A Survey on Computational Approach for Disease-Gene Associations

K. Mary Sudha Rani[1]
[1]Assistant Professor, Department of CSE, CBIT, India

Abstract

Understanding the link between genetic diseases and also the genes associated with them could be a crucial problem for human health. The vast amount of data created from a large number of high-throughput experiments performed within the last few years has resulted in a huge growth in computational methods to handle the disease gene association problem. Nowadays, it's clear that a lot of genetic diseases do not seem to be the consequence of defects present in a single gene. Proteins present together in a community are represented using a PPI network graph. These PPI networks indicate how proteins interact. In this paper, a computational approach for the disease-gene association is devised using Genetic Algorithm and Protein-Protein Interaction Networks (PPI).

Index Terms : genetic disease, PPI network, gene

1. INTRODUCTION TO DISEASE-GENE ASSOCIATION PROBLEM

Associating genes with a specific group of phenotypes, i.e. the genetic diseases is one amongst the key of gene-phenotype association research. Associating genes with genetic diseases and disorders is crucial to recognize the genetic basis of human diseases. The specific research area problem where the genes which are in any way involved in the existence of a given genetic disease are identified is called the disease-gene association problem, or identification of disease genes, or disease gene prediction. Understanding the complex correlation between genes and proteins requires the processing of a vast amount of data from a wide variety of genomic data sources. Thus, computational tools have become critical for the integration, representation, and visualization of heterogeneous biomedical data. To be extremely general, computational disease-gene association methods apply all the possible and potential findings throughout years of research in related areas, using whatever useful information can be found in the literature to associate genes with diseases.

2. RELATED WORKS

INTEGRO: an algorithm for data-integration and disease-gene association

A INTEGRO[1] is an algorithm used for disease gene association and data integration based on the information retrieved from OMIM, NCBI-MedGen, SNOMED-CT, Disease Ontology and DisGeNET. A latest selected version of DisGeNET was used to ensure the reliability and validity of the information.

INTEGRO's pipeline was summarized by the following steps:

1. To establish the relation between the input and the information is annotated in DO to identify a defined term of interest in this the input is referred to as the word Term.
2. To extract attributes related to the Term for example cross-references and DO-id.
3. Then visit the DO's graph to identify the Term with the right attributes contained inside its relationships.
4. Then analyze the cross-references to retrieve external information by OMIM, NCBI-MedGen, SNOMED-CT, Disease Ontology and DisGeNET. And at last, the disease-gene associations and other information such as definition, UMLS codes and treatments are also integrated while excluding redundancies.

   DO is a Directed Acyclic Graph which presents terms linked in the hierarchy and interrelated subtypes. In this attribute, "is_a" was used to identify the root node for a given term. In this DO's graph, the terms become more specific while the depth increases whereas the root nodes are therefore more generic. Here in this graph terms can have multiple values but the term with more similarity and greater depth is chosen discarding others.

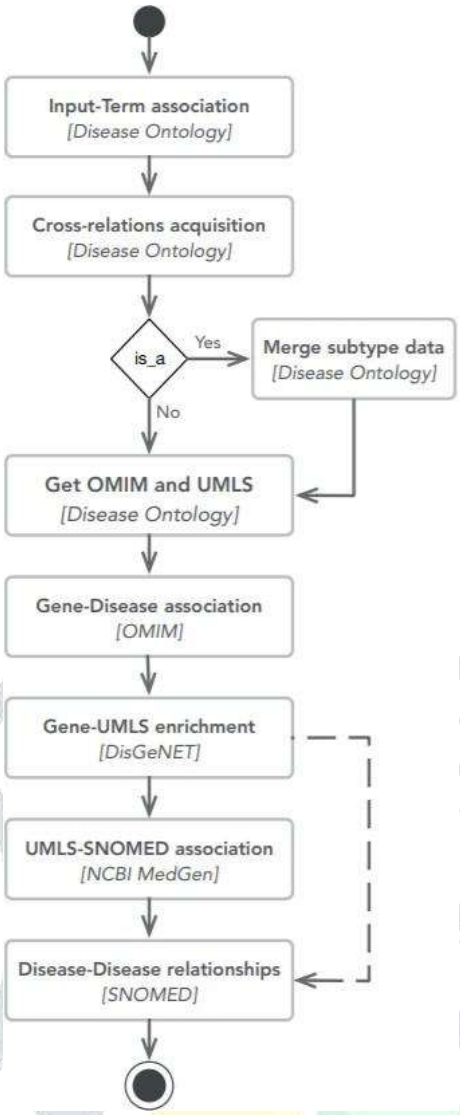

Fig. 1. Non-exhaustive workflow for the INTEGRO's pipeline

The above figure shows the Non-exhaustive workflow for the INTEGRO's pipeline. First, the DOID for a given Term is obtained by using the best match case and then in the next phase of data analysis and subsequent integration few OMIM identifiers which are also referred as cross-references in DO are used to perform Disease gene association using MorbidMap.

Disease-Gene Association Using a Genetic Algorithm

A disease-gene association using a genetic algorithm[2] has been proposed. The genetic algorithm can give an effective optimal solution for many NP problems. The genetic algorithm helps us to provide a population of some candidate solutions for a given problem. These candidate solutions are referred to as individuals or chromosomes. And a fitness function is used to evaluate the fitness of the solution obtained from the Genetic algorithm. This fitness is used to measure how good an obtained solution is for a given problem and a process of crossing and mutations are applied to obtain the new off- springs. The genetic algorithm will follow the below steps. The initial population is generated. This initial population is also considered as the candidate solution. Then we will be measuring the fitness of the solutions. If the fitness obtained is of desired value then the algorithm stops. Even if the number of iterations reaches the number of generations then the algorithm stops. In the next step, we will select the individuals as the parents and apply cross-over followed by mutation to obtain the next generation and flow continues. After obtaining the best community having highest fitness and then after obtaining the best genes it was compared with the CIPHER results which is one of the best disease- gene association frameworks.

Gene-disease association through topological and biological feature integration

A learning model [3] helps in classifying genes as diseased which are related to diseases based on both biological and topological features. When this model is given a list of genes, it classifies between a disease gene and not a disease gene class. The topology of the corresponding PPI network is combined with the various sources to discover similarities that characterize each class. It achieved an area under the receiver operating characteristic (ROC) curve of 0.941 using

Naive Bayes classifier.

A Novel Disease Gene Prediction Method Based on PPI

Function flow-based model[4] consists of two main steps. In the first step, a weighted protein interaction network from protein interactions between genes, and a map of the known disease gene and the candidate disease genes to the protein interaction network is constructed. In the second step, candidate genes are ranked based on the function flow model based on the functional similarity between the candidate genes and known disease genes, and the candidate genes with the highest are suspected to be disease genes.

Function Flow Model Algorithm:

This algorithm works on the principle of guilt by association. If the candidate genes are connected by physical interactions with the disease genes then they are more likely to cause the disease. It is achieved by treating each disease gene as a 'source' of 'functional flow' for a particular disease. Then they simulated the spread of the functional flow through the protein-protein interaction network and then each protein obtained the 'functional score' which corresponds to the amount of 'flow' that the protein has received from the source protein.

The function flow model algorithm is done in three steps:

1. Identify the disease gene in the PPI network
2. Then simulate the function flow to get the flow from the source (known disease genes) for each candidate disease gene as a function score. The score obtained by a candidate gene may be zero if the flow did not reach that protein.
3. Rank the genes according to the functional score, the genes with a high score are considered more likely to be disease genes.

The above algorithm is continued for every known disease gene and the score for each particular candidate gene is updated in iterations.

The process of iteration subject to some rules:

1. A node delivers the function flow preserved in its reservoir to its direct neighbors which are proportional to the weight of the edge between two nodes, the functional flow does not cross the capacity it can pass on.
2. When the disease gene node has enough functional flow in its reservoirs to deliver to the candidate gene. They simplified it by capping the maximum flow that can be passed to the candidate gene to 1 in each iteration irrespective of the weight of the edge between them.

As each candidate disease gene receives the function flow from a source node in all iterations. The source node with enough flow and the function flow pass on at the discrete-time, so the nodes closer to the source node (disease gene) receive more flow than the nodes which are far from the source node (disease gene).

Variables

u - a protein in PPI network

Rt(u) - the amount in the reservoir that a protein u has at time t. Gt(u,v) - the function flow from protein u to protein v at time t. d - Number of iterations

At time 0, only the disease gene node has a reservoir of function flow, the formula is as follows:

In every iteration, the reservoir of each protein will be updated according to the total inflow and total outflow of the node.

$$R_0(u) = \begin{cases} 1, & \text{if } u \text{ is a known disease gene} \\ 0, & \text{otherwise} \end{cases}$$

$$R_t(u) = R_{t-1}(u) + \sum_{v \in N(u)} (g_t(v,u) - g_t(u,v))$$

As the flow will occur from the nodes with more flow to nodes with less flow at time 0. And in each iteration the flow is towards downhill it is given by the following formula.

$$g_t(u,v)=\begin{cases}0, & \text{if } R_{t-1}(u)<R_{t-1}(v)\\ min(w_{u,v}, R_{t-1}(u)\dfrac{w_{u,v}}{\sum_{(u,y)\in E} w_{u,y}}), & \text{otherwise}\end{cases}$$

Finally, the score can be calculated for node u according to the total amount of flow it preserves over d iterations.

$$f(u)=\sum_{t=1}^{d}\sum_{v\in N(u)} g(v,u)$$

8959 proteins and 33528 distinct interactions were obtained using this data and then a weighted protein interaction network was constructed. This method successfully ranked 102 known disease genes in top1 out of all 723 known disease genes.

CONCLUSION

Genes associated with a specific disease may act in separate communities which work with one another or separate communities which overlap. We believe that GA-based computation is capable of finding disease genes working in different communities or in overlapped communities for the following reasons. First, as the evolving populations contain thousands of different communities, which will work with the known disease genes and have the chance to evolve and be in the population at the same time. Genes in all such communities will get high scores as they are often selected in a number of the population's communities over generations, thereby they can increase their scores. Secondly, potential disease genes which are present in more than one community working with disease genes are selected more frequently as they can have a major chance to be in many of the communities of the populations.

REFERENCES

[1] Pietro Cinaglia, Pietro H Guzzi, Pierangelo Veltri (2018), "INTEGRO: an algorithm for data-integration and disease-gene association", 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM).

[2] Koosha Tahmasebipour, Sheridan Houghten (2014) "Disease-Gene Association Using a Genetic Algorithm", 14th International Conference on Bioinformatics and Bioengineering.

[3] Eileen Marie, Hanna Nazar, M. Zaki (2015) "Gene-disease association through topological and biological feature integration", 11th International Conference on Innovations in Information Technology.

[4] Junmin Zhao, Tingting He, Xiaohua Hu, Yan Wang, Xianjun Shen, Minghong Fang, Jie Yuan (2014) "A Novel Disease Gene Prediction Method Based on PPI Network", IEEE International Conference on Bioinformatics and Biomedicine (BIBM).

[5] Xuebing Wu, Rui Jiang, Michael Q Zhang, and Shao Li(2008) "Networkbased global inference of human disease genes" Molecular systems biology, 4(1).