithms, and deep learning. To ensure a solid foundation, we provide an introductory-level overview of six well-established network types in systems biology. In addition, we offer a concise and accessible overview of strategies for network construction, including gene co-expression networks (GCNs), gene regulatory networks (GRNs), and literature-mined biomedical networks. We explain biological network mining in interdisciplinary domains, catering to both biomedical researchers and data mining experts. Our goal is to provide a comprehensive guide that doesn't require a significant time investment. We believe that these current trends will help readers become familiar with the topic and the practical applications of these tools in real-world studies.

***CCS concepts*:** mathematics of computing → graph algorithms • applied computing → bioinformatics • computing methodologies → machine learning

***Keywords*:** network, graph, mining, gene, microarray, protein, protein-protein interaction

1. Introduction

Biological networks such as gene co-expression networks (GCNs), gene regulatory networks (GRNs) and protein-protein interaction (PPI) networks have become the de facto standard in silico media for biomedical researchers to extract biomolecules of potential biological significance from experimental data, which can be further tested to discover pathways, molecular functions, and mechanisms of action that are vital for drug/vaccine discovery, disease control/prevention, etc. The biomedical community has created many well-established network analytics tools that are widely used by biomedical researchers, such as the weighted gene co-expression network analysis (WGCNA) [1] R package for weighted correlation network analysis. Moreover, new network analytic methods such as step-level differential response

(SLDR) [2], Biomedical Entity Expansion, Ranking and Explorations (BEERE) [3], Weighted In-Path Edge Ranking (WIPER) [4], Weighted In-Network Node Expansion and Ranking (WINNER) [5] and distance-bounded energy-field minimization algorithm (DEMA) [6] are being constantly developed to open new opportunities to find more insights from biological networks. For example, WIPER is a recent method that allows a node ranking method (that is usually used to find vital biomolecules) to be directly applied to rank edges (i.e., interactions between biomolecules). Another interesting breakthrough in the biomedical community is the availability of large knowledge databases such as Semantic MEDLINE Database (SemMedDB) [7] that mine predications from the vast amount of biomedical literature to allow the construction and enrichment of biological networks. Network mining has been widely applied to discover disease-specific genes [8], biomarkers [9], drug candidates [10, 11], drug side effects [12], leading to the emergence of network medicine [13].

On the other hand, the computer science community has recently made breakthroughs in graph mining and analytics, where new graph algorithms and machine learning models are emerging and have been applied to analyze biological networks to bring more insights [14, 15]. Specifically, dense and frequent subgraph mining problems were originally motivated by applications in Bioinformatics to find important substructures, but their biomedical applications are still very limited due to the prohibitive computation cost. Recent breakthroughs have been made in the field of graph mining with parallel and distributed systems being developed that allow dense and frequent subgraph mining to be more affordable, such as the G-thinker [14, 16-19] and PrefixPFM [15, 20-22] systems. These systems open a great potential for subgraph mining to be applied more widely in Bioinformatics applications. There are also new algorithms being developed to capture the temporal interactions modeled by a temporal graph, and those methods can potentially change the way that biological networks are analyzed (currently still dominantly in the static network setting) to bring new insights into the biological processes. New deep learning models are also emerging for tasks like biological function prediction over biological networks, such as graph neural networks (GNNs) that are currently being actively researched.

Given the breakthroughs in both communities, it is important to provide a timely survey of the current research status in biological network mining, as well as new recent methods in each community that can potentially be used to generate new tools. There have been related surveys in the biomedical community but they are application-driven, such as to inform biomedical researchers about the available software tools [23] or to test various methods in a particular experimental setting like single-cell gene expression experiments [24]. On the other hand, the core data mining community that studies graph mining algorithms mainly focus on methodology novelty and biological networks are merely used as motivating application (along with many others, such as online social networks that often exhibit different features than biological networks). The frequently asked question is, “Is there an interdisciplinary survey in grasping insights to inspire data mining researchers with biomedical questions and familiar biomedical researchers with the most advanced tools?”.

To address it, we designed this survey that sits in the middle ground, easily accessible to both communities: computer science researchers can gain better knowledge about the fundamentals and breakthroughs regarding biological network analysis in the biomedical context, while biomedical researchers can get the big picture of the latest breakthroughs in computer science that can be used to improve biological network analysis for their subjects of study. While it is not intended to be a comprehensive coverage of all recent works, or a deep dive into individual algorithms and methods, it is written to be brief and insightful, allowing researchers from both communities to gain a solid understanding of the fundamentals in this interdisciplinary domain. By investing minimal time, researchers can gain a big picture of recent trends and methods in biological network analysis, and ultimately, we hope to foster new research that delves into specific directions and accelerates convergence between the biomedical and data mining communities, especially in the graph mining field.

The rest of this survey is organized as follows. Section 2 familiarizes readers with the different types of biological networks that can be constructed from a diverse set of biological data. Section 3 reviews the methods to construct biological networks from experimental data as well as the biomedical literature. Section 4 then presents the various methods for prioritizing vital nodes (e.g., biomolecules), edges (e.g., interactions) and subgraphs (e.g., biomolecular structures and recurrent patterns), including case studies of recent methods such as BEERE [3] and WIPER [4]. Finally, Section 5 concludes this survey with a discussion of promising breakthroughs in computer science that could break new grounds in biological network mining research.

2. Networks in biology

Network construction is an essential step in systems biology analysis. The upstream analysis usually performs statistical analysis (e.g., data normalization and data filtering by *p*-value) in extracting the phenotype-related biomolecule candidates. Various in silico analysis techniques are invented for different biomolecules (genes, transcription factors, proteins, compounds and other regulatory molecules) yielding biomolecule-phenotype matrices. Although the relationships between the biomolecule candidates and phenotypes (e.g., diseases or specific conditions under treatments) have been revealed, it is still hard to interpret how the biomolecule candidates work systemically with biological relevance (e.g., the pathways, molecular functions, mechanisms of action). Therefore, the downstream systems biology analysis usually constructs the biological networks integrating prior knowledge and the statistical relationships between the biomolecules to enable the understanding at the higher levels of an organism, a tissue, or a cell. The rapidly emerging biological network analytic models and their applications have revealed unprecedented insights into cells, which speeds up drug discovery in complex diseases.

Regarding different aspects of biological focus, there are different types of biological networks constructed for network mining [25], such as PPI networks, GRN, GCN, signal transduction networks, metabolic and biochemical networks.

PPI networks [26] consist of the interactions information among proteins acting in concert to fulfill or maintain the biological functions of cells. Although the majority of proteins' complete sequences are discovered, their molecular function in specific diseases is not fully revealed. Protein function prediction remains to be a hurdle in computational biology. Tremendous efforts have been made to infer protein functions from their connected neighbors. Particularly, the prevalence of large-scale and high-throughput techniques enables the detection of protein-to-organism interactions. For instance, mass spectrometry [27], tandem affinity purification (TAP) [28], pull-down assays [29], yeast two-hybrid (Y2H) [30], microarrays [31] and phage display [32] are popular techniques in the past. Nowadays, several well-known datasets have been produced using the aforementioned techniques, e.g., the Tong et al. [33], Krogan et al. [34], the Munich Information Center for Protein Sequences (MIPS) [35], the Database of Interacting Proteins (DIP) [36], and Gavin et al. [37] datasets. Other than that, numerous biological databases containing PPI information have become accessible and they are mostly species-specific. Some notable databases are the Yeast Proteome Database (YPD) [38], MIPS [39], the Molecular Interactions (MINT) database [39], the IntAct database [40], DIP [36], the Biomolecular Interaction Network Database (BIND) [41], the Biological General Repository for Interaction Datasets (BioGRID) database [42], the Human Protein Reference Database (HPRD) [43], the Human Protein Interaction Database (HPID) [44] or the Drosophila Interactions Database (DroID) [45] for Drosophila. More well-known text mining-based services include the Human Annotated and Predicted Protein Interactions version 2.0 (HAPPI-2) [46], Search Tool for Interactions of Chemicals (STITCH) [47] and STRING [48] databases.

GRNs contain the direct regulatory relationships between the detected biomolecules (genes, microribonucleic acids (miRNAs) and transcript factors) with the indication of gene expression control. There are many variables in the modulation of this process, such as transcription factors [49], post-translational modifications and biomolecular associations [50]. GRNs are normally created as directed graphs to model how proteins and other biomolecules are involved in gene expression. The series of events that happen in different stages of the process were captured and they often show specific motifs and topological patterns. Data collection, data integration and analysis techniques have made it possible for GRN construction on a larger scale [51]. In terms of protein-deoxyribonucleic acid (DNA) interaction, the commonly used databases are JASPAR [52] and TRANSFAC [53], whereas post-translational modification is available in databases like Phospho.ELM [54], NetPhorest [55] and Phosphorylation Site Database (PHOSIDA) [56].

GCNs are undirected graphs. Nodes of GCNs represent genes, and edges are given by a set of node pairs representing the significant co-expression relationship among the nodes. Given a gene expression microarray where rows are genes and columns are different samples or biological conditions, GCN can be constructed with the gene pairs sharing similar expression patterns, i.e., their row values rise and fall together across samples (or conditions). Those patterns are featured by the same expression directions across different samples. GCNs are interesting since co-expressed genes are likely regulated by the same transcriptional regulatory agent, functionally related, in the same pathway, or in the same protein complex, which is of biological interest. GCNs have been used in performing network analysis using DNA microarray data, ribonucleic acids sequence (RNA-seq) data, miRNA data, etc. One of the most well-known tools for exploring co-expression modules and intramodular hub genes is WGCNA. Co-expression modules can imply pathways or cell types. The intramodular hubs can be regarded as representatives of their respective modules. Some example databases for building GCNs are Gene Expression Omnibus (GEO) [57] and Coexpressed Gene Database (COXPRESdb) [58].

Signal transduction networks usually utilize multi-edged directed graphs to reflect the protein cascades via interaction, activation and inhibition to convert an external signal or internal signal into a physiological response. The biological signal transmission is coupled with a series of molecular events, such as protein kinases catalyzing most protein phosphorylation and resulting in a cellular response. The perturbation of the cell homeostasis is caused by environmental changes or triggers different responses. Signal transduction networks exhibit common motifs and topological patterns [59] similar to GRNs. The signal transduction pathways can be found in the databases such as Microbial Signal Transduction (MiST) [60], TRANSPATH [61], etc.

Metabolic and biochemical networks [62] consist of directed edges modeling the chemical reactions in metabolic and physical processes within a cell. The edge annotations provide the details of physiological and biochemical reactions. There are several useful tools for modeling metabolisms in different organisms. In the metabolic pathways, chemical reactions occur at different time points in a cell. The enzymes play major roles as driving forces in catalyzing biochemical reactions in a metabolic network. Often, enzymes need cofactors such as vitamins to support biochemical reactions. A metabolic network consists of the pathways with information on a series of biochemical events and their correlations. Modern sequencing techniques have enabled network reconstruction using biochemical reactions in organisms from bacteria to human [1, 63]. Some popular biochemical network databases are the Kyoto Encyclopedia of Genes and Genomes (KEGG) [64], EcoCyc [65], BioCyc [66] and metaTIGER [67]. Various approaches are utilized for the pathway structure analysis of metabolic networks [68-72]. The above networks can also be integrated into a multi-omics network to provide a global and comprehensive view of the multi-level transitions in the health and disease of a cell.

Multi-omics networks (or trans-omics networks) are generated by integrating the multiple omics-layers such as genomics, transcriptomics, proteomics, epigenomics, metabolomics and/or microbiomics in molecular biology based on prior knowledge of biochemical interactions and data-driven analytics [73]. Compared to traditional approaches, omics techniques provide a more comprehensive molecular understanding of biological systems. However, integrating multiple omics platforms remains a challenge for researchers due to their inherent data differences. A multi-omics network involves the building blocks of a cell, including DNA, ribonucleic acid (RNA), protein, and metabolite, and provides more credibility in explaining the reactions in a cell than simple molecular networks such as those described before. A high-quality multi-omics integration can yield meaningful and accurate results. For example, the two distinct strategies in modeling *Escherichia coli* are flux rerouting and gene expression, which robustly control the metabolite level against genetic and environmental perturbations, respectively. Both strategies infer “metabolic regulation” from the metabolome, expression proteome, transcriptome, and metabolic flux data [74]. As another example, a combination of transcriptome, proteome, metabolome, chromatin immunoprecipitation (ChIP) analysis, and metabolic flux reveals that *Bacillus subtilis* responds to the carbon diauxic shift through two distinct adaptation modes: faster adaptation by the post-transcriptional regulation, and slower adaptation by changes in gene expression [75]. Also, the report of 48 novel regulatory pathways in the form of phosphorylation of metabolic enzymes in rat hepatoma FAO cells is enabled by the trans-omics network reconstruction of insulin action [76].

Subsequently, we present a survey of various prevalent algorithms and tools that pertain to the four primary areas of biological network analysis: building biological networks, ranking biomedical entities, ranking biomedical entity-to-entity associations, and mining significant subnetworks (Table 1). Furthermore, we provide detailed explanations of cutting-edge techniques such as WGCNA, SLDR, BEERE, and WIPER from the Bioinformatics community.

Table 1. Data summaries

Topic	Type	Algorithm/tool
Building biological network	GCN	WGCNA Partial correlation Mutual information Partial information decomposition
	GRN	SLDR Bayesian networks Decision tree ensembles Single-cell methods
	Semantic networks	Semantic interpreter (SemRep)
Ranking biomedical entities	Node ranking	BEERE WINNER ToppGene Endeavour
Ranking biomedical entity-to-entity associations	Edge ranking	WIPER Betweenness centrality Jaccard coefficient Bridgeness index Reachability index
Mining significant subnetworks	Subgraph mining	Dense subgraphs Frequent subgraph patterns
	Clustering	Biclustering

3. Building biological networks

Now that we have seen different kinds of biological networks, this section illustrates how those networks are constructed, either from experimental measurements (Sections 3.1 and 3.2) or from text mining over the biomedical literature (Section 3.3). In Sections 3.1 and 3.2, we consider networks whose nodes are genes, and whose edges are deduced from statistical inference (e.g., gene-wise correlation analysis). Section 3.1 focuses on GRN where edges are directed. These gene networks are typically built using datasets from high-throughput technologies such as DNA microarray, RNA sequencing or single-cell sequencing.

3.1 *GCN construction*

This subsection first introduces a famous tool called WGCNA to construct weighted networks in Section 3.1.1, and then briefly covers other techniques that decide if there is an undirected edge between two genes in Section 3.1.2.

3.1.1 *Weighted correlation network analysis by WGCNA*

One of the favorite tools for building GCN mining is WGCNA [77]. As an R package, WGCNA provides a list of R functions to perform weighted correlation network analysis. WGCNA here is the acronym of weighted correlation network analysis, not just weighted gene co-expression network analysis, since the input data can be substituted for many other omics datasets such as DNA methylation, miRNA, and functional magnetic resonance angiography (MRA) data, as long as the goal is to capture the correlation patterns among biomolecules. We focus on the gene correlation patterns across microarray samples in this subsection. WGCNA can discover gene clusters (aka gene modules) based on correlation. There are three purposes for extracting such gene clusters: (1) to find the module eigengene or an intramodular hub gene summarizing the clusters, (2) to find the module-to-module associations and the module-to-trait associations, where traits are from external samples, and (3) to compute membership measures in each module. In the sequel, we consider GCN construction, module detection, and relating genes/modules to clinical traits, and we describe WGCNA's approach for them.

Correlation network construction. The network is constructed from DNA microarray data which can be regarded as an $n \times m$ matrix $X = [x_{i\ell}]$ where each row i corresponds to a gene (or a node in the GCN to be constructed) and each column ℓ corresponds to a sample. If we use the block-matrix notation to list the rows $X = [x_{i\ell}] = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^T$, then the i-th row $\mathbf{x}_i$ is called the *node profile* of the i-th gene, which records its m sample measurements.

In WGCNA, a network is constructed as an adjacency matrix A [78], which is a symmetric $n \times n$ matrix with each entry a_{ij} taking a value in [0, 1] encoding the network connection strength between genes (aka nodes) i and j. The calculation of a_{ij} is based on the (Pearson) correlation coefficient between the expression profiles of nodes i and j, denoted by $cor(\mathbf{x}_i, \mathbf{x}_j)$, though it can be replaced with other more robust correlation measures such as the Spearman's rank correlation and the biweight midcorrelation [79] which is recommended by WGCNA.

Note that correlation takes a value in [-1, 1], but each adjacency a_{ij} takes a value in [0, 1], and a_{ij} is actually a function of $cor(\mathbf{x}_i, \mathbf{x}_j)$. The calculation of the adjacency matrix is based on the type of network. In an unsigned network, $a_{ij} = |cor(\mathbf{x}_i, \mathbf{x}_j)|^\beta$, where the power $\beta \geq 1$, which performs soft thresholding (i.e., to bias towards high correlation) to better model the scale-free topology of biological networks such as GCNs and PPI networks. Specifically, if we define the connectivity of a node i as $\Sigma_j\, a_{ij}$ (a concept similar to vertex degree in an unweighted graph), then the portion of nodes in different connectivity levels follow a power-law distribution with only a few high-connectivity nodes. In a signed network where we only want positively correlated genes to be in the same module, we have $a_{ij} = |0.5 + 0.5 \cdot cor(\mathbf{x}_i, \mathbf{x}_j)|^\beta$ to suppress negatively correlated gene pairs.

Thresholding is essential to prune gene pairs with low adjacency (i.e., to set a_{ij} to 0) to avoid generating a complete graph or overly dense network. A user can control the threshold parameter, for example, by using the criterion of approximate scale-free topology [80]. The idea is to choose the power parameter β such that the log-log scatter plot of node frequency vs. connectivity is approximately linear, as measured by R^2 where R is the correlation. As we increase β, the value of R^2 rises to a high value indicating that our network begins to satisfy the scale-free topology, and this is where we choose the value for β. If we increase β further, R^2 usually plateaus or even drops slightly. Another criterion is the mean connectivity among all nodes which drops when β increases, and we do not want the mean connectivity to be too low as we will miss too many genes with low connectivity values. That is why we typically use the smallest β such that R^2 has risen to a high value.

Module detection. In a GCN, genes are usually clustered into the so-called "modules" which are highly connected subgraphs. Genes in a module are highly correlated and thus tend to have a similar function or participate in biological processes together with interactions.

WGCNA identifies gene modules based on unsupervised clustering without any prior knowledge. A user has an option to choose the method of module detection. Hierarchical clustering is the default method, where the branches of the resulting dendrogram correspond to modules, and the available branch cutting methods can be directly used in the module identification, such as the constant-height cut or two dynamic-branch cut methods as introduced in [81], with the second one improving the detection of outlying members of each cluster by including a Partitioning Around Medoids (PAM)-like step.

A representative is often identified for each module to summarize the node profiles therein (which are gene expression profiles here), and it is much more computationally efficient to focus analysis on the level of modules (or their representatives). This can be regarded as a network-based data reduction method, in which the mappings are from modules (or their representatives), instead of nodes, to the clinical trait of a sample. Meanwhile, this method can also alleviate the multiple testing problem (aka multiple comparisons fallacy) in statistical analysis.

Two popular module representative definitions include eigengene and the most highly connected intramodular hub gene, as we review next. Module eigengene is defined as the first principal component of the corresponding module's expression matrix. The eigengene is not an actual gene expression profile in its module but can be regarded as a weighted average expression profile. Eigengene calculation in WGCNA comprises the imputation of missing values. Alternatively, one can use the intramodular connectivity to measure and filter the most highly connected intramodular hub gene as the module representative, which is an actual gene in its module in contrast to eigengene. The intramodular hub genes are highly correlated with the module eigengenes.

Many DNA microarrays report expression levels of several thousand (or more) distinct genes (or probes), which can be computationally challenging if only the high-complexity hierarchical clustering is used. WGCNA provides a block-wise module detection approach to address this scalability problem, where the first step is to pre-cluster nodes into large clusters called blocks, using a variant of *k*-means clustering which is efficient. Next, hierarchical clustering is applied only within each smaller block to find modules therein. Finally, an automatic module merging step is performed where modules from different blocks whose eigengenes are highly correlated get merged.

Sometimes it is interesting to find consensus modules present in different networks (e.g., most or all networks in a dataset). Here, two nodes should be connected in a consensus network only if most or all of the input networks agree on that connection, i.e., the similarity between two nodes can be defined as the minimum (or a suitable quantile) of the input network similarities. WGCNA also supports a block-wise approach to calculate consensus modules.

Relating modules to external data. Given a microarray sample trait $\mathbf{t}$ which is also measured as a vector with m components that correspond to the columns of the data matrix X, then we can measure how significant a gene $\mathbf{x}_i$ is related to the trait $\mathbf{t}$ by defining a gene significance (GS) measure. The module significance can be measured by the average GS across the module genes, or by the module eigengene as $\mathbf{x}_i$ to calculate GS. Modules with high trait significance may imply potential pathways associated with the sample trait.

WGCNA supports module and gene selection by applying a GS measure to assign a non-negative number to each gene. The GS measure can be the absolute value of the correlation between the i-th gene and the sample trait, i.e., $|cor(\mathbf{x}_i, \mathbf{t})|$, by default. It can be substituted by p-value from either a correlation test or a regression, i.e., -log(p-value) for assessing the statistical significance between $\mathbf{x}_i$ and $\mathbf{t}$. The higher the GS of a module/gene is, the more biologically significant the module/gene is. For instance, a GS measure could imply pathway membership importance in functional enrichment analysis or the knockout essentiality in gene-knockout.

Application: With the shift towards single-cell data, WGCNA has the potential to expand its applications for identifying novel prognostic models in cancer studies for survival prediction [82]. Furthermore, WGCNA can be combined with machine learning techniques to discover immune-related biomarkers for complex diseases, such as rheumatoid arthritis [83]. WGCNA can also be used in conjunction with other analytical methods, such as gene set variation analysis (GSVA), to explore potential biomarkers for novel preventive and pharmacological approaches in surgical samples using total RNA data [84].

3.1.2 *Other methods to infer undirected edges*

Next, we consider a few other methods that decide if an undirected edge exists between two genes in the resulting GCN.

Partial correlation. Besides the correlation metrics previously introduced in WGCNA, we can also use partial correlation to test whether or not two genes have highly correlated expression patterns. In probability theory and statistics, partial correlation (*pCorr*) measures the degree of association between two random variables subject to the elimination of a set of controlling random variables. In a GCN, the *pCorr* of the pairwise genes i and j, $pCorr(\mathbf{x}_i, \mathbf{x}_j)$, is defined by treating all the other genes $X - \{\boldsymbol{x}_i, \boldsymbol{x}_j\}$ as confounding variables (here, we abuse the notation X to mean its row set $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$). Under this definition, inferring if there is an edge between genes i and j is equivalent to testing the following hypothesis:

$$H_0\text{: } pCorr(\mathbf{x}_i, \mathbf{x}_j) = 0 \quad \text{vs.} \quad H_1\text{: } pCorr(\mathbf{x}_i, \mathbf{x}_j) \neq 0.$$

In this way, the presence of an edge between genes i and j implies that a correlation exists regardless of which other genes are being conditioned on. The Fisher transformation (aka Fisher z-transformation) can be used to test these hypotheses to calculate the p-value for thresholding (e.g., statistical significance was set to be 0.05). Since we need to perform this hypothesis testing over all pairs of genes, adjustment for multiple testing correction of the p-values is necessary, e.g., by using the Benjamini-Hochberg method [33].

Mutual Information (MI). Besides correlation, MI can also be used to decide the presence of an edge between genes i and j. Algorithm for the Reconstruction of Accurate Cellular Networks (ARACNE) [85] is a conventional information-theoretic network approach based on MI. Given two gene profiles $\mathbf{x}_i$ and $\mathbf{x}_j$ of two genes i and j in a DNA microarray $X = [x_{i\ell}]$, their MI, denoted by $I(\mathbf{x}_i, \mathbf{x}_j)$, can be computed by the following equation using the value distributions of genes i and genes j in X, denoted by $p(x_{i\ell})$ and $p(x_{j\ell})$, respectively, plus the joint distribution $p(x_{i\ell}, x_{j\ell})$.

$$I\left(x_i, x_j\right) = \sum_{x_{i\ell} \in x_i} \sum_{x_{i\ell} \in_j} p\left(x_{i\ell}, x_{j\ell}\right) \log_2 \frac{p\left(x_{i\ell}, x_{j\ell}\right)}{p\left(x_{i\ell}\right) p\left(x_{j\ell}\right)},$$

where $p(x_{i\ell})$ is fit from the elements of $\mathbf{x}_i$, and $p(x_{i\ell}, x_{j\ell})$ is fit from the column-aligned element pairs in $\mathbf{x}_i$ and $\mathbf{x}_j$, using kernel density estimation with a Gaussian kernel. MI $I(\mathbf{x}_i, \mathbf{x}_j)$ indicates the degree of dependency between two genes i and j.

Next, ARACNE applies the Data Processing Inequality (DPI) to eliminate indirect interactions inferred from the remaining interactions in the GCN network. DPI states that if genes i and k interact only through a third gene, gene j, (i.e., gene $i \leftrightarrow \ldots \leftrightarrow$ gene $j \leftrightarrow \ldots \leftrightarrow$ gene k and no alternative path exists between genes i and k), then $I(\boldsymbol{x}_i, \boldsymbol{x}_k) \leq min\ \{I(\boldsymbol{x}_i, \boldsymbol{x}_j), I(\boldsymbol{x}_j, \boldsymbol{x}_k)\}$. ARACNE removes every such indirect edge $i \leftrightarrow k$. To compensate for errors in estimating the probabilities when computing MI, a tolerance parameter *eps* is used so that edge $i \leftrightarrow k$ is removed only if min $\{I(\boldsymbol{x}_i, \boldsymbol{x}_j), I(\boldsymbol{x}_j, \boldsymbol{x}_k)\} - I(\boldsymbol{x}_i, x_k) > eps$. The tolerance *eps* is suggested to be set between 0.1 and 0.2.

Another MI-based approach is Context Likelihood of Relatedness (CLR) [86, 87]. The difference from ARACNE is that CLR considers the background distribution of MI values. The background correction is to reduce false positives in detection interactions due to data noise. CLR derives a modified (positive-only) z-score for $I(\boldsymbol{x}_i, \boldsymbol{x}_j)$ with respect to column j in the MI matrix:

$$z_i\left(x_i, x_j\right) = max\left(0, \frac{I\left(x_i, x_j\right) - \mu_i}{\sigma_i}\right),$$

where μ_i and σ_i are the mean and standard deviation of MI values $I(\mathbf{x}_i, \mathbf{x}_j), j = 1, \ldots, n$. Note that $z_i(\mathbf{x}_i, \mathbf{x}_j)$ may not equal $z_j(\boldsymbol{x}_j, \boldsymbol{x}_i)$, which is OK if we are constructing a directed GRN by assigning $z_i(\mathbf{x}_i, \mathbf{x}_j)$ as the weight of edge $i \rightarrow j$. For an undirected GCN, we can use the following weight for edge $i \leftrightarrow j$ instead.

$$z\left(x_i, x_j\right) = \sqrt{z_i^2\left(x_i, x_j\right) + z_j^2\left(x_j, x_i\right)}.$$

Partial Information Decomposition (PID). In information theory, another interaction metric is PID [88]. PID quantifies the information from several inputs individually (unique information), redundantly (shared information) or only jointly (synergistic information). In our case, PID considers the information flowing from the source genes i and j to another target gene k, which is partitioned into redundant, synergistic, and unique information:

$$I(k; i, j) = Synergy(k; i, j) + Unique_j(k; i) + Unique_i(k; j) + \text{Redundancy}(k; i, j),$$

where the PID term Redundancy(k; i, j) is the portion of information about gene k derived from either gene i or gene j alone; $Unique_j(k; i)$ is the portion of the information contributed uniquely by gene i (when the other gene is j); and $Synergy(k; i, j)$ is the portion of the information that is only yielded by knowledge of both gene i and gene j together. All these terms can be computed from the probability distribution of gene expression values that can be estimated from gene profiles, using proper data discretization and an MI estimator. Two methods for discretization, along with four MI estimators, were tested in [89], which uses PID to identify regulatory relationships between genes for single-cell data. Their inference algorithm, called PID and context (PIDC), uses the ratio of unique information between gene i and gene j across every of the third (target) gene k in $X - \{\boldsymbol{x}_i, \boldsymbol{x}_j\}$, giving rise to the following definition of Proportional Unique Contribution (PUC):

$$u_{i,j} = \sum\nolimits_{k\in[n]-\{i,j\}} \frac{Unique_k\left(i;j\right)}{I\left(i;j\right)} + \sum\nolimits_{k\in[n]-\{i,j\}} \frac{Unique_k\left(j;i\right)}{I\left(j;i\right)},$$

where $[n]= \{1, 2, \ldots, n\}$, and $I(i; j) = I(j; i)$ is the MI between gene i and gene j. Note that the difference between $I(i; j)$ and $Uniquek(i; j)$ is equal to the redundancy between all three genes i, j and k, meaning that the ratio $Unique_k(i; j)/I(i; j)$ represents the proportion of MI contributed from the unique information between X and Y, as opposed to redundant information between all three genes. Here, we consider both unique information directions to make the resulting network undirected, though one direction is OK if we allow a directed graph like in a GRN. To find a threshold for defining an edge, we should account for the difference in the distributions of PUC scores between genes. The CLR method introduced before can be similarly adapted to the PUC setting, where we estimate $F_i(u_{i,j})$, the cumulative distribution function of all the PUC scores involving gene i, from $u_{i,1}, u_{i,2}, \ldots, u_{i,n}$, assuming either a Gamma or Gaussian empirical probability distribution. The confidence of an edge $i \leftrightarrow j$ can then be calculated as $c = F_i(u_{i,j}) + F_j(u_{j,i})$. By integrating the distribution of PUC score for a particular gene, rather than simply keeping edges that are ranked the highest across all genes, PIDC is designed for detecting the most important set of inferred interactions per gene.

3.2 *GRN mining*

In a GRN, nodes represent genes and edges represent interactions between genes (e.g., direct physical connections or indirect regulation). Like a GCN, a GRN is also inferred from a data matrix $X = [x_{il}] = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n)^T$ where rows correspond to genes and columns correspond to samples (e.g., single-cell transcriptomes). Like GCN, a GRN can be inferred using metrics like correlation and MI as described in Section 3.1, but edge directions and the allowing of an asymmetric adjacency matrix bring the possibility of other methods, such as regression-model-based association measures [90] which can capture non-linear relationships. We next review a few additional techniques to construct a GRN.

3.2.1 *SLDR*

The highly-connected gene regulators, or “hubs” of gene regulations in the GRN, are covered reasonably well in the current databases. However, low-connectivity regulators, or “de-centric nodes” in the GRN regulatory, are poorly covered as the chances for randomly observing their activities are low.

SLDR [2] is designed to fill this gap by identifying “de-centric” nodes and their “rare” activation/inhibition genetic regulatory relationships. SLDR takes advantage of functional genomics data of the same species under different perturbations in different biological conditions. Each de-centric target gene is modeled as being controlled by a small number of at most N binary-state regulators, leading to $\leq 2^N$ observable states for the target. In other words, if we sort the RNA expression values of gene profile $\mathbf{x}_i$, we will observe $\leq 2^N$ “jumps” in the sorted value sequence.

The selection of de-centric target genes is conducted in five steps. (i) Firstly, for each gene i, the RNA expression values in its gene profile $\mathbf{x}_i$ are sorted to get an expression sequence $v_1, v_2, \ldots, v_m$, and the average value of its difference sequence $(v_2 - v_1), (v_3 - v_2), \ldots, (v_m - v_{m-1})$ is computed using the difference of the maximum expression and the minimum expression of gene i as $\Delta = \frac{v_m - v_1}{m-1}$. (ii) Secondly, for each gene i, SDLR counts the number of differences $(v_t - v_{t-1})$ in the difference-sequence that passes Δ_i as the cutoff threshold. We call such a difference as a jump, and the cutoff is expected to be useful in identifying jumps since most differences $(v_t - v_{t-1})$ are expected to be close to 0. (iii) We then sort the number of jumps for all genes into a sequence $k_1, k_2, \ldots, k_n$, and define $k_{max} = max\ \{k_1, k_2, \ldots, k_n\}$. (iv) To eliminate fake jumps with small differences that got counted to inflate the jump numbers of individual genes, a new cutoff threshold is computed for each gene i as $\Delta_{max} = \frac{v_m - v_1}{k_{max} - 1}$ which is larger than Δ and so better captures the $\leq k_{max}$

jumps that contribute mainly to the cumulative expression changes ($v_m - v_1$). (v) Finally, the number of binary-state genetic regulators N of each target gene i is inferred according to $2^N = k + 1$, where k is the number of jumps identified from the new cutoff threshold Δ_{max}.

Pearson correlation is applied to identify the genetic regulatory relationships among genes. Given a target gene i, we determine if gene j is a regulator of gene i by considering two expression sets extracted from the gene profile $\mathbf{x}_j$: (1) low expression set which consists of the columns with the lowest 20% expression values in $\mathbf{x}_j$; (2) high expression set which consists of the columns with the highest 20% expression in $\mathbf{x}_j$. We only deem an activation or inhibition to exist if (a) the correlation of gene i and gene j is close to 0 in the low expression set (since gene j is not active and should not dominate the impact on gene i), and meanwhile, (b) the correlation of gene i and gene j is close to ±1 in the high expression set (since gene j is active and should impact gene i). Now consider the rest 60%m columns after ruling out the low and high expression sets, which we call as the middle expression set. SLDR additionally requires that there is no significant change of expression (larger than $\Delta = \frac{v_m - v_1}{m-1}$)) for the target gene i in the middle expression set, or we reject the edge $j \rightarrow i$. This is because a big expression change in the middle expression set signals the existence of other potentially more dominant regulators. SLDR rejects $j \rightarrow i$ if the difference between the largest and the lowest expression of gene i in the 60%m columns of the middle expression set is $\geq \Delta \cdot 60\%m$.

3.2.2 *Other methods to infer directed edges*

Bayesian networks. A Bayesian network is a probabilistic graphical model to represent a set of variables and their conditional dependencies via a directed acyclic graph (DAG). In our GRN, each node corresponds to a gene i whose expression value (i.e., variable) is characterized by a conditional probability distribution that specifies the likelihood of obtaining that expression value for gene i given the expression values of its parent nodes (i.e., regulator genes).

Gaussian Bayesian Networks (GBNs) are commonly applied to reconstruct networks based on the continuous values of gene expressions. In a GBN, the joint distribution over all variables is assumed to follow a multivariable normal distribution, while the local conditional distributions are linear regression models where the parent nodes are used as explanatory variables. The network structure can be learned from the gene expression data X using a structure learning algorithm that uses the Bayesian Information Criterion (BIC) score to guide the network inference process.

Decision tree ensembles. GEne Network Inference with Ensemble of Trees (GENIE3) is developed based on a tree-based method to reconstruct GRNs, and it has been reported to be the best in the Dialogue for Reverse Engineering Assessments and Methods 4 (DREAM4) In Silico Multifactorial challenge [86]. GENIE3 decomposes the inference of a GRN into n subtasks, one for each gene i as the target gene, where the goal is set to identify the subset of other genes whose expression profiles are the most predictive of the expression profile of gene i.

Each subtask for a target gene i trains a tree ensemble model with each sample as a training data tuple, gene i's expression as the target for regression, and the expressions of the other ($n - 1$) genes as the input features. Two tree ensemble models are considered for each subtask: a random forest regressor and an extremely randomized tree regressor. Only binary node split is adopted, and at each node split, $k = \sqrt{n}$ features (or regulator gene candidates) are selected at random to choose the best split feature and its split condition. Each model produces a ranked gene list for screening regulator candidates of a target gene i. Ranks are assigned based on weights calculated as the sum of the total variance reduction of the output variable due to the split. Hence, the importance of predicted regulatory links between the target gene and a list of input gene can be measured.

Other single-cell methods. Single-Cell rEgulatory Network Inference and Clustering (SCENIC) [91] is an approach for simultaneous GRN reconstruction and cell-state identification from single-cell RNA-seq data. We now introduce the SCENIC workflow as follows. First, the identification of sets of genes (aka modules) that are co-expressed with transcription factors is accomplished with GENIE3. These modules may include many false positives and indirect, as they are derived solely depending on co-expression. To identify putative direct-binding targets, the cis-regulatory motif enrichment analysis is applied to each co-expression module to borrow information from a pre-built database RcisTarget [92] to identify enriched transcription factor binding motifs in the identified co-expression modules. To remove indirect target genes without motif support, those modules with significant motif enrichment of the correct upstream regulator are retained and pruned. Top-ranked genes for each motif are treated as the regulons, and each transcription-regulon combination is assigned in the edge list to form the network.

SCODE [93] enables GRN reconstruction for single-cell data via regulatory dynamics using ordinary differential equations (ODEs). Particularly, the transcription factors' (TFs') expression dynamics are described using linear ODEs:

$$d\mathbf{x} = A\mathbf{x}\, dt,$$

where $\mathbf{x}$ is a vector of length n (n is the number of TFs) that denotes the expression of TFs, and A is an $n \times n$ square matrix that denotes the regulatory network among TFs. Given an $n \times m$ expression matrix X where m is the number of cells, and each cell is associated with a pseudo-time estimated using Monocle [94], the TF regulatory network is inferred by optimizing A such that the ODE can successfully describe the observed expression data. A scalable algorithm optimizes A by integrating the transformation of linear ODEs and linear regression.

3.3 *Literature-mined semantic networks*

Besides experimental measurements, we can also construct various kinds of biological networks by mining from the biomedical literature, which we call as **literature-mined semantic networks**. The constantly increasing published works have allowed biologists to learn from the past and build the concept-level knowledge-based networks using data mining and natural language processing. One of the largest knowledge-based databases is SemMedDB [95] which currently stores over 97 million semantic predications (subject-predicate-object triples) obtained by processing over 29 million PubMed citations. To build the database, Semantic MEDLINE utilizes the SemRep natural language processing system [96] to extract semantic relationships between concepts found in the Unified Medical Language System (UMLS) Metathesaurus [97], by inspecting every sentence in every PubMed citation.

Here, semantic relationships are represented as predications, a formal representation having a predicate and arguments. For example, the predication "Genes AFFECTS Circadian Rhythms" was extracted from the sentence "Clock genes are the genes that control circadian rhythms in physiology and behavior." A SemRep predication has UMLS Metathesaurus concepts as arguments and a UMLS Semantic Network relation as a predicate. The ULMS Semantic Network stipulates relationships between concepts, stated in terms of the semantic types assigned to Metathesaurus concepts. Semantic MEDLINE provides enhanced access to biomedical research literature by combining PubMed document retrieval, semantic relationships, and automatic summarization.

Another popular text mining tool is PubTator [98]. PubTator is a web-based system for facilitating biocuration. PubTator advances the feature of a PubMed-like user interface. To ensure the quality of its automatic results, it incorporates several challenge-winning text mining algorithms. Compared to a formal evaluation using two external user groups, PubTator improves manual curation in terms of both efficiency and accuracy.

Both SemMedDB and PubTator have made it much easier nowadays for biomedical researchers to build high-quality and comprehensive literature-mined semantic networks.

4. Analyzing biological networks

Next, we investigate how to analyze the constructed biological networks. One common analysis is to find statistically significant nodes (e.g., candidate genes of a disease), edges (e.g., candidates of regulations) and subgraphs (e.g., candidates of molecular complexes and biological modules), which can be prioritized in experimental studies and verifications of their actual biological significance to identify unknown pathways, to speed up drug discovery, etc. This section reviews the methods of identifying significant nodes, edges and subgraphs.

4.1 *Ranking biomedical entities*

4.1.1 *BEERE*

We demonstrate how to prioritize network nodes using the random-walk based approach BEERE [3]. BEERE modifies two node scoring algorithms, PageRank [99] and network propagation [100], for node ranking.

BEERE also can expand an existing biological network with additional text-mined nodes and edges (e.g., semantic predications in SemMedDB [7]) for further score adjustment, giving a ranking of the so-called "biomedical entities", which include genes/proteins, biomedical terms, or their combinations.

Node ranking by modified PageRank. PageRank [99] was invented by Larry Page and Sergei Brin, founders of Google, to rank webpages. Specifically, the web is modeled as a graph where webpage i is linked to webpage j if there is a hyperlink to webpage j in the content of webpage i. A user is assumed to be surfing the web following hyperlink clicks with probability α, while randomly jumps to an arbitrary webpage with probability $(1 - \alpha)$. Here, α is called the damping factor. Assuming there are n webpages in total, then the PageRank for any node i is iteratively updated as follows until convergence:

$$PR_i^{(t)} = \frac{1-\alpha}{n} + \alpha \cdot \sum_{j \in \Gamma_i^{in}} \frac{PR_j^{(t-1)}}{\left|\Gamma_j^{out}\right|},$$

where t is the iteration number, $PR_i^{(t)}$ is the PageRank of node i in iteration t, Γ_i^{in} $\left(\text{resp. } \Gamma_i^{out}\right)$ is the set of inbound neighbors (resp. outbound neighbors) of node i. PageRank gives a uniform initial value $1/n$ to every node so that the jump back probability to each node i is $PR_i^{(0)} = 1/n$. The above update equation basically says that with probability $(1 - \alpha)$, a random jump gets to the current node i from some other node; and with probability α, the current node is reached from an inbound neighbor j, whose PageRank $PR_j^{(t-1)}$ was evenly distributed to all its $\left|\Gamma_j^{out}\right|$ outbound neighbors.

BEERE modifies PageRank to incorporate the rich edge weight information in a biological network. Let us denote the weight of an edge (i, j) by $w(i, j)$. In BEERE, $w(i, j)$ can be either assigned by PPI confidence scores generated from HAPPI-2 (the human-annotated and predicted protein interactions database) [46], or the so-called Relation Density Score (RDS) calculated from the semantic predications in SemMedDB [7] based on p-value derived from hypergeometric test to adjust for the frequency difference of different predicates.

Instead of using uniform initial scores (which equals the jump-back prior probabilities), BEERE uses the so-called "relevance score" (RS) developed in [8] to initialize $PR_i^{(0)}$:

$$PR_i^{(0)} = k \times In\left(\sum_{j \in \Gamma_i} w(i,j)\right) - In\left(\left|\Gamma_i\right|\right),$$

where RS is considering a PPI network with undirected edges and thus Γ_i means the neighbors of node i. Here, RS is taking logarithm over $\frac{\left[\Sigma_{j \in \Gamma_i} w(i,j)\right]^k}{\left|\Gamma_i\right|}$, which is promoted if a protein i has high-confidence interactions $w(i, j)$. RS introduces an empirical constant k ($k = 2$ by default) to control the RS distribution kurtosis, to reinforce high values of weight summation, so that if a protein i has more high-confidence interactions with other proteins, then its RS (or interaction score) will be higher. Since $\frac{\left[\Sigma_{j \in \Gamma_i} w(i,j)\right]^k}{\left|\Gamma_i\right|}$ can be less than 1 causing $PR_i^{(0)} < 0$, BEERE forces such negative $PR_i^{(0)}$ to be 0 which is reasonable as these proteins are weakly relevant and thus are assumed to have a zero jump-back probability.

After setting the initial weights to the bioentities, a PageRank-like iterative update method [101] is applied to evaluate bioentities importance globally. Here, we take the edge weights into consideration, by distributing the PageRank of a node i to its neighbors proportionally to the outbound edge weights, rather than evenly as in PageRank:

$$PR_i^{(t)} = (1-\alpha) \cdot PR_i^{(0)} + \alpha \cdot \sum_{j \in \Gamma_i^{in}} \frac{w(j,i) \cdot PR_j^{(t-1)}}{\sum_{\iota \in \Gamma_j} w(j,\iota)}.$$

Node ranking by network propagation. In the approach of network propagation, each node is initialized with a certain amount of fluid; then, a diffusion process is iteratively conducted where the fluid of a node flows towards its neighbors with an amount proportional to the outbound edge weights. Similar to PageRank, a damping factor α kicks in here: in each iteration, a node retains $(1 - \alpha)$ portion of fluid, and flows the rest α portion to its neighbors. This gives rise to the following update formula (we reuse the notation PR as the score, but please think about it as the amount of fluid):

$$PR_i^{(\iota)} = (1-\alpha)\cdot PR_i^{(\iota-1)} + \alpha\cdot\sum\nolimits_{j\in\Gamma_i^{in}} \frac{w(j,i)\cdot PR_j^{(\iota-1)}}{\sum\nolimits_{\iota\in\Gamma_j} w(j,\iota)},$$

where $(1-\alpha)\cdot PR_i^{(\iota-1)}$ is the amount of fluid retained by node i, and $\alpha\cdot\sum\nolimits_{j\in\Gamma_i^{in}} \frac{w(j,i)\cdot PR_j^{(\iota-1)}}{\sum\nolimits_{\iota\in\Gamma_j} w(j,\iota)}$, is the total amount of new incoming fluid contributed by neighbors. Interestingly, the only difference from PageRank is that the first term was $(1-\alpha)\cdot PR_i^{(0)}$ but now it is $(1-\alpha)\cdot PR_i^{(\iota-1)}$ and changes with iterations.

Evaluating the significance of ranking scores. For each bioentity (or node), the statistical likelihood that its score is generated by a random chance is estimated using a statistical significance test to obtain its associated *p*-value. The ranking scores first go through a log2-based transformation. Let Mo be the mode of the score distribution frequency histogram, let M be the median of the score distribution, and let IQR be the interquartile range [102] ($Q_3 - Q_1$, or the difference between 75th and 25th percentiles) of the score distribution. We can use 0.2×IQR to determine the bin size of the score histogram.

The score distribution is then checked to see if it has a bell shape that can be captured by a normal distribution (e.g., as is the case for Non-Small Cell Lung Cancer network obtained from the Human Annotated and Predicted Protein Interactions (HAPPI) subsets of the 4-star rank level and above [103]), which is judged by testing whether the difference between Mo and M falls within 0.5×IQR. If so, the *p*-value is computed assuming a normal distribution with the mean being M and the standard deviation being IQR/1.34 (as Q_3 and Q_1 are at -0.67σ and 0.67σ, respectively).

Otherwise, i.e., when the difference between Mo and M is greater than 0.5·IQR as in a scale-free network is expected, it is difficult to expect the distribution shape and *p*-value is then calculated empirically as a ranking-based percentile.

Network expansion. After the node scores are computed, we can expand the network to incorporate and rank new entities from other data sources. The network can be extended by retrieving undirected neighbors from a data source via the shortest paths from the seed entities.

Another more sophisticated approach is to use network propagation again with $\alpha = 0$, but initially, new entities have zero amount of fluid. The process can be halted after a few iterations when most of the fluids are still close to the seed nodes. This approach is desirable since a node that is not a direct neighbor of some seed node but is not too distant from many seed nodes (i.e., well connected through many short paths) will be ranked high; the process also automatically deprecates paths that pass highly connected or hub nodes, because the fluid will diffuse to the many neighbors of the hub, and only a small share will be allocated to each neighbor. An alternative way to halt the process is to use a random walk with restart (RWR, i.e., $\alpha > 0$) so that at each step of the propagation process, with some fixed probability the fluid at a node returns to the original source node rather than continues propagation. This version advances the propagation process to a steady state even when diffusion is restricted to the local neighborhood. Note that with a damping factor, long walks are unlikely since the amount of fluid that reaches a node decreases exponentially on the basis of its distance from the source.

Application. BEERE is a versatile tool for expanding multi-domain networks [104]. For example, to construct expanded semantic interaction networks from seeded genes, key terms from PAGER were combined with the lists of differentially regulated genes and used as input for BEERE. The resulting networks were integrated with long non-coding RNAs (lncRNAs) and their proximal genes to identify the key pair relationships, which enabled the construction of lncRNA to messenger RNA (mRNA) interaction networks. BEERE can also be used to statistically evaluate the relevance of terms [105], which is an important application. For example, to validate the keywords of enriched pathways from a differentially expressed gene list in a particular disease such as "melanoma", users can leverage BEERE to assess the significance of the terms in the disease context. This can help remove biases and refine the list of relevant pathways from a semantic perspective. Furthermore, BEERE can be easily incorporated into any pipeline for complex disease gene prioritization analysis [106].

4.1.2 *WINNER*

WINNER is a new software tool that can help characterize and prioritize disease genes from this project and generally for other cancer studies [5]. This tool allows researchers to process molecular interaction network data to create an expanded network where nodes are ranked by their statistical relevance within the network. The network where

WINNER can apply generically refers to GRN, PPI network, metabolic network, pathways, or subcomponents of the networks as network modules. Prior to the work, researchers generally use selected statistical properties, e.g., centrality score for gene nodes, or biological properties, e.g., transcriptional factors or cell-surface receptors, to prioritize genes [107]. WINNER provides a knowledge-agonstic approach to ranking genes based on network properties learned from the entire network architecture. In WINNER, two types of statistics are available for users to assess the robustness of the ranked results: 1) the node-expansion p-value, which evaluates the statistical significance of incorporating "non-seed" molecules into the original biomolecular interaction network consisting of "seed" molecules and molecular interactions; and 2) the node-ranking p-value, which assesses the relative statistical significance of each node's contribution to the overall network architecture.

WINNER's network-based ranking result is robust, which is measured by conducting several network permutation experiments involving noise spiking. In WINNER, the node degree-preservation randomization of the gene network yielded normally distributed ranking scores, surpassing those generated by other gene network randomization techniques. Additionally, WINNER-ranked genes had a stronger association with disease biology compared to existing methods, such as PageRank.

WINNER upstream ranking and expansion can identify genes that are missing from previously curated disease gene networks. For example, curated genes in the KEGG chronic myeloid leukemia (CML) pathways were collected and distributed into layers. Correlation among WINNER (WN), Igenunity Pathway Analysis (IPA), DIAMOnD (DM), node2vec (ND), random walk (RW), and GenePANDA (GP) ranking were compared to suggest that WINNER identified expanded and top-ranked genes more significantly related to disease biology than those specified by other gene prioritizing software tools, including IPA and DM.

4.1.3 *Other node ranking methods*

Network-based methods. Multiple heuristic algorithms have been developed in the prioritization of nodes using **network-based methods** [108-111], including various models such as random walks (e.g., PageRank and Hyperlink-Induced Topic Search), conditional random field, and flow propagation.

Graph-theoretical node importance metrics. There are a number of graph-theoretical node importance metrics, such as node centrality. Betweenness centrality of a node is the number of shortest paths (between all possible pairs of nodes) that pass through the node. Closeness centrality (or closeness) of a node is a measure of centrality in a network, calculated as the reciprocal of the sum of the length of the shortest paths between the node and all other nodes. Similarly, the eccentricity centrality (or eccentricity) of a node i is calculated as the reciprocal of the longest shortest path between the i and all other nodes. Since the previous centralities are expensive to compute, semi-local centrality [112] is proposed as a more compute-efficient alternative that is more effective than degree centrality (or simply, node degree which is the number of links incident upon a node) for identifying the influencers. Let $N(w)$ be the number of vertices within two hops from node w, and define a node metric $Q(j) = \Sigma_{w \in \Gamma_j N(w)}$ where Γ_j means the set of node j's direct neighbors, then the semi-local centrality of node i is equal to $\Sigma_{w \in \Gamma_i N(w)} Q(j)$

Degree-based metric. Degree-based metric is the core number of a vertex. Specifically, the k-core of a network is a maximal connected subgraph in which all nodes have a degree of at least k. The k-core is unique and can be computed using a peeling algorithm that repeatedly removes a vertex with degree $< k$ until no more such vertex exists. The k-shell is the subgraph of nodes in the k-core but not in the $(k + 1)$-core, and in this case, every node in the k-shell has a core number equal to k.

Other bioentity ranking methods. According to the hypothesis that functionally related gene disorders result in phenotypically similar diseases, gene prioritization implements the similarity scores for the ranking between the candidate genes and the known disease genes.

Popular tools. One of the popular tools is ToppGene [113] which provides a grand similarity score that combines multiple similarity scores by comparing the test genes' annotations to the training genes' enriched annotations. The annotation can be either categorical terms such as Gene Ontology, pathways, drug targets, etc., or numeric annotations, i.e., the microarray expression values. A fuzzy-based similarity measure [114] is applied for the categorical terms, and Pearson correlation is measured for the numeric correlations. To provide a grand score that is able to aggregate all the similarity scores of different annotations, ToppGene applies a statistical meta-analysis. In this analysis, an empirical p-value will be assigned to each annotation based on the similarity score ranking in the score distribution generated from a random sampling of the whole genome. Assuming p-values come from independent tests, ToppGene adopts Fisher's inverse chi-square to combine the multiple annotations' p-values into a grand p-value. The grand similarity score is simply 1 – the grand p-value. Another tool called Endeavour [115] applies a similar principle but with more data sources integrated. To perform an integrated pathway analysis, upstream kinomics data were combined with lncRNA proximal genes of interest using MetaCore/GeneGo (Clarivate Analytics) in parallel with the aforementioned analyses.

4.2 *Ranking biomedical entity-to-entity associations*

4.2.1 *WIPER*

The edge metrics we have seen so far do not utilize edge weights which are common in biological networks. We show how to prioritize network edges using WIPER [4], which is able to effectively utilize weighted edges along with graph topology to rank the edges so as to solve biomedical problems such as finding a therapeutic strategy by "targeting the right interactions in the interactome" [116].

WIPER treats an input biological network as a probabilistic network where each edge (i, j) has a weight between 0 and 1 modeling the probability that node i successfully regulates node j, denoted by $Pr(i, j)$. A key innovation of WIPER is that it will transform this probabilistic network among bioentities into another weighted network where each node is an edge in the original graph, so that node ranking algorithms as introduced in Section 4.1.2 (e.g., BEERE) can be readily used to rank the edges. WIPER can also predict new edges in the network, and those newly discovered interactions may represent novel biological mechanisms to form new hypotheses.

Given an input biological network $G = (V, E)$ where V is the bioentity node set and E is the edge set where each edge (i, j) has weight $Pr(i, j)$, WIPER first resets the edge weight into $w(i, j)$ which is the score contributed by the optimal path from i to j. Let us denote the path by i -> k_1 -> k_2 -> … -> k_t -> j, then the probability that node i successfully regulates node j along this path is given by:

$$Pr\left(i,k_1\right)\cdot \Pi_{\tau=1}^{t-1} Pr\left(k_\tau,k_{\tau+1}\right)\cdot Pr\left(k_t,j\right),$$

which can be higher than the weight of the original edge $Pr(i, j)$, making this new weight more appropriate.

Therefore, the new edge weight $w(i, j)$ is decided by the optimal path from i to j that maximizes the probability of regulation:

$$w\left(i,j\right)=\max_{k_1,k_2,\ldots,k_t} Pr\left(i,k_1\right)\cdot \Pi_{\tau=1}^{t-1} Pr\left(k_\tau,k_{\tau+1}\right)\cdot Pr\left(k_t,j\right),$$

which can be calculated by the Floyd-Warshall algorithm that computes the lengths of shortest paths between all pairs of vertices, modified so that edge weights are aggregated by multiplication instead of summation, and instead of keeping the shortest path length, here, $w(i, j)$ keeps the highest probability product among potential paths. After this step, the biological network G now has edge weights $w(i, j)$ replacing $Pr(i, j)$ (unless the original edge (i, j) is already the optimal path connection between i and j).

Next, WIPER transforms this updated bioentity graph G into another graph $\mathfrak{G}$ where each node corresponds to an edge in G. Let us denote $v_{i,j}$ as a node of $\mathfrak{G}$ that corresponds to the edge (i, j) in G. Then, we can set the weight of an edge $(v_{i,j}, v_{k,\ell})$ as:

$$w\left(v_{i,j},v_{k,\ell}\right)=Pr\left(i,j\right)\cdot Pr\left(k,\ell\right)\cdot max\left\{Pr\left(i,k\right),Pr\left(i,\ell\right),Pr\left(j,k\right),Pr\left(j,\ell\right)\right\},$$

which is a product of the existence probabilities of (i, j) and (j, ℓ), and the probability of the most likely connecting path between two node sets $\{i, j\}$ and $\{j, \ell\}$, which is similar to the idea of single-link hierarchical clustering. Finally, the edge WIPER scores are calculated by performing the aforementioned node ranking algorithms on the graph, and p-value estimation. Please refer to Section 4.1.2 for the details.

Application: WIPER can be easily accessed using an API and is an effective method for evaluating network edge importance in any weighted network [117]. In the context of literature retrieval, WIPER uses the PubMed score as a measure of semantic validation [105]. The p-value evaluation in WIPER can be adapted to work with any ranking algorithm tools, such as the Polar Gini Curve [118].

4.2.2 *Other node ranking methods*

Ranking edges in a biological network allows quick focus on top-ranked edges for subsequent biological interpretations [119-121]. Critical edges can be divided into two categories based on their roles: some enhance the locality, like the ones inside a cluster, while others contribute to global connectivity, like the ones connecting two clusters.

There are a number of graph-theoretical metrics for finding critical edges. Among locality metrics, **degree product** (and its variants) [122] supposes that edges whose end-nodes have high degrees are critical.

Betweenness centrality. The betweenness centrality of an edge (i, j) is defined as

$$BC(i,j) = \sum\nolimits_{s \neq t \in V} \frac{\delta_{st}(i,j)}{\delta_{st}},$$

where δ_{st} denotes the number of all the shortest paths between node s and node t, $\delta_{st}\ (i, j)$ denotes the number of all the shortest paths between node s and node t that pass through the edge (i, j). The larger the betweenness centrality score is, the more shorter paths that pass through the edge, and hence the more important the edge is.

Jaccard coefficient. Among global connectivity metrics, the Jaccard coefficient of an edge (i, j) is defined as

$$J(i,j) = \frac{\left|\Gamma_i \cap \Gamma_j\right|}{\left|\Gamma_i \cup \Gamma_j\right|},$$

where i and j are the two end nodes of the edge (i, j) and Γ_i is the set of i's neighbors. Edge (i, j) is essential (regarding global connectivity) if its Jaccard coefficient is low, since node i and node j have less common neighbors, so information is more likely to have to pass through (i, j).

Bridgeness index. The bridgeness index (or bridgeness) [123] of an edge (i, j) is defined as

$$B(i,j) = \frac{\sqrt{S_i S_j}}{S_{i,j}},$$

where S_i, S_j and $S_{i,j}$ are the sizes of the maximum cliques containing nodes i, j and edge (i, j), respectively. Similar to Jaccard coefficient, edge (i, j) is important in terms of global connectivity if its bridgeness is low. Intuitively, edges in smaller cliques are more important since their removal tends to impact global connectivity more.

Reachability index. The reachability index [124] of an edge (i, j) is defined as

$$R(i,j) = \frac{1}{|V|} \sum\nolimits_{s \in V} \left|R(s; G-(i,j))\right|,$$

where $|V|$ is the number of nodes, $G - (i, j)$ is the subnetwork by removing edge (i, j), and $|R(s;G - (i, j))|$ is the number of nodes reachable from a node s in $G - (i, j)$. A lower reachability index means that after removing edge (i, j), the average number of reachable nodes is decreased more and thus edge (i, j) is important.

We remark that while locality metrics for edges are intuitive in ranking bioentity-wise associations, global connectivity metrics are also useful. For example, [125] explores the weak-ties effect in PPI networks to discover protein complexes, where a similarity function is defined based on bridgeness to quantify how close a node is to its core component.

4.3 *Literature-mined semantic networks*

Besides finding significant nodes and edges, it is also interesting to find active subnetworks (or, modules) which is useful in applications such as detecting/uncovering molecular complexes/structures (densely connected regions in large PPI networks) [126, 127], and identifying recurrent patterns across multiple networks to discover biological modules [128].

Many algorithms have been designed to identify active subnetworks or modules participating in condition-specific biological functions. In fact, in many such algorithms, the previously introduced node and edge importance metrics serve as their input.

The survey of [23] classifies subnetwork mining methods into six categories: (1) greedy algorithms, (2) evolutionary algorithms, (3) maximal clique identification, (4) random walk algorithms, (5) diffusion emulation models, and (6) clustering-based methods. Most of them are similar to what we have covered, such as greedy search (node expansion from seed nodes), random walks, and clustering (e.g., the gene modules in WGCNA found by hierarchical clustering). Thus, this section covers more unique methods, including dense subgraphs, frequent subgraph patterns, and biclustering.

We remark that some methods, such as evolutionary algorithms, directly maximize a subgraph scoring function, such as the aggregate score of nodes and edges, or the similarity between the connected modules provided for different diseases; while others are based on the structural and expression-order relationships. The statistical significance of the subnetworks is estimated by its score's empirical p-value in the sample and gene permutations or subnetwork randomization. We can also evaluate the cohesion of the subnetworks found using metrics such as degeneracy and h-index. The degeneracy of a graph is the largest value k such that it has a k-core, while the h-index of a graph is the maximum number h such that the graph contains h vertices of degree at least h.

Dense subgraphs. There are many definitions of dense subgraph structures that can be mined from underlying biological networks. Cliques are subnetworks in which every node is connected with all others, and maximal cliques can be used to find active protein modules though preprocessing is often needed to simplify the network to reduce computation cost [23]. Since requiring a subnetwork to be a complete graph is too strict, relaxations called quasi-cliques can be used to improve recall. A degree-based p-quasi-clique requires that every node in the subnetwork be connected to at least p fraction of the other vertices, which has been used to explore significant protein modules in a PPI network [129] and to classify whole-genome protein sequences in a linkage graph [130]. A similar definition to the degree-based p-quasi-clique is k-plex, where a node is allowed to be a member of the subnetwork when it has connections to all but at most k other members. A density-based p-quasi-clique requires the edge density of the subnetwork to be at least p, where the density is evaluated by comparing to the number of all possible edges among nodes of the subnetwork (i.e., when it is a clique), and it has been used for clustering PPI networks to detect functional groups [127]. Other definitions include k-core which was introduced before, and k-truss which is the maximal subnetwork where each edge is involved in at least k triangles (i.e., 3-cliques). Another way is to find hub-induced subgraphs from the neighborhoods of high-degree nodes, such as [131] which keeps removing edges with high betweenness centrality to obtain sub-groups such that the intra-group connections are dense and intergroup connections sparse, which serve as potential functional modules in a PPI network.

Frequent subgraph patterns. Frequent pattern mining (FPM) is a classical problem at the core of data mining research, which aims to find patterns that appear in at least a fraction α of all data objects (called transactions in FPM terminology) in a database. We consider the problem of frequent subgraph pattern mining where a pattern is a subgraph, and a data object is a network. There are two variants: one is to find patterns that occur in many networks, and the other is to find patterns that occur in different places in a big network. The patterns can be network motifs, i.e., statistically overrepresented sub-structures (sub-graphs) in a network, and the study of biological network motifs may reveal answers to many important biological questions.

The seminal work of [132] mines the transcriptional regulation network of *E. coli* and finds that much of it consists of repeated appearances of three highly significant motifs. For each network motif, there is a function in determining gene expression, such as generating temporal expression programs and governing the responses to fluctuating external signals. The motif structure also provides an easily interpretable view of the entire known transcriptional network of the organism. CODENSE [128] is an algorithm to mine frequent coherent dense subgraphs across large numbers of graphs, and has been used across multiple networks to identify recurrent patterns as biological modules; CODENSE was applied to 39 co-expression networks derived from microarray datasets, which discovered a large number of functionally homogeneous clusters and made functional predictions for 169 uncharacterized yeast genes. NemoProfile [133] identified five network motifs from *E. coli* and *Saccharomyces cerevisiae* PPI networks, which are used to construct protein attributes that are fed into a decision tree to predict essential proteins. Network motifs have also been mined in metabolic networks to study pathway evolution [134]. Network motifs are also defined in bipartite graphs that can model enzyme-reaction links in metabolic pathways and gene-disease associations, such as butterflies (or 2 × 2 bicliques) [135].

Biclustering. Biclustering can be applied directly on the microarray X, and finds the so-called biclusters (R, C) where R is a row set of biomolecules and C is a column set of test conditions, such that expression values $\{x_{ij}\}$ where i ϵR and j ϵC are highly correlated. An example of biclustering is the concept of order-preserving submatrices (OPSM)

[136]: a submatrix S of X is an OPSM if there is a permutation of the columns of S, under which the expression values of each row in S are strictly increasing.

In the gene expression data analysis using microarray experiments, genes (rows) with simultaneous mRNA expression changes across different time points (columns) may share the same cell-cycle related properties [137]; columns may also refer to different experimental conditions as in [138]. In this application, an OPSM represents co-expressed gene patterns among cohorts in a particular stage of a disease or under the same drug treatment, etc. [139]. This biclustering method overcomes the shortcoming of traditional gene clustering methods in the identification of patterns corresponding to only a part of the expression matrix X, since the computation of gene correlations in GCNs are based on all columns.

Biclustering also applies to a bipartite graph. Statistical-Algorithmic Method for Bicluster Analysis (SAMBA) [140] identifies active modules by building a weighted bipartite graph in which two types of nodes represent genes and proteins, respectively. A connection between these two types of nodes represents the probability of gene protein associations. Biclustering enables the identification of the locally optimal subgraphs. Overlap is minimized by limiting the shared properties between subgraphs. The SAMBA performance relies on the selections of the properties layer.

5. Conclusions

We have seen how existing works build and analyze biological networks. However, the combination of different methods is still largely ad hoc, without a clear consensus on what is right or wrong. The goal is to organize these methods in a way that best serves biological interpretability. For example, while the original design of WGCNA was for mRNA networks, it has been expanded to miRNA and lncRNA networks [141, 142]. Moreover, studies have shown that integrating PPI network complexes and WGCNA co-expression networks can enhance the identification of hub gene signatures associated with important phenotypes, such as cancer and polygenic disorders [143-148]. By combining various network mining techniques and integrating different types of biological networks, researchers can increase their ability to identify and interpret gene signatures associated with specific phenotypes. Ultimately, this can help us better understand the molecular basis of complex diseases and design more effective strategies for their prevention and treatment. Although many algorithms have been developed for biological network mining, there are still inspiring and promising approaches that have yet to be fully explored. For example, several functional dependencies (FD) mining algorithms for data profiling [149-152] could be applied to extract subnetworks from discrete expression data. Similarly, the backtracking algorithm for the transversal hypergraph problem [153] could be utilized to condense biological networks. In terms of data retrieval, Prefix Closed Pattern Tree based Frequent Closed Itemset mining algorithm (PCPT-FCI) [154] could be useful in mining frequently closed entities in biological subnetworks. Furthermore, specific domains such as inferring drug Mechanisms of Action (MOA) or off-target mechanisms from cell-line specific perturbational profiles [134] have the potential to yield valuable drug-gene networks [155].

Several directions are promising for future research, including (i) speeding up analysis by parallel computing, (ii) new approaches utilizing the temporal information of the networks, and (iii) deep learning models.

Parallel computing. Concurrent network construction is easy since the computing of correlation between different pairs of biomolecules (rows in a microarray) is embarrassingly parallel. Data-intensive graph-parallel systems are abundant such as the think-like-a-vertex programming paradigm pioneered by Google's Pregel [156] and followed by numerous other systems as reviewed by [157]. These systems are ideal for implementing concurrent network propagation and random walk algorithms for ranking nodes and edges; depending on the available hardware, one can choose from single-machine out-of-core systems such as GraphChi [158] and X-Stream [159], or distributed systems such as Pregel+ [160-162], Giraph [163], GraphLab [78], Blogel [164], Quegel [165, 166], and lightweight checkpointing (LWCP) [167], to list just a few. These systems are more than enough as they target the billion-node scale of online social networks, well surpassing the size of a typical biological network.

On the other hand, mining significant subgraphs are often more computationally expensive (often nondeterministic polynomial (NP)-hard) as the possible node set of a subgraph is the power set of the set of biomolecules in the biological network. Hence the search space grows exponentially with the network size. It is critical to share the computing workloads among all central processing unit (CPU) cores available, which is ill-suited for data-intensive systems mentioned before where execution is input output (IO)-bound. Fortunately, there are recent breakthroughs in systems research that allows massively parallel mining of dense subgraphs as supported by G-thinker [14, 16-19], and massively parallel mining of frequent subgraph patterns as supported by PrefixPFM [15, 20-22] and time-frequency signature matrix (T-FSM) [168]. Both systems allow the efficient finding of significant subgraphs and thus bring substantial potential to expedite biological findings using the approaches we introduced in Section 4.3.

An alternative to parallel computing is to use graph sketches which are succinct summaries of massive data (e.g., all shortest paths between nodes in a biological network) often estimated from a sample of the data and provide approximate answers with certain quality guarantees (e.g., with 99% probability that the error is within 1%). For example, the reachability index is usually estimated using bottom-*k* sketches [124] while node betweenness centrality can be estimated using hypergraph sketches [169].

Temporal network metrics. It is important to capture the dynamics of interactions among biomolecules that change over time, and we have seen that one solution is to use ODE to construct a GRN as by SCODE [93], but the constructed GRN for further analytics is still a static graph.

There are some recent breakthroughs in defining meaningful metrics useful in our network analysis as described in Section 4, which are worth exploring in future research. Some of these metrics consider the order that interactions can happen, such as temporal shortest path [170], temporal reachability [171] and node centralities [172]. Others consider the persistence of dense interactions through a consecutive period of time, such as temporal quasi-clique [173] and temporal k-core [174]. These new definitions have great potential to find more convincing network modules that respect temporal order and persist over some time.

Deep learning. Deep learning has revolutionized the domain of artificial intelligence (AI) due to its superior performance that dominates traditional machine learning models when given sufficient training data. Convolutional neural networks can extract features from matrix/tensor input and can be explored over the microarray data or the network adjacency matrix (e.g., for supervised learning tasks like functional classification), while recurrent neural networks can be utilized to capture the temporal dependencies. Both structures can be combined to capture the network dynamics using hybrid models such as convolutional neural network long short-term memory (CNN-LSTM) or convolutional long short-term memory (ConvLSTM) [175].

When a biological network is constructed, a more graph-native approach should be preferred which aligns well with the recent trend of GNNs [176] that is made popular by the seminal work on graph convolutional networks [177]. In this thread of research, the graph convolution layer operates on the neighborhood of each node and is the counterpart of the convolution layer in ConvNets, and graph pooling operates on the intermediate graph layer similar to pooling in ConvNets. Compared with other node embedding methods like DeepWalk and node2vec which only utilizes topology, graph convolution networks also allow each node to contain input features that are utilized in computing the extracted node/graph features for use by the ultimate node or whole-graph classification. Graph convolutional networks have recently been applied to predict the interfaces between proteins [178], to infer cell-cell interactions [179], to predict metastatic events in breast cancer [180], and to encode biological knowledge such as Gene Ontology for gene set enrichment analysis and for selecting Gene Ontology terms that are important to target clinical variables [181].

Ensemble multimodal. The Differential Evolution Artificial Neural Network (DE-ANN) algorithm [182] is a hybrid optimization technique that combines the strengths of differential evolution and artificial neural networks. It is a combination of the DE-ANN algorithm and Fuzzy c-Means (FCM) clustering for skin cancer detection. The ensemble multimodal deep learning model developed by Kumar et al. [183] utilizes Uniform-Net and convolutional neural network (CNN) to extract distinguishing features for coronavirus disease (COVID-19) patient diagnosis.

Acknowledgements

This work is partially supported by NSF OAC-1755464 (CRII).

Conflict of interest

Jake Chen has received research grants from NIH, Bristol-Myers Squibb Company, Swedish Health Services, V Foundation for Cancer Research, Seattle Children's Research Institute, Lupus Research Alliance, Oregon Health &

Science University, University of Maryland. Da Yan has received research grants from NSF and ADECA. The authors declare that they have no additional conflict of interest other than co-authorship.

References

[1] Feist AM, Herrgård MJ, Thiele I, Reed JL, Palsson BØ. Reconstruction of biochemical networks in microorganisms. *Nature Reviews Microbiology*. 2009; 7(2): 129-143. https://doi.org/10.1038/nrmicro1949

[2] Yue Z, Wan P, Huang H, Xie Z, Chen JY. SLDR: A computational technique to identify novel genetic regulatory relationships. *BMC Bioinformatics*. 2014; 15(Suppl 11): S1. https://doi.org/10.1186/1471-2105-15-S11-S1

[3] Yue Z, Willey CD, Hjelmeland AB, Chen JY. BEERE: A web server for biomedical entity expansion, ranking and explorations. *Nucleic Acids Research*. 2019; 47(W1): W578-W586. https://doi.org/10.1093/nar/gkz428

[4] Yue Z, Nguyen T, Zhang E, Zhang J, Chen JY. WIPER: Weighted in-path edge ranking for biomolecular association networks. *Quantitative Biology*. 2019; 7(4): 313-326. https://doi.org/10.1007/s40484-019-0180-y

[5] Nguyen T, Yue Z, Slominski R, Welner R, Zhang J, Chen JY. WINNER: A network biology tool for biomolecular characterization and prioritization. *Frontiers in Big Data*. 2022; 5: 1016606. https://doi.org/10.3389/fdata.2022.1016606

[6] Weng Z, Yue Z, Zhu Y, Chen JY. DEMA: A distance-bounded energy-field minimization algorithm to model and layout biomolecular networks with quantitative features. *Bioinformatics*. 2022; 38(Supplement_1): i359-i368. https://doi.org/10.1093/bioinformatics/btac261

[7] Kilicoglu H, Shin D, Fiszman M, Rosemblat G, Rindflesch TC. SemMedDB: A PubMed-scale repository of biomedical semantic predications. *Bioinformatics*. 2012; 28(23): 3158-3160. https://doi.org/10.1093/bioinformatics/bts591

[8] Chen JY, Shen C, Sivachenko AY. Mining Alzheimer disease relevant proteins from integrated protein interactome data. In: Altman RB, Murray T, Klein TE, Dunker AK, Hunter L. (eds.) *Biocomputing 2006.* Singapore: World Scientific; 2005. p.367-378. https://doi.org/10.1142/9789812701626_0034

[9] Zhang F, Chen JY. A neural network approach to multi-biomarker panel development based on LC/MS/MS proteomics profiles: A case study in breast cancer. In: *2009 22nd IEEE International Symposium on Computer-Based Medical Systems*. California, United States: IEEE Computer Society; 2009. p.1-6. https://doi.org/10.1109/CBMS.2009.5255456

[10] Yue Z, Arora I, Zhang EY, Laufer V, Bridges SL, Chen JY. Repositioning drugs by targeting network modules: a Parkinson's disease case study. *BMC Bioinformatics*. 2017; 18(Suppl 14): 532. https://doi.org/10.1186/s12859-017-1889-0

[11] Li J, Zhu X, Chen JY. Discovering breast cancer drug candidates from biomedical literature. *International Journal of Data Mining and Bioinformatics*. 2010; 4(3): 241-255. https://doi.org/10.1504/IJDMB.2010.033519

[12] Huang LC, Wu X, Chen JY. Predicting adverse side effects of drugs. *BMC Genomics*. 2011; 12(Suppl 5): S11. https://doi.org/10.1186/1471-2164-12-S5-S11

[13] Chen JY, Piquette-Miller M, Smith BP. Network medicine: finding the links to personalized therapy. *Clinical Pharmacology & Therapeutics*. 2013; 94(6): 613-616. https://doi.org/10.1038/clpt.2013.195

[14] Yan D, Guo G, Chowdhury MMR, Özsu MT, Ku WS, Lui JCS. G-thinker: A distributed framework for mining subgraphs in a big graph. In: *2020 IEEE 36th IEEE International Conference on Data Engineering (ICDE)*. California, United States: IEEE Computer Society; 2020. p.1369-1380. https://doi.org/10.1109/ICDE48307.2020.00122

[15] Yan D, Qu W, Guo G, Wang X. PrefixFPM: A parallel framework for general-purpose frequent pattern mining. In: *2020 IEEE 36th IEEE International Conference on Data Engineering (ICDE)*. California, United States: IEEE Computer Society; 2020. p.1938-1941. https://doi.org/10.1109/ICDE48307.2020.00208

[16] Yan D, Guo G, Khalil J, Özsu MT, Ku WS, Lui JCS. G-thinker: A general distributed framework for finding qualified subgraphs in a big graph with load balancing. *The VLDB Journal*. 2022; 31: 287-320. https://doi.org/10.1007/s00778-021-00688-z

[17] Guo G, Yan D, Özsu MT, Jiang Z, Khalil J. Scalable mining of maximal quasi-cliques: An algorithm-system codesign approach. *Proceedings of the VLDB Endowment*. 2020; 14(4): 573-585. https://doi.org/10.14778/3436905.3436916

[18] Khalil J, Yan D, Guo G, Yuan L. Parallel mining of large maximal quasi-cliques. *The VLDB Journal*. 2022; 31(4): 649-674. https://doi.org/10.1007/s00778-021-00712-2

[19] Guo G, Yan D, Yuan L, Khalil J, Long C, Jiang Z, et al. Maximal directed quasi -clique mining. In: *2022 IEEE 38th International Conference on Data Engineering (ICDE)*. California, United States: IEEE Computer Society; 2022. p.1900-1913. https://doi.org/10.1109/ICDE53745.2022.00188

[20] Yan D, Qu W, Guo G, Wang X, Zhou Y. PrefixFPM: A parallel framework for general-purpose mining of frequent and closed patterns. *The VLDB Journal*. 2022; 31: 253-286. https://doi.org/10.1007/s00778-021-00687-0

[21] Qu W, Yan Da, Guo G, Wang X, Zou L, Zhou Y. Parallel mining of frequent subtree patterns. In: Qin L, Zhang W, Zhang Y, Peng Y, Kato H, Wang W, et al. (eds.) *Software foundations for data interoperability and large scale graph data analytics*. Cham, Switzerland: Springer; 2020. p.18-32. https://doi.org/10.1007/978-3-030-61133-0_2

[22] Cheng J, Yan D, Qu W, Hao X, Long C, Ng W, et al. Mining order-preserving submatrices under data uncertainty: A possible-world approach and efficient approximation methods. *ACM Transactions on Database Systems*. 2022; 47(2): 7. https://doi.org/10.1145/3524915

[23] Nguyen H, Shrestha S, Tran D, Shafi A, Draghici S, Nguyen T. A comprehensive survey of tools and software for active subnetwork identification. *Frontiers in Genetics*. 2019; 10: 155. https://doi.org/10.3389/fgene.2019.00155

[24] Chen S, Mar JC. Evaluating methods of inferring gene regulatory networks highlights their lack of performance for single cell gene expression data. *BMC Bioinformatics*. 2018; 19: 232. https://doi.org/10.1186/s12859-018-2217-z

[25] Pavlopoulos GA, Secrier M, Moschopoulos CN, Soldatos TG, Kossida S, Aerts J, et al. Using graph theory to analyze biological networks. *BioData Mining*. 2011; 4: 10. https://doi.org/10.1186/1756-0381-4-10

[26] Pellegrini M, Haynor D, Johnson JM. Protein interaction networks. *Expert Review of Proteomics*. 2014; 1(2): 239-249. https://doi.org/10.1586/14789450.1.2.239

[27] Gavin AC, Bösche M, Krause R, Grandi P, Marzioch M, Bauer A, et al. Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature*. 2002; 415: 141-147. https://doi.org/10.1038/415141a

[28] Puig O, Caspary F, Rigaut G, Rutz B, Bouveret E, Bragado-Nilsson E, et al. The tandem affinity purification (TAP) method: A general procedure of protein complex purification. *Methods*. 2001; 24(3): 218-229. https://doi.org/10.1006/meth.2001.1183

[29] Vikis HG, Guan KL. Glutathione-S-transferase-fusion based assays for studying protein-protein interactions. In: Fu H. (ed.) *Protein-protein interactions methods and application: Methods in molecular biology, volume 261*. New Jersey, United States: Humana Press; 2004. p.175-186. https://doi.org/10.1385/1-59259-762-9:175

[30] Ito T, Chiba T, Ozawa R, Yoshida M, Hattori M, Sakaki Y. A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Biological Sciences*. 2001; 98(8): 4569-4574. https://doi.org/10.1073/pnas.061034498

[31] Stoll D, Templin MF, Bachmann J, Joos TO. Protein microarrays: Applications and future challenges. *Current Opinion in Drug Discovery & Development*. 2005; 8(2): 239-252. https://pubmed.ncbi.nlm.nih.gov/15782547/

[32] Willats WGT. Phage display: Practicalities and prospects. *Plant Molecular Biology*. 2002; 50: 837-854. https://doi.org/10.1023/A:1021215516430

[33] Tong AHY, Lesage G, Bader GD, Ding H, Xu H, Xin X, et al. Global mapping of the yeast genetic interaction network. *Science*. 2004; 303(5659): 808-813. https://doi.org/10.1126/science.1091317

[34] Krogan N, Cagney G, Yu H, Zhong G, Guo X, Ignatchenko A, et al. Global landscape of protein complexes in the yeast *Saccharomyces cerevisiae*. *Nature*. 2006; 440: 637-643. https://doi.org/10.1038/nature04670

[35] Mewes HW, Frishman D, Mayer KFX, Münsterkötter M, Noubibou O, Pagel P, et al. MIPS: Analysis and annotation of proteins from whole genomes in 2005. *Nucleic Acids Research.* 2006; 34(Suppl_1): D169-D172. https://doi.org/10.1093/nar/gkj148

[36] Xenarios I, Salwínski L, Duan XJ, Higney P, Kim S-M, Eisenberg D. DIP, the database of interacting proteins: A research tool for studying cellular networks of protein interactions. *Nucleic Acids Research*. 2002; 30(1): 303-305. https://doi.org/10.1093/nar/30.1.303

[37] Gavin A-C, Aloy P, Grandi P, Krause R, Boesche M, Marzioch M, et al. Proteome survey reveals modularity of the yeast cell machinery. *Natur*e. 2006; 440: 631-636. https://doi.org/10.1038/nature04532

[38] Costanzo MC, Hogan JD, Cusinck ME, Davis BP, Fancher AM, Hodges PE, et al. The yeast proteome database (YPD) and *Caenorhabditis elegans* Proteome database (WormPD): Comprehensive resources for the organization and comparison of model organism protein information. *Nucleic Acids Research*. 2000; 28(1): 73-76. https://doi.org/10.1093/nar/28.1.73

[39] Licata L, Briganti L, Peluso D, Perfetto L, Iannuccelli M, Galeota E, et al. MINT, the molecular interaction database: 2012 update. *Nucleic Acids Research*. 2012; 40(D1): D857-D861. https://doi.org/10.1093/nar/gkr930

[40] Kerrien S, Alam-Faruque Y, Aranda B, Bancarz I, Bridge A, Derow C, et al. IntAct-Open source resource for molecular interaction data. *Nucleic Acids Research*. 2007; 35(Suppl_1): D561-D565. https://doi.org/10.1093/nar/

gkl958

[41] Bader GD, Betel D, Hogue CWV. BIND: The biomolecular interaction network database. *Nucleic Acids Research*. 2003; 31(1): 248-250. https://doi.org/10.1093/nar/gkg056

[42] Oughtred R, Chatr-aryamontri A, Breitkreutzet B-J, Chang CS, Rust JM, Theesfeld CL, et al. BioGRID: A resource for studying biological interactions in yeast. *Cold Spring Harbor Protocols*. 2016; 2016(1): 29-33. https://doi.org/10.1101/pdb.top080754

[43] Goel R, Harsha HC, Pandey A, Prasad TSK. Human protein reference database and human proteinpedia as resources for phosphoproteome analysis. *Molecular BioSystems*. 2012; 8(2): 453-463. https://doi.org/10.1039/C1MB05340J

[44] Han K, Park B, Kim H, Hong J, Park J. HPID: The human protein interaction database. *Bioinformatics*. 2004; 20(15): 2466-2470. https://doi.org/10.1093/bioinformatics/bth253

[45] Murali T, Pacifico S, Yu J, Guest S, Roberts GG, Finley RL. DroID 2011: A comprehensive, integrated resource for protein, transcription factor, RNA and gene interactions for *Drosophila. Nucleic Acids Research*. 2011; 39(Suppl_1): D736-D743. https://doi.org/10.1093/nar/gkq1092

[46] Chen JY, Pandey P, Nguyen TM. HAPPI-2: A comprehensive and high-quality map of human annotated and predicted protein interactions. *BMC Genomics*. 2017; 18: 182. https://doi.org/10.1186/s12864-017-3512-1

[47] Kuhn M, Szklarczyk D, Franceschini A, Campillos M, von Mering C, Jensen LJ, et al. STITCH 2: An interaction network database for small molecules and proteins. *Nucleic Acids Research*. 2010; 38(Suppl_1): D552-D556. https://doi.org/10.1093/nar/gkp937

[48] Szklarczyk D, Morris JH, Cook H, Kuhn M, Wyder S, Simonovic M, et al. The STRING database in 2017: Quality-controlled protein-protein association networks, made broadly accessible. *Nucleic Acids Research*. 2017; 45(D1): D362-D368. https://doi.org/10.1093/nar/gkw937

[49] Zhu F, Farnung L, Kaasinen E, Sahu B, Yin Y, Wei B, et al. The interaction landscape between transcription factors and the nucleosome. *Nature*. 2018; 562: 76-81. https://doi.org/10.1038/s41586-018-0549-5

[50] Filtz TM, Vogel WK, Leid M. Regulation of transcription factor activity by interconnected post-translational modifications. *Trends in Pharmacological Sciences*. 2014; 35(2): 76-85. https://doi.org/10.1016/j.tips.2013.11.005

[51] Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, Gerber GK, et al. Transcriptional regulatory networks in *Saccharomyces cerevisiae. Science*. 2002; 298(5594): 799-804. https://doi.org/10.1126/science.1075090

[52] Sandelin A, Alkema W, Engström P, Wasserman WW, Lenhard B. JASPAR: An open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*. 2004; 32(Suppl_1): D91-D94. https://doi.org/10.1093/nar/gkh012

[53] Matys V, Kel-Margoulis OV, Fricke E, Liebich I, Land S, Barre-Dirrie A, et al. TRANSFAC and its module TRANSCompel: Transcriptional gene regulation in eukaryotes. *Nucleic Acids Research*. 2006; 34(Suppl_1): D108-D110. https://doi.org/10.1093/nar/gkj143

[54] Dinkel H, Chica C, Via A, Gould CM, Jensen LJ, Gibson TJ, et al. Phospho.ELM: A database of phosphorylation sites-update 2011. *Nucleic Acids Research*. 2011; 39(Suppl_1): D261-D267. https://doi.org/10.1093/nar/gkq1104

[55] Miller ML, Jensen LJ, Diella F, Jørgensen C, Tinti M, Li L, et al. Linear motif atlas for phosphorylation-dependent signaling. *Science Signaling*. 2008; 1(35): ra2. https://doi.org/10.1126/scisignal.1159433

[56] Gnad F, Ren S, Cox J, Olsen JV, Macek B, Oroshi M, et al. PHOSIDA (phosphorylation site database): Management, structural and evolutionary investigation, and prediction of phosphosites. *Genome Biology*. 2007; 8(11): R250. https://doi.org/10.1186/gb-2007-8-11-r250

[57] Edgar R, Domrachev M, Lash AE. Gene expression omnibus: NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*. 2002; 30(1): 207-210. https://doi.org/10.1093/nar/30.1.207

[58] Obayashi T, Hayashi S, Shibaoka M, Saeki M, Ohta H, Kinoshita K, et al. COXPRESdb: A database of coexpressed gene networks in mammals. *Nucleic Acids Research*. 2007; 36(Suppl_1): D77-D82. https://doi.org/10.1093/nar/gkm840

[59] Kholodenko BN, Hancock JF, Kolch W. Signalling ballet in space and time. *Nature Reviews Molecular Cell Biology*. 2010; 11: 414-426. https://doi.org/10.1038/nrm2901

[60] Gumerov VM, Ortega DR, Adebali O, Ulrich LE, Zhulin IB. MiST 3.0: An updated microbial signal transduction database with an emphasis on chemosensory systems. *Nucleic Acids Research*. 2020; 48(D1): D459-D464. https://doi.org/10.1093/nar/gkz988

[61] Krull M, Voss N, Choi C, Pistor S, Potapov A, Wingender E. TRANSPATH: An integrated database on signal transduction and a tool for array analysis. *Nucleic Acids Research*. 2003; 31(1): 97-100. https://doi.org/10.1093/nar/gkg089

[62] Jeong H, Tombor B, Albert R, Oltvai ZN, Barabási A-L. The large-scale organization of metabolic networks.

Nature. 2000; 407: 651-654. https://doi.org/10.1038/35036627

[63] Ma H, Sorokin A, Mazein A, Selkov A, Selkov E, Demin O, et al. The Edinburgh human metabolic network reconstruction and its functional analysis. *Molecular Systems Biology.* 2007; 3(1): 135. https://doi.org/10.1038/msb4100177

[64] Kanehisa M, Goto S, Furumichi M, Tanabe M, Hirakawa M. KEGG for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Research*. 2010; 38(Suppl_1): D355-D360. https://doi.org/10.1093/nar/gkp896

[65] Keseler IM, Bonavides-Martínez C, Collado-Vides J, Gama-Castro S, Gunsalus RP, Johnson DA, et al. EcoCyc: A comprehensive view of *Escherichia coli* biology. *Nucleic Acids Research*. 2009; 37(Suppl_1): D464-D470. https://doi.org/10.1093/nar/gkn751

[66] Karp PD, Ouzounis CA, Moore-Kochlacs C, Goldovsky L, Kaipa P, Ahrén D, et al. Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. *Nucleic Acids Research*. 2005; 33(19): 6083-6089. https://doi.org/10.1093/nar/gki892

[67] Whitaker JW, Letunic I, McConkey GA, Westhead DR. MetaTIGER: A metabolic evolution resource. *Nucleic Acids Research*. 2009; 37(Suppl_1): D531-D538. https://doi.org/10.1093/nar/gkn826

[68] Schilling CH, Letscher D, Palsson BØ. Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. *Journal of Theoretical Biology*. 2000; 203(3): 229-248. https://doi.org/10.1006/jtbi.2000.1073

[69] Schilling CH, Palsson BØ. Assessment of the metabolic capabilities of *Haemophilus influenzae* Rd through a genome-scale pathway analysis. *Journal of Theoretical Biology*. 2000; 203(3): 249-283. https://doi.org/10.1006/jtbi.2000.1088

[70] Schilling CH, Schuster S, Palsson BØ, Heinrich R. Metabolic pathway analysis: Basic concepts and scientific applications in the post-genomic era. *Biotechnology Progress*. 2008; 15(3): 296-303. https://doi.org/10.1021/bp990048k

[71] Schuster S, Fell DA, Dandekar T. A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotechnology*. 2000; 18: 326-332. https://doi.org/10.1038/73786

[72] Schuster S, Dandekar T, Fell DA. Detection of elementary flux modes in biochemical networks: A promising tool for pathway analysis and metabolic engineering. *Trends in Biotechnology*. 1999; 17(2): 53-60. https://doi.org/10.1016/S0167-7799(98)01290-6

[73] Yugi K, Kubota H, Hatano A, Kuroda S. Trans-omics: How to reconstruct biochemical networks across multiple 'omic' layers. *Trends in Biotechnology.* 2016; 34(4): 276-290. https://doi.org/10.1016/j.tibtech.2015.12.013

[74] Ishii N, Nakahigashi K, Baba T, Robert M, Soga T, Kanai A, et al. Multiple high-throughput analyses monitor the response of *E. coli* to perturbations. *Science*. 2007; 316(5824): 593-597. https://doi.org/10.1126/science.1132067

[75] Buescher JM, Liebermeister W, Jules M, Uhr M, Muntel J, Botella E, et al. Global network reorganization during dynamic adaptations of *Bacillus subtilis* metabolism. *Science*. 2012; 335(6072): 1099-1103. https://doi.org/10.1126/science.1206871

[76] Yugi K, Kubota H, Toyoshima Y, Noguchi R, Kawata K, Komori Y, et al. Reconstruction of insulin signal flow from phosphoproteome and metabolome data. *Cell Reports*. 2014; 8(4): 1171-1183. https://doi.org/10.1016/j.celrep.2014.07.021

[77] Langfelder P, Horvath S. WGCNA: An R package for weighted correlation network analysis. *BMC Bioinformatics*. 2008; 9: 559. https://doi.org/10.1186/1471-2105-9-559

[78] Gonzalez JE, Low Y, Gu H, Bickson D, Carlos Guestrin. PowerGraph: Distributed graph-parallel computation on natural graphs. In: *10th USENIX Symposium on Operating Systems Design and Implementation (OSDI '12), Hollywood, CA, USA, October 8-10, 2012*. California, United States: USENIX Association; 2012. p.17-30. https://www.usenix.org/system/files/conference/osdi12/osdi12-final-167.pdf

[79] Wilcox RR. Correlation and tests of independence. In: Wilcox RR. (ed.) *Introduction to robust estimation and hypothesis testing*. 3rd ed. Waltham, United States: Academic Press; 2012. p.441-469. https://doi.org/10.1016/B978-0-12-386983-8.00009-3

[80] Zhang B, Horvath S. A general framework for weighted gene co-expression network analysis. *Statistical Applications in Genetics and Molecular Biology*. 2005; 4(1): 17. https://doi.org/10.2202/1544-6115.1128

[81] Langfelder P, Zhang B, Horvath S. Defining clusters from a hierarchical cluster tree: The dynamic tree cut package for R. *Bioinformatics*. 2008; 24(5): 719-720. https://doi.org/10.1093/bioinformatics/btm563

[82] Di Z, Zhou S, Xu G, Ren L, Li C, Ding Z, et al. Single-cell and WGCNA uncover a prognostic model and potential oncogenes in colorectal cancer. *Biological Procedures Online*. 2022; 24: 13. https://doi.org/10.1186/

s12575-022-00175-x

[83] Chen Y, Liao R, Yao Y, Wang Q, Fu L. Machine learning to identify immune-related biomarkers of rheumatoid arthritis based on WGCNA network. *Clinical Rheumatology.* 2022; 41: 1057-1068. https://doi.org/10.1007/s10067-021-05960-9

[84] Zhang R, Chen Y, He J, Gou HY, Zhu YL, Zhu YM. WGCNA combined with GSVA to explore biomarkers of refractory neocortical epilepsy. *IBRO Neuroscience Reports*. 2022; 13: 314-321. https://doi.org/10.1016/j.ibneur.2022.09.010

[85] Margolin AA, Nemenman I, Basso K, Wiggins C, Stolovitzky G, Favera RD, et al. ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*. 2006; 7: S7. https://doi.org/10.1186/1471-2105-7-S1-S7

[86] Greenfield A, Madar A, Ostrer H, Bonneau R. DREAM4: Combining genetic and dynamic information to identify biological networks and dynamical models. *PLoS One*. 2010; 5(10): e13397. https://doi.org/10.1371/journal.pone.0013397

[87] Faith JJ, Hayete B, Thaden JT, Mogno I, Wierzbowski J, Cottarel G, et al. Large-scale mapping and validation of *Escherichia coli* transcriptional regulation from a compendium of expression profiles. *PLoS Biology*. 2007; 5(1): e8. https://doi.org/10.1371/journal.pbio.0050008

[88] Williams PL, Beer RD. Nonnegative decomposition of multivariate information. *arXiv* [Preprint] 2010. Version 1. https://doi.org/10.48550/arXiv.1004.2515v1

[89] Chan TE, Stumpf MPH, Babtie AC. Gene regulatory network inference from single-cell data using multivariate information measures. *Cell Systems*. 2017; 5(3): 251-267. https://doi.org/10.1016/j.cels.2017.08.014

[90] Song L, Langfelder P, Horvath S. Comparison of co-expression measures: Mutual information, correlation, and model based indices. *BMC Bioinformatics*. 2012; 13: 328. https://doi.org/10.1186/1471-2105-13-328

[91] McLeay RC, Bailey TL. Motif enrichment analysis: A unified framework and an evaluation on ChIP data. *BMC Bioinformatics*. 2010; 11: 165. https://doi.org/10.1186/1471-2105-11-165

[92] Aibar S, González-Blas CB, Moerman T, Huynh-Thu VA, Imrichova H, Hulselmans G, et al. SCENIC: Single-cell regulatory network inference and clustering. *Nature Methods*. 2017; 14: 1083-1086. https://doi.org/10.1038/nmeth.4463

[93] Matsumoto H, Kiryu H, Furusawa C, Ko MSH, Ko SBH, Gouda N, et al. SCODE: An efficient regulatory network inference algorithm from single-cell RNA-Seq during differentiation. *Bioinformatics*. 2017; 33(15): 2314-2321. https://doi.org/10.1093/bioinformatics/btx194

[94] Trapnell C, Cacchiarelli D, Grimsby J, Pokharel P, Li S, Morse M, et al. The dynamics and regulators of cell fate decisions are revealed by pseudotemporal ordering of single cells. *Nature Biotechnology*. 2014; 32: 381-386. https://doi.org/10.1038/nbt.2859

[95] Cairelli MJ, Miller CM, Fiszman M, Workman TE, Rindflesch TC. Semantic MEDLINE for discovery browsing: using semantic predications and the literature-based discovery paradigm to elucidate a mechanism for the obesity paradox. *AMIA Annual Symposium Proceedings Archive*. 2013; 2013: 164-173. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3900170/

[96] Rindflesch TC, Fiszman M. The interaction of domain knowledge and linguistic structure in natural language processing: Interpreting hypernymic propositions in biomedical text. *Journal of Biomedical Informatics.* 2003; 36(6): 462-477. https://doi.org/10.1016/j.jbi.2003.11.003

[97] Bodenreider O. The unified medical language system (UMLS): Integrating biomedical terminology. *Nucleic Acids Research*. 2004; 32(Suppl_1): D267-D270. https://doi.org/10.1093/nar/gkh061

[98] Wei CH, Kao HY, Lu Z. PubTator: A web-based text mining tool for assisting biocuration. *Nucleic Acids Research*. 2013; 41(W1): W518-W522. https://doi.org/10.1093/nar/gkt441

[99] Page L, Brin S, Motwani R, Winograd T. The PageRank citation ranking: Bringing order to the web. *Stanford InfoLab*. 1999.

[100] Cowen L, Ideker T, Raphael BJ, Sharan R. Network propagation: A universal amplifier of genetic associations. *Nature Reviews Genetics*. 2017; 18: 551-562. https://doi.org/10.1038/nrg.2017.38

[101] Senanayake U, Piraveenan M, Zomaya A. The pagerank-index: Going beyond citation counts in quantifying scientific impact of researchers. *PLoS One*. 2015; 10(8): e0134794. https://doi.org/10.1371/journal.pone.0134794

[102] Wan X, Wang W, Liu J, Tong T. Estimating the sample mean and standard deviation from the sample size, median, range and/or interquartile range. *BMC Medical Research Methodology*. 2014; 14: 135. https://doi.org/10.1186/1471-2288-14-135

[103] Yue Z, Zheng Q, Neylon MT, Yoo M, Shin J, Zhao Z, et al. PAGER 2.0: An update to the pathway, annotated-list and gene-signature electronic repository for Human Network Biology. *Nucleic Acids Research*. 2018; 46(D1):

D668-D676. https://doi.org/10.1093/nar/gkx1040

[104] Stackhouse CT, Anderson JC, Yue Z, Nguyen T, Eustace NJ, Langford CP, et al. An in vivo model of glioblastoma radiation resistance identifies long noncoding RNAs and targetable kinases. *JCI Insight*. 2022; 7(16): e148717. https://doi.org/10.1172/jci.insight.148717

[105] Yue Z, Slominski R, Bharti S, Chen JY, et al. PAGER Web APP: An interactive, online gene set and network interpretation tool for functional genomics. *Frontiers in Genetics*. 2022; 13: 820361. https://doi.org/10.3389/fgene.2022.820361

[106] Gong E, Chen JY. Prioritizing complex disease genes from heterogeneous public databases. *bioRxiv* [Preprint] 2023. https://doi.org/10.1101/2023.02.09.527562 [Accessed 21 March 2023].

[107] Zhang F, Chen JY. Discovery of pathway biomarkers from coupled proteomics and systems biology methods. *BMC Genomics*. 2010; 11(Suppl 2): S12. https://doi.org/10.1186/1471-2164-11-S2-S12

[108] Nitsch D, Tranchevent L-C, Gonçalves JP, Vogt JK, Madeira SC, Moreau Y. PINTA: A web server for network-based gene prioritization from expression data. *Nucleic Acids Research*. 2011; 39(Suppl_2): W334-W338. https://doi.org/10.1093/nar/gkr289

[109] Xie B, Agam G, Balasubramanian S, Xu J, Gilliam TC, Maltsev N, et al. Disease gene prioritization using network and feature. *Journal of Computational Biology*. 2015; 22(4): 313-323. https://doi.org/10.1089/cmb.2015.0001

[110] Navlakha S, Kingsford C. The power of protein interaction networks for associating genes with diseases. *Bioinformatics*. 2010; 26(8): 1057-1063. https://doi.org/10.1093/bioinformatics/btq076

[111] Chen J, Aronow BJ, Jegga AG. Disease candidate gene identification and prioritization using protein interaction networks. *BMC Bioinformatics*. 2009; 10: 73. https://doi.org/10.1186/1471-2105-10-73

[112] Chen D, Lü L, Shang MS, Zhang YC, Zhou T. Identifying influential nodes in complex networks. *Physica A: Statistical Mechanics and its Applications*. 2012; 391(4): 1777-1787. https://doi.org/10.1016/j.physa.2011.09.017

[113] Chen J, Bardes EE, Aronow BJ, Jegga AG. ToppGene suite for gene list enrichment analysis and candidate gene prioritization. *Nucleic Acids Research*. 2009; 37(Suppl_2): W305-W311. https://doi.org/10.1093/nar/gkp427

[114] Popescu M, Keller JM, Mitchell JA. Fuzzy measures on the gene ontology for gene product similarity. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. 2006; 3(3): 263-274. https://doi.org/10.1109/TCBB.2006.37

[115] Tranchevent L-C, Ardeshirdavani A, ElShal S, Alcaide D, Aerts J, Auboeuf D, et al. Candidate gene prioritization with Endeavour. *Nucleic Acids Research*. 2016; 44(W1): W117-W1121. https://doi.org/10.1093/nar/gkw365

[116] Ivanov AA, Khuri FR, Fu H. Targeting protein-protein interactions as an anticancer strategy. *Trends in Pharmacological Sciences*. 2013; 34(7): 393-400. https://doi.org/10.1016/j.tips.2013.04.007

[117] Yue Z, Yan D, Guo G, Chen JY. Biological network mining. In: Mukhtar S. (ed.) *Modeling transcriptional regulation: Methods and protocols*. New York, United States: Humana Press; 2021. p.139-151. https://doi.org/10.1007/978-1-0716-1534-8_8

[118] Nguyen TM, Jeevan JJ, Xu N, Chen JY. Polar Gini Curve: A technique to discover gene expression spatial patterns from single-cell RNA-seq data. *Genomics, Proteomics & Bioinformatics*. 2021; 19(3): 493-503. https://doi.org/10.1016/j.gpb.2020.09.006

[119] Theodosiou T, Efstathiou G, Papanikolaou N, Kyrpides NC, Bagos PG, Iliopoulos I, et al. NAP: The network analysis profiler, a web tool for easier topological analysis and comparison of medium-scale biological networks. *BMC Research Notes*. 2017; 10: 278. https://doi.org/10.1186/s13104-017-2607-8

[120] Wang Z, Dueñas-Osorio L, Padgett JE. A new mutually reinforcing network node and link ranking algorithm. *Scientific Reports*. 2015; 5: 15141. https://doi.org/10.1038/srep15141

[121] Wang J, Li M, Wang H, Pan Y. Identification of essential proteins based on edge clustering coefficient. *IEEE/ACM Transactions Computational Biology and Bioinformatics*. 2011; 9(4): 1070-1080. https://doi.org/10.1109/TCBB.2011.147

[122] Giuraniuc CV, Hatchett JPL, Indekeu JO, Leone M, Castillo IP, Schaeybroeck BV, et al. Trading interactions for topology in scale-free networks. *Physical Review Letters*. 2005; 95(9): 098701. https://doi.org/10.1103/PhysRevLett.95.098701

[123] Cheng XQ, Ren FX, Shen HW, Zhang ZK, Zhou T. Bridgeness: A local index on edge significance in maintaining global connectivity. *Journal of Statistical Mechanics: Theory and Experiment*. 2010; 2010: P10011. https://doi.org/10.1088/1742-5468/2010/10/P10011

[124] Saito K, Kimura M, Ohara K, Motoda H. Detecting critical links in complex network to maintain information flow/reachability. In: Booth R, Zhang ML. (eds.) *PRICAI 2016: Trends in Artificial Intelligence. 14th Pacific Rim International Conference on Artificial Intelligence, Phuket, Thailand, August 22-26*. Cham, Switzerland: Springer;

2016. p.419-432. https://doi.org/10.1007/978-3-319-42911-3_35

[125] Ma X, Gao L. Discovering protein complexes in protein interaction networks via exploring the weak ties effect. *BMC Systems Biology*. 2012; 6: S6. https://doi.org/10.1186/1752-0509-6-S1-S6

[126] Bader GD, Hogue CWV. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*. 2003; 4: 2. https://doi.org/10.1186/1471-2105-4-2

[127] Bu D, Zhao Y, Cai L, Xue H, Zhu X, Lu H, et al. Topological structure analysis of the protein-protein interaction network in budding yeast. *Nucleic Acids Research*. 2003; 31(9): 2443-2450. https://doi.org/10.1093/nar/gkg340

[128] Hu H, Yan X, Huang Y, Han J, Zhou XJ. Mining coherent dense subgraphs across massive biological networks for functional discovery. *Bioinformatics*. 2005; 21(Suppl_1): i213-i221. https://doi.org/10.1093/bioinformatics/bti1049

[129] Bhattacharyya M, Bandyopadhyay S. Mining the largest quasi-clique in human protein interactome. In: Bouchachia A, Nedjah N, Mourelle L, Pedrycz W. (eds.) *Proceedings of the 2009 International Conference on Adaptive and Intelligent Systems, Klagenfurt, Austria, 24-26 September 2009*. California, United States: IEEE Computer Society; 2009. p.194-199. https://doi.org/10.1109/ICAIS.2009.39

[130] Matsuda H, Ishihara T, Hashimoto A. Classifying molecular sequences using a linkage graph with their pairwise similarities. *Theoretical Computer Science*. 1999; 210(2): 305-325. https://doi.org/10.1016/S0304-3975(98)00091-7

[131] Ucar D, Asur S, Catalyurek U, Parthasarathy S. Improving functional modularity in protein-protein interactions graphs using hub-induced subgraphs. In: Fürnkranz J, Scheffer T, Spiliopoulou M. (eds.) *Knowledge Discovery in Databases: PKDD 2006. 10th European Conference on Principles and Practice of Knowledge Discovery in Databases, Berlin, Germany, September 18-22, 2006*. Berlin, Germany: Springer; 2006. p.371-382. https://doi.org/10.1007/11871637_36

[132] Shen-Orr SS, Milo R, Mangan S, Alon U. Network motifs in the transcriptional regulation network of *Escherichia coli. Nature Genetics*. 2002; 31: 64-68. https://doi.org/10.1038/ng881

[133] Kim W, Haukap L. NemoProfile as an efficient approach to network motif analysis with instance collection. *BMC Bioinformatics*. 2017; 18(Suppl 12): 423. https://doi.org/10.1186/s12859-017-1822-6

[134] Lacroix V, Fernandes CG, Sagot M-F. Motif search in graphs: Application to metabolic networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*. 2006; 3(4): 360-368. https://doi.org/10.1109/TCBB.2006.55

[135] Sanei-Mehri S-V, Zhang Y, Sariyüce AE, Tirthapura S. FLEET: Butterfly estimation from a bipartite graph stream. In: *CIKM '19: The 28th ACM International Conference on Information and Knowledge Management*. New York, United States: Association for Computing Machinery; 2019. p.1201-1210. https://doi.org/10.1145/3357384.3357983

[136] Cheng J, Yan D, Qu W, Hao X, Long C, Ng W, et al. Mining order-preserving submatrices under data uncertainty: A possible-world approach. In: *2019 IEEE 35th International Conference on Data Engineering (ICDE 2019)*. California, United States: IEEE Computer Society; 2019. p.1154-1165. https://doi.org/10.1109/ICDE.2019.00106

[137] Yip KY, Kao B, Zhu X, Chui CK, Lee SD, Cheung DW. Mining order-preserving submatrices from data with repeated measurements. *IEEE Transactions on Knowledge and Data Engineering*. 2013; 25(7): 1587-1600. https://doi.org/10.1109/TKDE.2011.167

[138] Fang Q, Ng W, Feng J, Li Y. Mining order-preserving submatrices from probabilistic matrices. *ACM Transactions on Database Systems*. 2014; 39(1): 6. https://doi.org/10.1145/2533712

[139] Ben-Dor A, Chor B, Karp R, Yakhini Z. Discovering local structure in gene expression data: The order-preserving submatrix problem. *Journal of Computational Biology*. 2003; 10(3-4): 373-384. https://doi.org/10.1089/10665270360688075

[140] Tanay A, Sharan R, Kupiec M, Shamir R. Revealing modularity and organization in the yeast molecular network by integrated analysis of highly heterogeneous genomewide data. *PNAS*. 2004; 101(9): 2981-2986. https://doi.org/10.1073/pnas.0308661100

[141] Pascut D, Pratama MY, Gilardi F, Giuffrè M, Crocè LS, Tiribelli C. Weighted miRNA co-expression networks analysis identifies circulating miRNA predicting overall survival in hepatocellular carcinoma patients. *Scientific Reports*. 2020; 10: 18967. https://doi.org/10.1038/s41598-020-75945-2

[142] Lu Y, Fang Q, Qi M, Li X, Zhang X, Lin Y, et al. Copy number variation-associated lncRNAs may contribute to the etiologies of congenital heart disease. *Communications Biology*. 2023; 6: 189. https://doi.org/10.1038/s42003-023-04565-z

[143] Nangraj AS, Selvaraj G, Kaliamurthi S, Kaushik AC, Cho WC, Wei DQ. Integrated PPI- and WGCNA-retrieval of hub gene signatures shared between Barrett's esophagus and esophageal adenocarcinoma. *Frontiers in*

Pharmacology. 2020; 11: 00881. https://doi.org/10.3389/fphar.2020.00881

[144] Xu M, Ouyang T, Lv K, Ma X. Integrated WGCNA and PPI network to screen hub genes signatures for infantile hemangioma. *Frontiers in Genetics*. 2020; 11: 614195. https://doi.org/10.3389/fgene.2020.614195

[145] Zhao Y, Ma T, Zou D. Identification of unique transcriptomic signatures and hub genes through RNA sequencing and integrated WGCNA and PPI network analysis in nonerosive reflux disease. *Journal of Inflammation Research*. 2021; 14: 6143-6156. https://doi.org/10.2147/JIR.S340452

[146] Xu Z, Zhou T, Wang Y, Zhu L, Tu J, Xu Z, et al. Integrated PPI- and WGCNA-retrieval of hub gene signatures for soft substrates inhibition of human fibroblasts proliferation and differentiation. *Aging*. 2022; 14(17): 6957-6974. https://doi.org/10.18632/aging.204258

[147] Wu L, Chen Y, Wan L, Wen Z, Liu R, Li L, et al. Identification of unique transcriptomic signatures and key genes through RNA sequencing and integrated WGCNA and PPI network analysis in HIV infected lung cancer. *Cancer Medicine*. 2023; 12(1): 949-960. https://doi.org/10.1002/cam4.4853

[148] Zheng G, Zhang C, Zhong C. Identification of potential prognostic biomarkers for breast cancer using WGCNA and PPI integrated techniques. *Annals of Diagnostic Pathology*. 2021; 50: 151675. https://doi.org/10.1016/j.anndiagpath.2020.151675

[149] Caruccio L, Cirillo S, Deufemia V, Polese G. Efficient validation of functional dependencies during incremental discovery. In: Greco S, Lenzerini M, Masciari E, Tagarelli A. (eds.) *SEBD 2021: The 29th Italian Symposium on Advanced Database Systems, September 5-9, 2021, Pizzo Calabro (VV), Italy*. Aachen, Germany: CEUR Workshop Proceedings; 2021. https://ceur-ws.org/Vol-2994/paper1.pdf

[150] Caruccio L, Cirillo S, Deufemia V, Polese G. Efficient discovery of functional dependencies from incremental databases. In: Pardede E, Indrawan-Santiago M, Haghighi PD, Steinbauer M, Khalil I, Kotsis G. (eds.) *The 23rd International Conference on Information Integration and Web Intelligence*. New York, United States: Association for Computing Machinery; 2021. p.400-409. https://doi.org/10.1145/3487664.3487719

[151] Schirmer P, Papenbrock T, Kruse S, Hempfing D, Meyer T, Neuschäfer-Rube D, et al. DynFD: Functional dependency discovery in dynamic datasets. In: Herschel M, Galhardas H, Reinwald B, Fundulaki I, Binnig C, Kaoudi Z. (eds.) *Proceedings of the 22nd International Conference on Extending Database Technology (EDBT), March 26-29, 2019*. Konstanz, Germany: Open Proceedings; 2019. p.253-264. https://doi.org/10.5441/002/edbt.2019.23

[152] Caruccio L, Cirillo S. Incremental discovery of imprecise functional dependencies. *Journal of Data and Information Quality*. 2020; 12(4): 19. https://doi.org/10.1145/3397462

[153] Bläsius T, Friedrich T, Lischeid J, Meeks K, Schirneck M. Efficiently enumerating hitting sets of hypergraphs arising in data profiling. *Journal of Computer and System Sciences*. 2022; 124: 192-213. https://doi.org/10.1016/j.jcss.2021.10.002

[154] Liu J, Ye Z, Yang X, Wang X, Shen L, Jiang X. Efficient strategies for incremental mining of frequent closed itemsets over data streams. *Expert Systems with Applications*. 2022; 191(1): 116220. https://doi.org/10.1016/j.eswa.2021.116220

[155] Douglass EFJr, Allaway RJ, Szalai B, Wang W, Tian T, Fernández-Torras A, et al. A community challenge for a pancancer drug mechanism of action inference from perturbational profile data. *Cell Reports Medicine*. 2022; 3(1): 100492. https://doi.org/10.1016/j.xcrm.2021.100492

[156] Malewicz G, Austern MH, Bik AJC, Dehnert JC, Horn I, Leiser N, et al. Pregel: A system for large-scale graph processing. In: *SIGMOD '10: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data*. New York, United States: Association for Computing Machinery; 2010. p.135-146. https://doi.org/10.1145/1807167.1807184

[157] Yan D, Bu Y, Tian Y, Deshpande A. Big graph analytics platforms. *Foundations and Trends in Databases*. 2017; 7(1-2): 1-195. https://doi.org/10.1561/1900000056

[158] Kyrola A, Blelloch G, Guestrin C. GraphChi: Large-scale graph computation on just a PC. In: *10th USENIX Symposium on Operating Systems Design and Implementation (OSDI '12)*. USENIX Association; 2012. p.31-46. https://www.usenix.org/system/files/conference/osdi12/osdi12-final-126.pdf

[159] Roy A, Mihailovic I, Zwaenepoel W. X-Stream: Edge-centric graph processing using streaming partitions. In: *SOSP '13: Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*. New York, United States: Association for Computing Machinery; 2013. p.472-488. https://doi.org/10.1145/2517349.2522740

[160] Yan D, Cheng J, Lu Y, Ng W. Effective techniques for message reduction and load balancing in distributed graph computation. In: *WWW '15: Proceedings of the 24th International Conference on World Wide Web*. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee; 2015. p.1307-1317. https://doi.org/10.1145/2736277.2741096

[161] Yan D, Cheng J, Xing K, Lu Y, Ng W, Bu Y, et al. Pregel algorithms for graph connectivity problems with performance guarantees. *Proceedings of the VLDB Endowment*. 2014; 7(14): 1821-1832. https://doi.org/10.14778/2733085.2733089

[162] Lu Y, Cheng J, Yan D, Wu H. Large-scale distributed graph computing systems: An experimental evaluation. *Proceedings of the VLDB Endowment*. 2014; 8(3): 281-292. https://doi.org/10.14778/2735508.2735517

[163] Ching A, Edunov S, Kabiljo M, Logothetis D, Muthukrishnan S. One trillion edges: Graph processing at Facebook-scale. *Proceedings of the VLDB Endowment*. 2015; 8(12): 1804-1815. https://doi.org/10.14778/2824032.2824077

[164] Yan D, Cheng J, Lu Y, Ng W. Blogel: A block-centric framework for distributed computation on real-world graphs. *Proceedings of the VLDB Endowment*. 2014; 7(14): 1981-1992. https://doi.org/10.14778/2733085.2733103

[165] Yan D, Cheng J, Özsu MT, Yang F, Lu Y, Lui JCS, et al. A general-purpose query-centric framework for querying big graphs. *Proceedings of the VLDB Endowment*. 2016; 9(7): 564-575. http://www.vldb.org/pvldb/vol9/p564-yan.pdf

[166] Zhang Q, Yan D, Cheng J. Quegel: A General-purpose system for querying big graphs. In: *SIGMOD '16: Proceedings of the 2016 International Conference on Management of Data*. New York, United States: Association for Computing Machinery; 2016. p.2189-2192. https://doi.org/10.1145/2882903.2899398

[167] Yan D, Cheng J, Chen H, Long C, Bangalore P. Lightweight fault tolerance in Pregel-like systems. In: *ICPP '19: Proceedings of the 48th International Conference on Parallel Processing*. New York, United States: Association for Computing Machinery; 2019. Article 69. https://doi.org/10.1145/3337821.3337823

[168] Yuan L, Yan D, Qu W, Adhikari S, Khalil J, Long C. T-FSM: A task-based system for massively parallel frequent subgraph pattern mining from a big graph. *SIGMOD*. [Postprint] 2023.

[169] Yoshida Y. Almost linear-time algorithms for adaptive betweenness centrality using hypergraph sketches. In: *KDD '14: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, United States: Association for Computing Machinery; 2014. p.1416-1425. https://doi.org/10.1145/2623330.2623626

[170] Wu H, Cheng J, Huang S, Ke Y, Lu Y, Xu Y. Path problems in temporal graphs. *Proceedings of the VLDB Endowment*. 2014; 7(9): 721-732. https://www.vldb.org/pvldb/vol7/p721-wu.pdf

[171] Wu H, Huang Y, Cheng J, Li J, Ke Y. Reachability and time-based path queries in temporal graphs. In: *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*. California, United States: IEEE Computer Society; 2016. p.145-156. https://doi.org/10.1109/ICDE.2016.7498236

[172] Takaguchi T, Yano Y, Yoshida Y. Coverage centralities for temporal networks. *The European Physical Journal B*. 2016; 89(2): 35. https://doi.org/10.1140/epjb/e2016-60498-7

[173] Yang Y, Yan D, Wu H, Cheng J, Zhou S, Lui JCS. Diversified temporal subgraph pattern mining. In: *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, United States: Association for Computing Machinery; 2016. p.1965-1974. https://doi.org/10.1145/2939672.2939848

[174] Wu H, Cheng J, Lu Y, Ke Y, Huang Y, Yan D, et al. Core decomposition in large temporal graphs. In: *2015 IEEE International Conference on Big Data (Big Data)*. California, United States: IEEE Computer Society; 2015. p.649-658. https://doi.org/10.1109/BigData.2015.7363809

[175] Shi X, Chen Z, Wang H, Yeung DY, Wong WK, Woo WC. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Cortes C, Lee DD, Sugiyama M, Garnett R. (eds.) *NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*. Massachusetts, United States: MIT Press; 2015. p.802-810. https://dl.acm.org/doi/10.5555/2969239.2969329

[176] Wu Z, Pan S, Chen F, Long G, Zhang C, Yu PS. A comprehensive survey on graph neural networks. *arXiv* [Preprint] 2019. https://doi.org/10.48550/arXiv.1901.00596 [Accessed 21 March 2023].

[177] Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. *arXiv* [Preprint] 2017. Version 4. https://doi.org/10.48550/arXiv.1609.02907v4

[178] Fout A, Byrd J, Shariat B, Ben-Hur A. Protein interface prediction using graph convolutional networks. In: von Luxburg U, Guyon I, Bengio S, Wallach H, Fergus R. (eds.) *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems*. New York, United States: Curran Associates; 2017. p.6530-6539. https://dl.acm.org/doi/10.5555/3295222.3295399

[179] Yuan Y, Bar-Joseph Z. GCNG: Graph convolutional networks for inferring cell-cell interactions. *bioRxiv* [Preprint] 2019. https://doi.org/10.1101/2019.12.23.887133 [Accessed 21 March 2023].

[180] Chereda H, Bleckmann A, Kramer F, Leha A, Beissbarth T. Utilizing molecular network information via graph convolutional neural networks to predict metastatic event in breast cancer. In: Röhrig R, Binder H, Sax U,

Zapf A, Prokosch H-U, Schmidtmann I, et al. (eds.) *Studies in Health Technology and Informatics Volume 267.* Amsterdam, Netherlands: IOS Press; 2019. p.181-186. https://doi.org/10.3233/SHTI190824

[181] Ma T, Zhang A. Incorporating biological knowledge with factor graph neural network for interpretable deep learning. *arXiv* [Preprint] 2019. Version 1. https://doi.org/10.48550/arXiv.1906.00537v1

[182] Kumar M, Alshehri M, AlGhamdi R, Sharma P, Deep V. A DE-ANN inspired skin cancer detection approach using fuzzy C-means clustering. *Mobile Networks and Applications.* 2020; 25(4): 1319-1329. https://doi.org/10.1007/s11036-020-01550-2

[183] Kumar S, Gupta SK, Kumar V, Kumar M, Chaube MK, Naik NS. Ensemble multimodal deep learning for early diagnosis and accurate classification of COVID-19. *Computers and Electrical Engineering*. 2022; 103: 108396. https://doi.org/10.1016/j.compeleceng.2022.108396