Columbia International Publishing
American Journal of Bioinformatics and Computational Biology
(2015) Vol. 3 No. 1 pp. 1-4
doi:10.7726/ajbcb.2015.1001
Research Article

KNGP: A Network-based Gene Prioritization Algorithm that Incorporates Multiple Sources of Knowledge

Chad Kimmel[1*] and Shyam Visweswaran[1]

Received 2 January 2015; Published online 25 April 2015

Abstract

Background: Candidate gene prioritization is the process of identifying and ranking new genes as potential candidates of being associated with a disease or phenotype. Integrating multiple sources of biological knowledge for gene prioritization can improve performance.

Results: We developed a novel network-based gene prioritization algorithm called Knowledge Network Gene Prioritization (KNGP) that can incorporate node weights in addition to the usually used link weights. The online Web implementation of KNGP can handle small input files while the downloadable R software package can handle larger input files. We also provide several files of coded biological knowledge that can be used by KNGP.

Availability: KNGP is available as an online Web application at:
http://spark.rstudio.com/kimmelcp/KNGPApp
and as a downloadable R software package at:
https://github.com/kimmelcp/KNGP_Algorithm_Repository

Keywords: Gene prioritization; Bioinformatics

1. Introduction

An ongoing challenge is to integrate existing biologic knowledge to identify and prioritize genes that are likely to be associated with a disease or phenotype. Such prioritization of genes is helpful in focusing the attention of the researcher on a few likely candidates. For instance, in high-throughput techniques such as gene expression profiling, the analyses of differential gene expression may result in hundreds of candidate genes that are associated with the disease of interest. Candidate gene prioritization methods can reduce the large number of candidate genes to a manageable few high priority candidates.

*Corresponding e-mail: kimmelcp@gmail.com
1 Department of Biomedical Informatics, University of Pittsburgh

A common approach followed by candidate gene prioritization algorithms is to use the genes already known to be associated with the disease of interest to rank the other genes. Two categories of gene prioritization algorithms have been described in the literature: similarity-based and network-based algorithms. Similarity-based methods identify those candidate genes whose features are most similar to genes that are already known to be associated with the disease of interest (Radivojac et al. 2008) (J. Chen et al. 2007). Network-based algorithms use a network with nodes and links, where the nodes represent genes, and the links represent a wide variety of different types of interactions between genes such as interactions between corresponding gene products (J. Y. Chen et al. 2006) (Kohler et al. 2008). A small number of gene prioritization algorithms are available as online tools (Aerts et al. 2006) (J. Chen et al. 2009) (Bornigen et al. 2012).

We developed and evaluated a novel network-based gene prioritization method called the Knowledge Network Gene Prioritization (KNGP) algorithm (Kimmel and Visweswaran 2013). The novelty of KNGP is that it can incorporate biological knowledge both as node weights (e.g., knowledge specific to single genes) and as link weights (e.g., knowledge related to pairs of genes) in the network. An example of a node weight is the number of Gene Ontology (GO) terms associated with a gene, and an example of a link weight is the GO similarity between a pair of genes. Previously described network-based algorithms can incorporate knowledge only as link weights that restrict their ability to integrate knowledge such as number of gene ontology terms associated with a gene.

We implemented an online Web application of the KNGP algorithm, which is available at http://spark.rstudio.com/kimmelcp/KNGPApp/. It takes as inputs comma-separated values (csv) files that can be uploaded and outputs a ranked list of genes that can be downloaded as a csv file. This implementation handles relatively small input files but is user friendly and is simple to use. In addition, we provide a software package that is implemented in R that can be downloaded from https://github.com/kimmelcp/KNGP_Algorithm_Repository. This package can handle very large input files and is limited only by the memory resources of the user's hardware.

2. Methods

The KNGP algorithm creates a network from biological knowledge related to genes. The biological knowledge can be provided as 1) node knowledge related to a gene which is represented as a node weight associated with a node in the network (e.g., number of GO terms associated with a gene), and 2) link knowledge related to a pair of genes which is represented as a link weight associated with a link (e.g., GO similarity between a pair of genes). Given a list of genes that are known to be associated with a disease of interest, the algorithm outputs a ranked list of genes that are prioritized according to their relevance to the disease of interest. The KNGP algorithm takes four input files that we describe below (see Fig. 1 for a schematic overview).

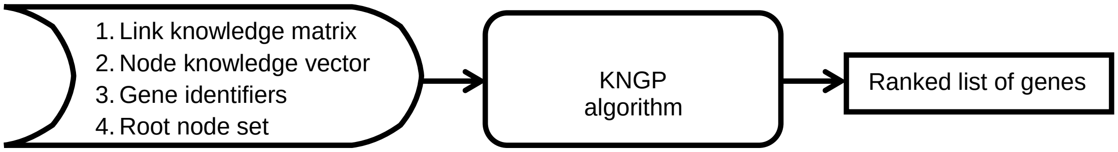

Fig. 1. Schematic overview of KNGP algorithm

1) A file containing a **link knowledge matrix**. This is a *n* **n* matrix where *n* is the number of genes, and an entry in the matrix represents the link weight for a pair of genes. An example of a link weight is the GO similarity between a pair of genes, with a value near 1 indicating a greater degree of similarity and a value near 0 a lesser degree of similarity between the genes. Other examples include protein-protein interactions and the co-expression similarity between genes.

2) A file containing a **node knowledge vector**. This is an *n* dimensional vector where *n* is the number of genes, and an entry represents the node weight for a gene. An example of a node weight is the number of GO terms or PubMed articles associated with a gene.

3) A file containing **gene identifiers**. This is a list of *n* genes that corresponds to the node knowledge vector and contains a gene identifier for each node. For example, the entries may be the Human Gene Nomenclature Committee (HGNC) identifiers or UniProt protein identifiers.

4) A file containing a **root node set**. This is a list of *r* genes specified using the gene identifiers described in 3) that are known to be associated with the disease of interest; this list is a subset of the gene identifiers list ($r << n$). For example, the root node set may consist of those genes already known to be associated with Alzheimer's disease.

To rank genes, KNGP uses a novel extension of the PageRank algorithm that was developed to identify important web pages in the World Wide Web network in response to a user's query. The premise underlying the algorithm is that genes in the network that are in close proximity to genes in the root set are more likely to be associated with the disease than those that are further away. More specifically, a prior probability vector is created from the node knowledge vector that assigns a prior probability for each node. A transitional probability matrix is created from the link knowledge matrix and KNGP performs a random walk where the probability of jumping from one node to a neighboring node is proportional to the weight of the link that connects the two nodes. The number of visits to each node is then used to update the prior probability vector to produce a posterior probability vector. The computed posterior probability vector depends both on the prior probability (that encodes node knowledge) and the transitional probability matrix (that encodes link knowledge). The posterior probability associated with a node is interpreted as the importance of that node relative to other nodes in the network, and the output consists of genes that are ranked based on the posterior probability.

The online application provides examples of the four input files. The files are to be uploaded as csv files without headers. The displayed output consists of the top 50 ranked genes according to the KNGP algorithm, and the complete ranked list of genes can be downloaded as a csv file.

3. Discussion

The KNGP algorithm was evaluated on synthetic data and gene networks based on biological knowledge. In the gene networks, link knowledge was encoded as protein-protein interactions, and node knowledge was derived from the three GO ontologies (GO molecular function ontology, GO biological process ontology, and GO cellular component ontology). When evaluated on 19 diseases for which we derived root node sets, KNGP using both link and node knowledge performed better that using only link knowledge or using only node knowledge. For example, when applied to asthma, the top 5 ranked genes contained two genes that were ranked far lower when using only link knowledge or only node knowledge. For both of these genes, we obtained evidence from the literature that they are associated with asthma.

The KNGP algorithm is a novel gene prioritization algorithm that can encode both link and node knowledge, and we hope that the publically available web implementation will help other researchers utilize the algorithm in their own research.

In our implementation, we used a desktop PC with 30 GB of RAM and 8 cores for our experiments. We suggest using a parallelized version of R that can take advantage of multi-threaded processing. A good R binary that automatically uses parallel routines is the Revolution Analytics binary (http://www.revolutionanalytics.com/download-r).

4. Conclusion

We have created an online application which implements the KNGP algorithm. The user can easily upload their own data files and obtain a ranked list of genes from in the context of a set of genes that are known to be associated with the disease of interest. Furthermore, we have developed a R software package that can handle large datasets. We hope that these resources will be useful to researchers for candidate gene prioritization.

Funding: This research was funded by NLM grant T15 LM007059 to the University of Pittsburgh Biomedical Informatics Training Program.

Conflict of Interest: None declared.

References

Aerts, S., et al. (2006), 'Gene prioritization through genomic data fusion', Nat Biotechnol, 24 (5), 537-44. http://dx.doi.org/10.1038/nbt1203

Bornigen, D., et al. (2012), 'An unbiased evaluation of gene prioritization tools', Bioinformatics, 28 (23), 3081-8. http://dx.doi.org/10.1093/bioinformatics/bts581

Chen, J., et al. (2007), 'Improved human disease candidate gene prioritization using mouse phenotype', BMC Bioinformatics, 8, 392. http://dx.doi.org/10.1186/1471-2105-8-392

Chen, J., et al. (2009), 'ToppGene Suite for gene list enrichment analysis and candidate gene prioritization', Nucleic Acids Res, 37 (Web Server issue), W305-11.

Chen, J. Y., Shen, C., and Sivachenko, A. Y. (2006), 'Mining Alzheimer disease relevant proteins from integrated protein interactome data', Pac Symp Biocomput, 367-78. http://dx.doi.org/10.1142/9789812701626_0034

Kimmel, C. and Visweswaran, S. (2013), 'An algorithm for network-based gene prioritization that encodes knowledge both in nodes and in links', PLoS One, 8 (11), e79564. http://dx.doi.org/10.1371/journal.pone.0079564

Kohler, S., et al. (2008), 'Walking the interactome for prioritization of candidate disease genes', Am J Hum Genet, 82 (4), 949-58. http://dx.doi.org/10.1016/j.ajhg.2008.02.013

Radivojac, P., et al. (2008), 'An integrated approach to inferring gene-disease associations in humans', Proteins, 72 (3), 1030-7. http://dx.doi.org/10.1002/prot.21989