*GOAL2.0*: Gene Ontology Analyzer v2.0

Manual

Scientific Data Mining Team

# Digital Technologies Research Centre

National Research Council Canada

1200 Montreal Road, Ottawa, Ontario, Canada K1A 0R6

1

1 Introduction

[2 Manual Overview 3](#_Toc82426738)

[3 Preliminaries 3](#_Toc82426739)

[3.1 System Requirements 3](#_Toc82426740)

[3.2 Installation 3](#_Toc82426741)

[3.3 Configuring *GOAL2.0* 4](#_Toc82426742)

[4 Running *GOAL2.0* 5](#_Toc82426743)

[4.1 Using GUI: Input 5](#_Toc82426744)

[4.2 Using GUI: User-Defined data files 8](#_Toc82426745)

[4.3 Output 8](#_Toc82426746)

[4.4 Command Line Interface (CLI) 9](#_Toc82426747)

[5 Understanding *GOAL2.0* Input and Output 10](#_Toc82426748)

[5.1 Input Files 10](#_Toc82426749)

[5.2 Output File 11](#_Toc82426750)

[6 Developer Information (Advanced) 12](#_Toc82426751)

[6.1 Use as Plug-in 12](#_Toc82426752)

[6.2 Adding and Retrieving Species 13](#_Toc82426753)

[7 CASE STUDY 13](#_Toc82426754)

[8 REFERENCES 14](#_Toc82426755)

1. Introduction

*GOAL2.0* is an application that groups genes based on their relationships defined in Gene Ontology (GO: <http://www.geneontology.org/>), transcription factors (TF) that co-regulate them, their interaction with microRNA, their association with disease, drug, KEGG (<http://www.genome.jp/kegg/pathway.html>) and reactome pathways. GO relationships are derived by analyzing annotation and ontology (obo format) files. Once the genes are grouped by their association with GO, TF, microRNA, disease, drug or KEGG and reactome pathways, they are analyzed for statistical significance using p-values and adjusted p-values. The results interface then displays the sorted data and provides links to the gene ontology and other websites.

1. Manual Overview

This manual covers all one needs to know to use *GOAL2.0* successfully. First it covers installation and configuration, then describes how to run *GOAL2.0*, and finally includes a section for developers.

1. Preliminaries

## System Requirements

*GOAL2.0* is implemented entirely in Java. It is available as an executable jar file and works with Java SDK 14 or later (<http://www.java.com/en/download/index.jsp>) under Windows and Linux operating system. The java virtual machine must be set properly before running *GOAL2.0*. This is done automatically with the runGOAL\_Windows.bat or runGOAL\_linux.sh for Window system or Linux system respectively.

## Installation

Download the GOAL2.0 from GitHub (<https://github.com/DT-NRC/GOAL2.0>), and unzip it to the desired install location. Then unzip the “diseaseGeneFiles.zip”, “drugGeneFiles.zip”, “goSynonyms.zip”, “keggFiles.zip”, “microRNAGeneFiles.zip”, “reactomeGeneFiles.zip”, “tfGeneFiles.zip” and “gofiles1.zip” to “gofiles5.zip” and move the unzipped annotation files into corresponding folder under: GOAL2.0\Deployment\Data. The structure should be exactly as the Figure 1.

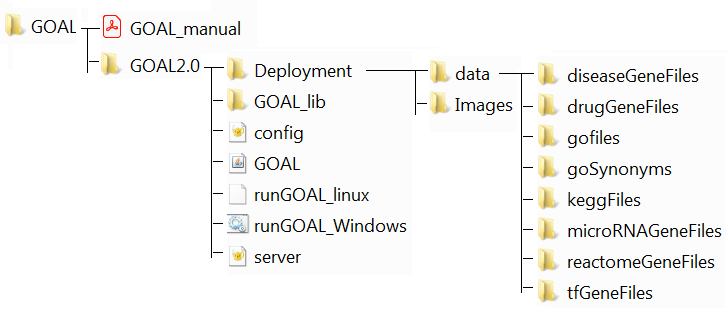


Figure 1: Directory Tree Structure

## Configuring *GOAL2.0*

A configuration file with default values is generated at runtime. If the user wishes to use non-default directories/download, changes may be made in the file. In most cases, no change in configuration is needed.

G*OAL* uses a simple config.prop file (Figure 2) to store directories and URLs. If this file does not exist it will be created with default values at first use. This file is in the standard java property file format, with keys on the left and the values on the right. It contains the annotation files, ontology files and the download URL, additional URL tags for downloading annotation/ontology files

#Wed Dec 02 13:59:15 EST 2020

DRUG\_Dir=

Ontology\_Ext=

MirGene\_Dir=

KEGG\_SYN\_URL=https\://ftp.ncbi.nlm.nih.gov/gene/DATA/GENE\_INFO/

MirGene\_URL=

ReactomeGene\_URL=https\://reactome.org/download/current/

KEGG\_URL=ftp\://ftp.genome.jp/pub/kegg/genes/organisms/

Ontology\_URL=http\://current.geneontology.org/ontology/

KEGG\_Dir=

KEGG\_URL\_PATHWAY=http\://rest.kegg.jp/link/pathway/

ReactomeGene\_Dir=

DiseaseGene\_Dir=

DiseaseGene\_URL=https\://www.disgenet.org/static/disgenet\_ap1/files/downloads/

Annotation\_URL=http\://release.geneontology.org/2019-06-09/annotations/

Synonym\_Dir=

Annotation\_Dir=

Annotation\_Ext=

Ontology\_File=

TF\_Dir=

KEGG\_Ext=

TF\_URL=https\://www.grnpedia.org/trrust/data/

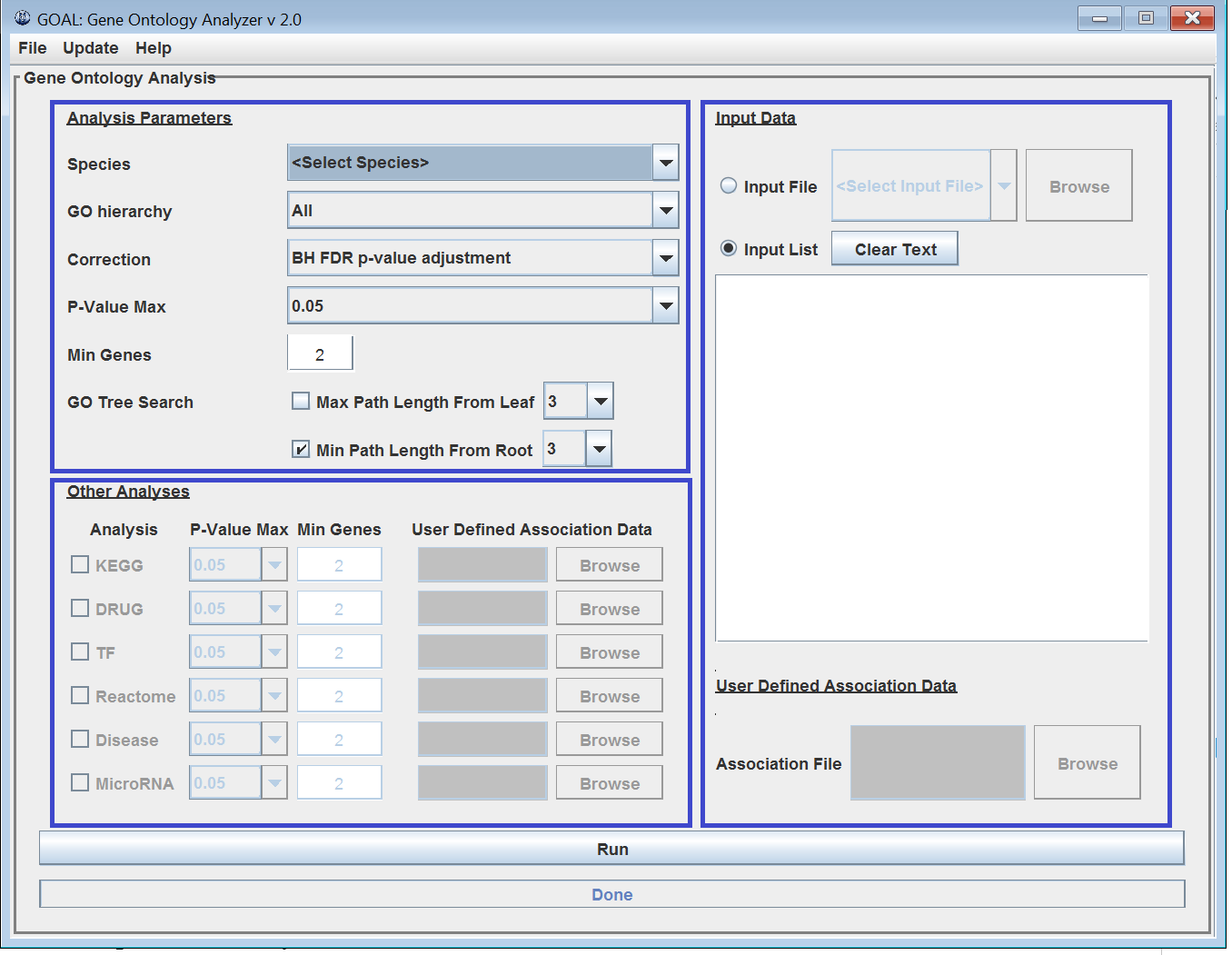
DRUG\_URL=https\://www.dgidb.org/data/monthly\_tsvs/2020-Nov/

Figure 2. Example configuration.

1. Running *GOAL2.0*

## Using GUI: Input

To run *GOAL2.0* double click on runGOAL\_Windows.bat or runGOAL\_Linux. **Figure 3** shows the Graphical User Interface (GUI). It has settings for file input and output, algorithm parameters, a run button, and a progress indicator.



**Figure 3. *GOAL2.0* GUI using Java's AWT and Swing**

When the GUI starts up, it first runs preprocessing in the background. During this time users cannot run *GOAL*. However this should only take at most a few seconds and does not disable parameter entry.

On the top of the GUI is the manual bar, which includes “File”, “Update”, and “Help”. “File” allows a user to exit the program; “Update” allows a user to update corresponding background association data files for each module. When GOAL2.0 is first launched in a computer, users should click on the “Update All” button at the bottom right, which allows the system to update the background association data files for all modules (Figure 4); this will take up to 45 minutes to an hour or so depending on computer while its CPU and network are free from other processes. In this regard, we highly recommend performing “Update All” free from other computations or network activities, probably it is good idea to perform at the end of the work day and leave the computer run for overnight. Alternatively, users are encouraged to download all up-to-date (within last six month) association files from GitHub https://github.com/DT-NRC/GOAL2.0.

The slowness of “Update All” is mainly caused by updating association data for reactome-gene associations, which takes 95% time of overall updating. Users can also update association files for individual modules by selecting the species (e.g. *Arabidopsis thaliana*) or “Update All” to update as indicated in Figure 4b and press the “Update” button for example of update the GO-gene association files. **Practically, users are highly encouraged to perform any update at fresh start of *GOAL2.0* to avoid possible complication**.



Figure 4. Update manager panel. a) “Update All” does not require selecting any module. b) Selecting *Arabidopsis thaliana* for GO-gene association file update.

“Help” provides links to the “User Manual”, “About GOAL” and “Run Example” as a short demo, which will allow user to run GOAL Analyzer by given identifiers (IDs) of 10 genes.

The main body of the GUI contains three panels (Figure 3),

1. “Analysis Parameters” panel: which allows user to make selections, set parameters for Gene Ontology Analysis. The “*Max Path Length From Leaf”* and “*Min Path Length From Root”* are check boxes, they can be used separately or together, at least one must be checked which define the algorithm used. At startup all parameters are set to default. The only essential parameter is the species, since no results will be displayed if it does not match the species of the input genes. **Table 1** summarizes the input parameters.

|  |  |
| --- | --- |
| Table 1: Overview of input parameters. | |
| **Parameter** | **Description** |
| Species | Species of the input genes. This is required to determine the correct annotation file. |
| GO hierarchy | Allows a choice of either the whole ontology or a specific GO category: biological process, cellular component or molecular function. |
| Correction | Multiple testing correction type for correcting p-values. The choices are Bonferroni, Bonferroni step-down, Benjamini False Discovery Rate and BH (Benjamini and Hochberg) FDR p-value adjustment listed in order of decreasing stringency. For example, Bonferroni will have fewest false positives at the cost of increased false negatives. |
| P-value | This parameter sets the p-value filter on the results. The results with p-values equal to or better than the specified value will be shown. |
| Min Genes | This parameter sets a filter for the minimum number of gene in the results. Only groupings with equal to or more than the specified number of genes will be shown. |
| GO tree search | |
| Max path Length From Leaf | Maximum number of steps of parental relationships considered during grouping. Genes are matched to their associated GO term and then trace the parental path to related GO terms. The lower this value is the more specific the GO terms are. |
| Min Path Length From Root | Minimum number of GO paths from the root (e.g. Biological Process) that a GO term must be for it to be included in the analysis. The higher this value is the more specific the GO terms are. |

1. “Other Analysis” panel, allows user to set parameters for other modules that run after GO analysis that groups genes by their associated transcription factors (TF), KEGG and Reactome pathways, Drug and/or Disease, and/or their interactions with MicroRNA based on a provided annotation. User has to set thresholds for p-values and minimum number of genes in each group.
2. “Input Data” panel, will allow user to select input data by clicking on the “browse” button, or paste the gene identifiers the text field.

If user select species as “User Defined File” in the “Analysis Parameters” panel, the “User Defined Association Data” will be active in “Other Analysis” panel and “Input Data” panel, which contains “Input Data” and “User Defined Association Data”, the “Input Data” panel allows user for inputting gene list by specifying an input file or entering or pasting a list of genes. The “User Defined Association Data” at the bottom of the panel allows user to specify gene ontology background association data. It becomes active only when the user select “User Defined File” in “Analysis Parameters” panel. The background data are used for p-value calculations and could improve accuracy for specific situations. For example, if a user is analyzing a cluster derived from a microarray experiment, the probe set (gene IDs) printed on the microarray could make a better background than the entire genome, which is used by default.

The default type of gene ID accepted is the default for the database for the selected species and given under the text field for gene IDs. Key species (*Arabidopsis*, Yeast, Human, Mouse, etc.) have greater gene ID support that allows most common types of gene IDs (such as gene symbols and Ensembl database IDs), supported by respective ID synonym files at $:\GOAL2.0 \Deployment\data\goSynonyms. When such species is selected, “various IDs accepted” shows under the text field. The input file format is specified in section 5.1.

After all this information is entered, press the “Run” button on the bottom to start the analysis.

## Using GUI: User-Defined data files

The format of user-defined association data files are identical to those existing data files in respective modules. The file is tab delimited text file. Taking gene ontology as an example, gene IDs are at the second and third columns, and the gene ontology ID, description and categories are at the fifth, sixth, and ninth columns. Otherwise the data in various other columns are for user information, will not be used by the software. Even though users can browse the data file anywhere in the local computer, it is highly recommended to place the user-defined files in the same folders as the other data files of the same modules to avoid confusion.

## Output

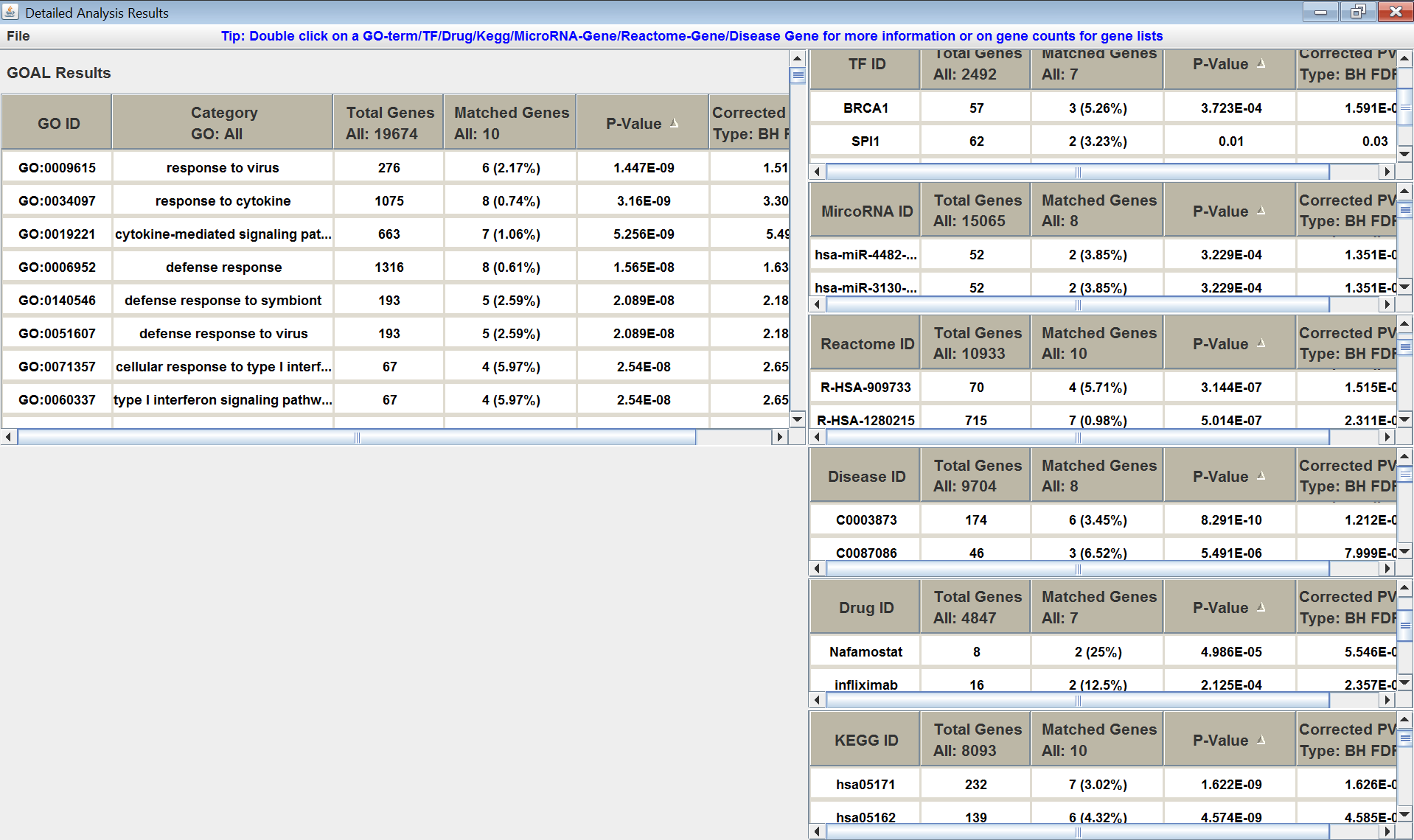


Figure 5. *GOAL2.0* after execution.

Soon after all analyses in various modules complete, the output panel appear (**Figure 5**); otherwise, a dialog box will appear with corresponding popup information, or an error message if the system encounters an exception. Any query genes not found from the association data file will be displayed in separate windows, one for each analysis (GO, KEGG, Drug, TF-gene association, MicroRNA, Reactome and Disease association).

Next, it displays up to seven result panels in the Output window (Figure 5), GO result on the left and the other results on the right, each containing a table. The tables are sortable and can be copied to clipboard. The first column is the group identifiers (e.g. GO IDs, KEGG IDs, Drug IDs, etc.), with a second column is the GO description for GO results. When the ID in first column is double clicked, a browser will open and display the entry in corresponding to a database. For example, the AmiGo 2 (http://amigo.geneontology.org/amigo/term/GO:0044267), KEGG (<http://www.genome.jp/dbget-bin/www_bget?hsa00970>), Drug

([https://www.dgidb.org/drugs/GUTTIFERONE%20K#\_interactions](https://www.dgidb.org/drugs/GUTTIFERONE%20K%23_interactions)), TF (<https://www.ncbi.nlm.nih.gov/gene/?term=MAP3K5>), MicroRNA (<http://www.mirbase.org/cgi-bin/mirna_entry.pl?acc=hsa-miR-126-3p>), Reactome (<http://www.reactome.org/content/detail/R-HSA-6791403>), and Disease (<https://www.disgenet.org/browser/0/1/0/C0027746/>). The third column shows the number of known genes in the group from the background data, and the fourth column shows the number of genes from input data in this group. When double clicked on the numbers, the actual gene IDs are displayed in a pop up panel, which can be saved to a file (see 5.2 Output File for details). Clicking on each individual gene ID leads to NCBI Entre gene card. The next two columns include the p-value and corrected p-value of each group calculated based on the numbers in the previous two columns.

A menu bar is also included in the above tables. They allow users to save the data to a file or close the window. Any number of windows can be open at a time so multiple results can be displayed and compared.

## Command Line Interface (CLI)

**Figure 6** shows a Command Line Interface (CLI) version of the GUI shown in **Figure 3**. Users may prefer this when running *GOAL2.0* as part of a tool suite. Default values are made available through run*GOAL*-windows.bat or run*GOAL*\_linux. Either --gui, --client, --server or *GOAL2.0* parameters must be included. If the *GOAL* parameters are used, all input and output options must be used as well (except for others, e.g. optional for TF-gene, KEGG etc.). --gui and --server ignore other options. --client may be combined with –gui to display the client GUI. Currently one parameter, degree, is used for both *Max Path Length From Leaf* and *Min Path Length From Root* seen in the GUI. Also p-values, minGenes and background are equal for all analyses.

Figure 6: Output of *GOAL2.0* –help

Startup:

-h, --help

-v, --version

IO:

-in, --input-file=FILE path to gene names data file

-tf, --tf-gene file=FILE path to TF-gene annotation (Optional)

-kegg, --kegg-gene file=FILE path to KEGG annotation (Optional)

-drug, --drug-gene association file=FILE path to drug annotation (Optional)

-reactome, --reactome-gene association file=FILE path to reactome annotation (Optional)

-disease, --disease-gene association file=FILE path to disease-gene association (Optional)

-microRNA, --microRNA-gene association file=FILE path to microRNA-gene association (Optional)

-b, --background-file=FILE path to background gene names data file (Optional, annotation will be used if null)

-o, --output-dir=DIR path to output results

UI:

--gui run with gui. Implies necessary fields

*GOAL*:

--species=STR name of species (Must have annotation)

--hierarchy=STR biological\_process/molecular\_function/cellular\_component

--correction=STR Benjamini/Bonferroni/BonferroniSD/BonferroniSD\_AdjPvalue

--maxParental=INT Max path length from leaf

--minRoot=INT Min path length from root

--pvalue=DOUBLE Max P value

--minGenes=INT Min genes required in a group

Client/Server:

--client run client. Implies necessary fields

--server Run server. Port required

-s, --serverIp=STR IP address of started RMI server

-p, --port=INT Port used for RMI server

Configuration:

--config-file Path to configuration file

1. Understanding *GOAL2.0* Input and Output

## Input Files

There is a variety of different input files, each with different formats. They can be categorized into the following, 1) gene list input, 2) GO files, 3) gene synonyms, 4) TF-gene annotations, 5) KEGG and 6) reactome pathway, 7) drug-gene interaction, 8) microRNA-gene and 9) disease-gene association annotations.

Gene list input is the input of genes to be analyzed by *GOAL2.0*. The format of these files can be simply a text file (\*.txt, \*.dat) or CSV file (\*.csv). The delimiter can be a tab, newline, space, comma, or colon character.

The GO files folder includes the Gene Ontology annotation files and ontology files (obo format). Format details are available on the Gene Ontology website at: <http://www.geneontology.org/GO.contents.doc.shtml>.

Gene synonyms files are simple TSV files that contain the gene in the first column, and synonyms in all other columns. These files are named as “syn\_” plus short form of the corresponding GO annotation file, e.g. “syn\_ath.tsv” as synonyms of Arabidopsis thaliana, “syn\_hsa.tsv” as synonyms of Homo sapiens

The KEGG pathway file contains a list of genes and their associated KEGG pathway as first and second column respectively.

Drug and gene interaction file contains a list of genes and the drug ID, which interacted with, as first and second column respectively. It is available at website https://www.dgidb.org/downloads/

The TF-gene association files are available at website: <https://www.grnpedia.org/trrust/data/>, or compiled by the user. As we mentioned earlier, the TF-gene association data can be derived from Chromatin Immunoprecipitation (ChIP) experiments. They can also be compiled from literature search and/or TF databases of well-known and well characterized biological interactions such as TRANSFAC, JASPAR, and RegulonDB, etc. For example, TRANSFAC is a database on eukaryotic transcriptional regulation. The database contains data on transcription factors, their target genes and their experimentally validated binding motifs in genes. The TF-gene association data is represented as a matrix, where the rows correspond to genes and the columns to transcription factors. The entries of the matrix are either 0 or 1, with 1 for a known interaction between a TF and its target gene and 0 for no interaction.

Reactome pathway and gene association file contains a list of genes and the reactome pathway ID the gene involved in, in the first and second column respectively, which is available at <https://reactome.org/download/current/>

Disease and gene association file contains a list of genes and the disease the gene related to, in the first and second column respectively, which is available at <https://www.disgenet.org/static/disgenet_ap1/files/downloads/>

Micro-RNA gene interaction file contains gene IDs and the interactive microRNA IDs in the first and second column respectively, which is available at <http://mirtarbase.cuhk.edu.cn/cache/download/8.0/>

## Output File

Figure 7 shows the output of KEGG pathway analysis. This result is saved by selecting “File -> save KEGG” from menu bar as an individual file. User also can save all the results at once by selecting “File ->save ALL” from menu bar; the files are named with time stamp and module name. Output files are all in CSV (comma-separated values) format with comments at the top of each file providing information about the parameters used to obtain the results. The remainder of the file gives more details similar to the output GUI table with an additional column as description of or the association with the identifier in the second column, and all the matched genes in the group at the end of that row. The output file is sorted based on the p-values.

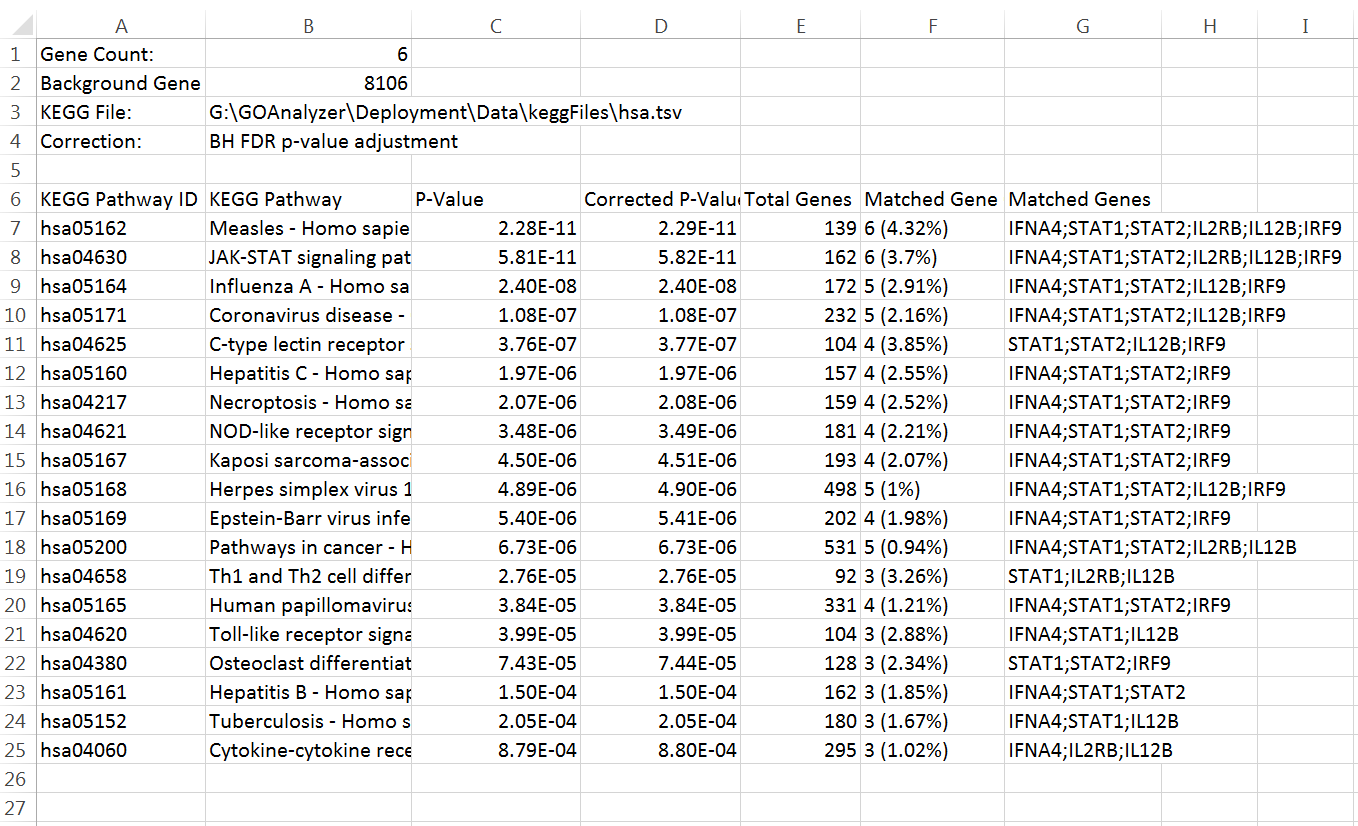


Figure 7: Output of GOAL – KEGG output file

1. Developer Information (Advanced)

This section is for people who would like to use this application as a plug-in.

## Use as Plug-in

*GOAL2.0* was designed for easy integration into other java tools. To run *GOAL2.0* as part of another tool, first the JAR file and GOAL\_lib must be included in the project build path. All the functionality of *GOAL2.0* is made available through a class call *GOAL*Plugin. The way this class can be used is very flexible. The input and output GUIs can be displayed or returned to be integrated in an existing GUI. Either way, they are fully functional by using ActionListeners within the classes created and supplemented with listeners from the *GOAL*Plugin class itself.

If the GUI is not needed, then *GOAL2.0* can be used independently through this class. This is also achieved in two ways, through a regular method call or by returning a started thread as a GOAnalysisTask, TFAnalysisTask, or KEGGAnalysisTask. These classes can be used like normal java threads which throw a PropertyChangeEvent when complete. If the user is unfamiliar with PropertyChangeListeners the *GOAL*Plugin class can be used as a listener and will invoke the results GUI when the thread is complete. Furthermore these methods include versions that run with default parameters if the user is confused by them.

The last functionality provided by this class is the ability to run the *GOAL* server. This way the server can run in the background without the GUI.

If the user does not want to deal with coding at all then *GOAL2.0* can be used through command line with full functionality, with or without GUI.

## Adding and Retrieving Species

As more annotation files are created they may need to be added to *GOAL2.0*. This is done simple by adding a line to the constructor of SpeciesExtMap. The key is the species and the value is the extension of the annotation file.

e.g. m\_mapping.put("Drosophila melanogaster", "fb");

The function getSpecieNames()handles the retrieving of the species names. The Vector<String> returned can be used to populate whatever input GUI that is used.

1. CASE STUDY

To showcase GOAL2.0 in this study, we used the 330 genes published in (Davis, 2020) also available on GOAL GitHub [<https://github.com/DT-NRC/GOAL2.0>]. This set of genes were shown to be differently expressed in micrometastatic cells in human patient-derived-xenigraph (PDX) models using single-cell RNA sequencing (scRNA-seq). **Fig. 8** shows the GOAL analysis results of these 330 genes. GO, TF, microRNA, pathway (KEGG and REACTOME), drug and disease genes associations showed that most of the genes in this set were highly enriched and statistical significant with p-values as low as P-Value 2.925 e-17.

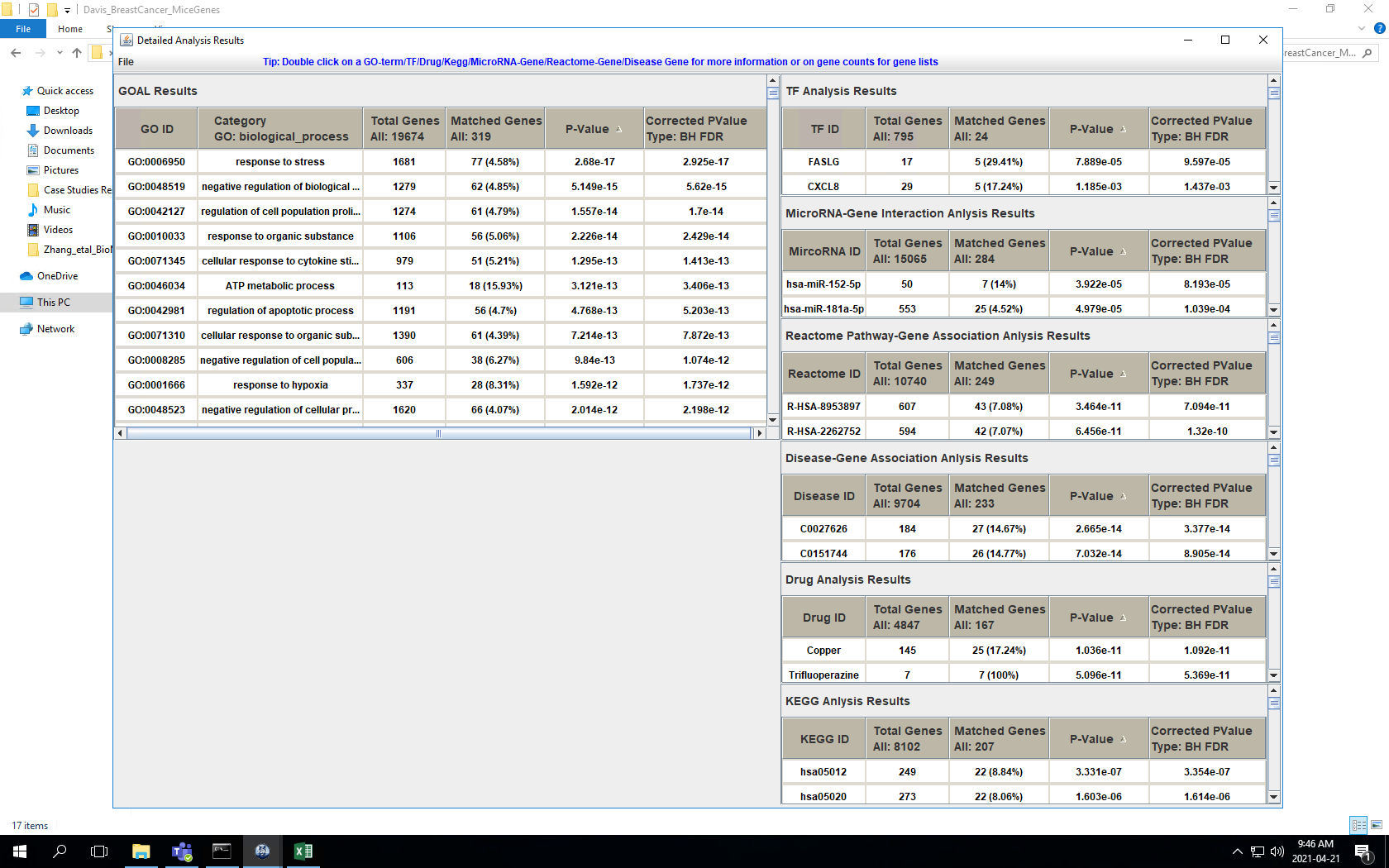
****

Figure 8: GOAL results of the case study

**Fig. 9** shows a network summarizing the biological information embedded in these 330 genes. The network describes the regulatory interactions between TFs, microRNAs, drugs and genes and it is obtained by considering the top five (most statistically significant based on their p-values) TF, microRNA, drugs gene associations in the results Table (Figure 8) and plot using Cytoscape (Shannon, 2003). Figure 8 shows an interaction between, TFs, microNRAs, Drugs and a subset of the 330genes. This regulatory network which can also be seen as a regulatory module, suggests that genes, TFs, and microRNAs involved may belong to the same biological pathways or play a role in the same regulatory process. Indeed pathways analysis show that they are associated to the same REACTOME pathways (R-HAS-8953897: Cellular responses to external stimuli) with pvalue < 7.09e-11. Furthermore, this network may also facilitate the discovery of a therapeutic target, given the interactions between some genes (RP11-551L14.1; AC011294.3; SERPINA3.1; S100A4; AC073325.2; CALM1; RP11-161H23.5) and drugs such as (trifluoperazine, crolibulin and dinaciclib). Last but not least, Disease gene associations suggest that most of these genes are not only link to cancer, but also to prion and Parkinson disease.

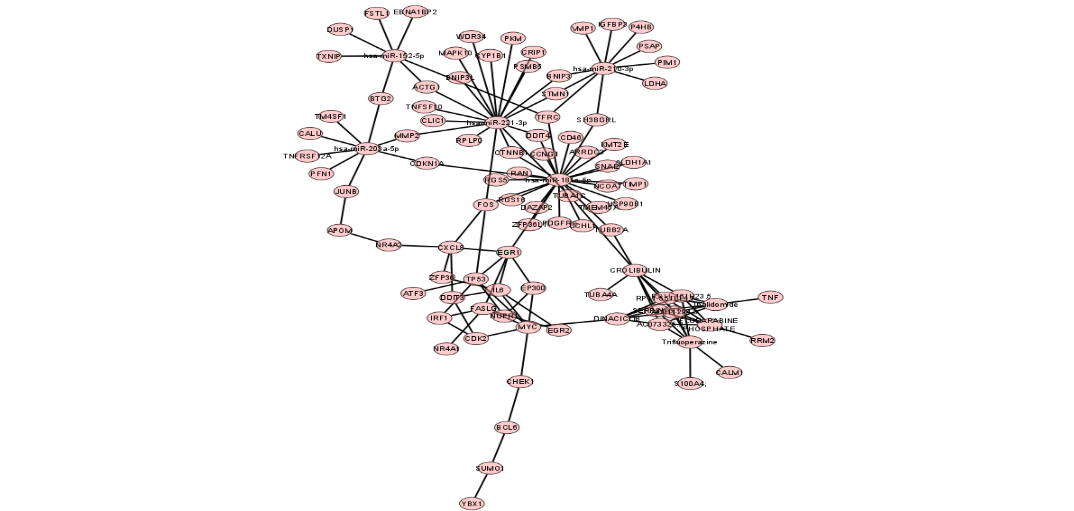
****

Figure 9: Network summary of the GOAL analysis results

1. REFERENCES
2. Davis, R.T. *et al.* (2020) Transcriptional diversity and bioenergetic shift in human breast cancer metastasis revealed by single-cell RNA sequencing. *Nat Cell Biol* 22**,**310–320.
3. Shannon P.et al. (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*, 13, 2498–2504.