HRGRN: enabling graph search and integrative analysis of Arabidopsis signaling transduction, metabolism and gene regulation networks

Speaker: Xinbin Dai
The Samuel Roberts Noble Foundation
Plant & Animal genome Conference (PAG), 2016

Challenging of utilizing known biological interaction data

- The interaction data is heterogeneous
  - Protein-protein interaction, TF-target, miRNA-target, enzyme-compound, and transporter-target
  - Generated from different experiment methods

- and is scattered
  - Low-throughput experiment data, literature
  - High-throughput experiment data, Microarray, RNA-seq, ChIP-chip
  - Predicted interactions using bioinformatics software, Gene co-expression analysis and promoter motif analysis

We need full knowledges of related biological interactions to understand gene function

Complex biological networks

- Nodes: Genes, proteins, non-coding RNAs and small compounds
- Interactions among these elements

Difficulty in utilizing the heterogeneous and scattered biological interaction data for gene function annotation

Example: Is there any relationship between gene A and gene D?

Edge 1: Protein-protein interaction (BioGrid/AtPIN database)

Edge 2: co-expressed gene pair (predicted)

Edge 3: TF-target gene regulation (literature)

HRGRN utilizes graph model to integrate these heterogeneous and scattered interaction data

- Node: gene, protein, small RNA, and compound
- Seven type of edges: Heterogeneous interaction data are organized by biological definition:
  - Protein-protein interaction: BioGrid and AtPIN database
  - Compound-protein interaction: curation from literature
  - Transcription factor-target: curation from literature
  - Small RNA-target: prediction from psRNATarget and curation
  - Transporter-substrate: transporter prediction and TCDB database
  - Enzyme-compound: KEGG database
  - Co-expressed gene pair: data analysis from transcriptomic data

Technical Implementation of HRGRN

- Neo4j database: 3-4 orders of magnitude faster than SQL database for graph path search between nodes
  - Host node and interaction data
  - Provide framework of graph search algorithm for HRGRN in Java

- Cytoscape.js: a HTML5 JavaScript library on front end, which display node/edge in browser.

- Customized code in Groovy/JavaScript (front-end) and Java (graph search at the back-end)
Case #1: HRGRN provides all associated nodes and interactions of a specific gene in a graph of the neighborhood

- Individual gene-centered sub-network
- Graph traversal algorithms in Neo4j database - Breadth-first search
- Customized Java code for our defined edge types and other properties during search

Graph search and visualization are user-customizable

- Highlight node by keyword
- Change color for highlighting
- Toggle predicted/validated edge display
- Change graph layout
- Export graph figure

Change graph path search option:
- Type of interaction by biological definition
- Validate or predicted interaction
- Quantified similarity of co-expression pattern between genes

Case #2: HRGRN can discover “unknown” relationships between genes

- The relationships between ATCUL3 and IAA28 will be skipped during a traditional SQL database query
- Shortest path search algorithm in unweighted graph model
- Searching behavior is customizable in terms of interaction properties: biological type, evidence and etc.
- Future: Dijkstra’s algorithm in weighted model

Connecting to Araport --- Step 1: REST web service

- We developed REST interface for each service in HRGRN
- Example: searching “unknown” relationship between genes (case #2)
  - When user visits http://plantgrn.noble.org/hrgrn/path?hasParams=T&node1=np02084&steps=3&node2=np12356&pathalg=allSimplePaths&PPI_validated=T&GENEEXPREGU_validated=T&COEXP_predicted=T
  - Web site will generate JSON code as below instead of full HTML5 web page:
Connecting to Araport --- Step 2: proxy portal

• A python proxy script providing common interface for upstream web services (Araport team developed)
• GitHub account
• https://www.araport.org/api-explorer

Connecting to Araport --- Step 3: Science App

https://www.araport.org/apps/enkif/hrgrn-app

• Araport can generate an individual-gene centered subnetwork based on REST web service from HRGRN
• No user customization panel

Suggestion for Araport

• Supporting more languages for Proxy adapter, e.g., Java, JavaScript and Perl.

Acknowledgement

The Noble Foundation
• Patrick Zhao
• Tingsong Liu
• Jun Li
• Zhaohong Zhuang
• Junil Chang
• Wenchao Zhang
• Irina Belyaeva
• Jason Miller
• Chris Town
• Vivek Krishnakumar

Open Source Community
• CentOS Linux
• Oracle Java
• Resin, Java web server
• Groovy, Java-based script language
• Neo4j, open source Graph database
• Cytoscape.js, HTML5 web front-end
• JQuery, JavaScript framework