Research Article

Gene Ontology Mining: A Survey

Lakshmi K. S*[1], G. Vadivu[2]
[1]Assistant Professor, Department of Information Technology, Rajagiri School of Engineering & Technology, Kochi, Kerala, India.
[2]Professor and Head, Department of Information Technology, SRM University, SRM Nagar, Kattankulathur, Tamil Nadu, India.
***Corresponding author's E-mail:** lekshmy.shalu@gmail.com

Received: 28-12-2016; **Revised:** 19-02-2017; **Accepted:** 11-03-2017.

ABSTRACT

Gene ontology is a key bioinformatics initiative to provide a common language to express varying facets of a gene product's biology. Data mining provides various methods for extracting useful information from Gene ontologies. Among them, association rule mining is a most promising method which can be used for extracting novel associations between terms in Gene ontology. This paper presents a survey on various algorithms used for association rule mining in gene ontology. Apriori, FP Growth and other association rule mining algorithms used in single-level as well as cross ontology mining are discussed and compared in this study.

Keywords: Gene ontology, Data mining, Association Rule Mining.

INTRODUCTION

Biological and genetic information are rich knowledge sources. For the effective utilization of this information, computer technology can be made use of. Bioinformatics deals with the application of computer technology for the manipulation of biological and genetic information which can then be applied to gene-based drug discovery and other purposes.

With the advancement of technology, lot of biological and genetic data, are generated day by day.

Discovering novel information out of these data requires refined computational scrutiny.

The significance of this latest field of inquest will grow as more and more genomic, proteomic, and other data are generated throughout.

The overall goal of bioinformatics is to permit the discovery of new biological insights.

The foundation of the Gene Ontology (GO) Consortium was a vital step toward the adoption of formal and objective knowledge representations in biological sciences[1].

As our knowledge of biological phenomena and our ability to represent that knowledge are continuously growing, the ontology is undergoing constant development.

Gene Ontology contains concepts called GO Terms. The process by which a GO term is associated with another biological term is known as annotation.

GO is presently the real standard for functional annotation of gene products. The whole corpus of annotations is stored into publicly available databases, such as the Gene Ontology Annotation (GOA) database[2].

GO is subdivided in three non-overlapping ontologies: Cellular Component (CC), Molecular Function (MF) and Biological Process (BP). Each of this ontology describes a particular aspect of a biological term. Terms and relationships are stored in GO in the form of Directed Acyclic Graph (DAG). In the Directed Acyclic Graph structure of GO, the nodes are terms and the relations among terms are the edges. The GO annotations of gene ontology represents every GO term associated with one of the three ontology term, cellular component, molecular function, or biological process. There are about 14 different methods for gene ontology annotation which are generally grouped into two: IEA (Inferred from Electronic Annotation) and Manual Annotation. IEA annotations are done by computational techniques. Each annotation is labelled with an evidence code (EC). EC is useful to keep track of the method of annotation. Classical approaches for analyzing annotation data are functional enrichment analysis (i.e. the determination of over or under representation of an annotation in a set of annotated data) or semantic similarity (i.e. the determination of relatedness of two or more annotating terms)[4-6].

Data mining has emerged as a promising field in solving biological problems. Various data mining techniques like association rule mining, classification and clustering play a significant role in genomic and proteomic researches including cancer prediction, protein structure prediction etc.

Frequent pattern mining on annotated data is an emerging field of research. In this paper, we will have a comparative study on the different association rule mining techniques used on annotated data for the prediction of novel annotations.

Association Rule Mining

Association rule mining (ARM)[7-8] is one of the most significant methods of data mining. It has attracted the

interest of researchers, and is one of the scientific potentials of data mining that allows discovered correlations and association between capacious datasets. The relevance of ARM is increasing with the increasing demand of finding frequent patterns from large data sources. ARM involves two important steps: i.e. 1) Extraction of frequent items 2) Extracting association rules from these frequent items. The complexity of extraction process is determined in terms of response time and memory space. The number of frequent items is exponential to the number of items in database.

Let T be a database, where T = $\{t_1,t_2,\ldots\ldots,t_m\}$ is a set of transactions over I = $\{i_1, i_2,\ldots,i_n\}$ which is a set of items. A non-empty subset of I is called an item set. Each transaction t_i in T is defined as an itemset $i_1, i_2,\ldots,i_k$ of length k. An Association Rule is an implication between two item sets X and Y, in the form of X → Y where X∩Y= Φ. X is called the antecedent and Y is the conclusion, or the consequent, of the association rule. Each association rule X → Y may be characterized by two measures Support and confidence. They are used for selecting ARs according to their potential significance to the user:

Support (X): Support of an item X is the number of times, X occurs in transactions in a database.

Confidence: Confidence of a rule X→Y is an indication of how often the items X and Y occur together in the transaction. It is defined mathematically as Confidence (X→Y) = Support (X∩Y) / Support(X).

Association Rule Mining can be broadly classified into single level and multilevel ARM. If mining is done on single level of abstraction, then it is known as single level ARM and if mining is done on more than one layer of data then it is known as multilevel ARM. Many algorithms have been designed to generate frequent patterns in ARM. The most popular ARM algorithms are Apriori and FP-Growth algorithm. Both these algorithms are scalable and efficient. Apriori algorithm mines the frequent item set by generating the candidate data set. But this takes lot of time and space. To overcome this drawback FP growth algorithm is introduced which mines the frequent items without generating candidate data set. But the obstacle is it generates a massive number of conditional FP trees. Apart from Apriori and FP-Growth algorithm, other important algorithms used in ARM are Apriori TID, Apriori Hybrid, Eclat, Partition and MaxMiner.

Problems in Gene Ontology Annotations

Two main issues are related to gene ontology annotations: (i) Annotation Count (ii) Type of Annotations.

Annotation Count

Count of annotation for each protein or gene may vary substantially within the same GO taxonomy and over different species due to the different methods used for annotation, and to the different availability of experimental data.

Type of Annotations

There are two types of annotations: Manuals annotations and IEA annotations. IEA annotations are more general compared to manual annotation. Manual annotations are more precise. But the number of annotations obtained using manual annotation is less than IEA annotations.

Algorithms used in Single Level Go Mining

Apriori Algorithm

Apriori[9] algorithm is the key algorithm for the extraction of association rules because it constituted the basis of the majority algorithms that are designed to extract the association rules. It is a research iterative algorithm of frequent itemsets by level, based on the anti-monotonicity property: i.e. subset of a frequent itemset will also be frequent.

Apriori algorithm consists of two steps. In the first step, candidate itemsets are generated and in the next step infrequent itemsets are pruned out.

Procedure for Apriori algorithm

C_k: Candidate itemset of size k
F_k: Frequent itemset of size k
F_1 = Frequent itemset;
For (k=1; F_k ! = null; k++)
Do Begin
C_{k+1} = candidates generated from F_k;
For each transaction t in database
Increment the count of all candidates in C_{k+1} that are contained in t
F_{k+1} = candidates in C_{k+1} with min_support
End
Return $\upsilon_k F_k$

Fp-Growth Algorithm

J. Han proposed FP-Growth[9] Algorithm, aims at overcoming the disadvantages of Apriori for ARM. FP-growth is an efficient and scalable method for extracting the complete set of frequent patterns from a transaction database. It uses the technique of pattern fragment growth. FP-growth uses an extended prefix-tree structure (frequent pattern tree otherwise known as FP tree) for storing compressed and crucial information about frequent patterns. FP growth uses divide and conquer strategy. It needs 2 scans on the dataset. During the first scan, it computes a listing of frequent items sorted by frequency in dropping order (F-List). Within the second scan, the database is compressed into a FP-tree. This algorithmic rule performs mining on FP-tree recursively.

Phases of FP-Growth algorithm:

Phase 1: Construction of FP-Tree.

Phase 2: Mining frequent itemsets from the FP-Tree.

Phase1: Construction of FP-TREE

FP-tree construction phase of FP-growth algorithm is divided into 2 passes. In the first pass, entire database is scanned and the support of each item is found. Infrequent items based on support count will be discarded. Then the frequent items are sorted based on

their support count in decreasing order. During the second pass, FP-tree is constructed with nodes corresponding to items. A counter is maintained for each node. FP-Growth reads one transaction at a time and maps it to a path. Paths are traversed in fixed order, same as the order of items in transactions. Paths may overlap when more than one transaction share the same item. In such cases counters will be incremented.

Table 1: Transaction Database

Tid	(Ascended) Frequent Items
1	p,m,a,c,f
2	m,b,a,c,f
3	b,f
4	p,b,c
5	p,m,a,c,f

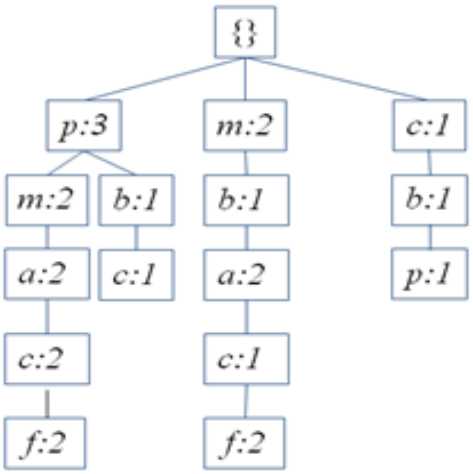

Figure 1: FP-Tree corresponding to Table 1

Phase 2: Mining frequent itemsets from the FP-Tree

In phase 2, conditional pattern base are generated from the FP-tree. Support counts are assigned to pattern base which is the minimum of the support counts for the items in the pattern base. For each frequent item, conditional patterns are generated. Conditional FP trees are generated from these conditional patterns. The process is repeated for each newly created conditional FP-tree and continues until the resulting FP-tree is empty or it contains only one path.

Eclat Algorithm

Eclat[9] algorithm was proposed by Zaki. Unlike Aprioi algorithm which uses a breadth first search on transactions, Eclat uses a depth first traversal on the transaction database. Let D be a transaction database with transactions {T1, T2, T3, T4, T5, T6, T7, T8, T9} and I is an itemset with items, {I_1, I_2, I_3, I_4, I_5}. Table 2 shows the transaction database. In order to apply Eclat algorithm on D, first D is converted to vertical format as shown in Table.3.

Support count can be obtained for each item in I from the vertical format of the transaction database. From the vertical database, frequent itemsets are generated repeatedly until no candidate itemsets can be found.

Table 2: Transaction Database

TID	List of Items
T1	I_3,I_5
T2	I_3,I_4
T3	I_3,I_5
T4	I_1,I_4
T5	I_1, I_3
T6	I_3,I_2
T7	I_1, I_2
T8	I_1, I_2,I_5
T9	I_1, I_2

Table 3: Transaction database in vertical format

Item set	TID_set
I_1	{T4,T5,T7,T8,T9}
I_2	{T6,T7,T8,T9}
I_3	{T1,T2,T3,T5,T6}
I_4	{T2,T4}
I_5	{T1,T3,T8}

Algorithm Used in Cross Ontology Mining

Classical AR algorithms are not able to deal with different sources of production of GO annotations. As a result, classical algorithms generate candidate rules with low Information Content (IC).

There are several algorithms available for mining gene ontology at sub-ontology levels which is referred as cross ontology mining[10-11].

G. Agapito[4] presented GO-WAR that can generate candidate rules with a high level of IC. In GO-WAR algorithm, Support and Confidence are not lost during the rule discovery phase. GO-WAR does not require post-

processing strategies for eliminating uninteresting rules. Publicly available GO annotation data is used for explaining GO-WAR mining algorithm[5].

Procedure for GO-WAR Algorithm

Require: TDB, wminSupp, Conf
Ensure: mining of weighted association rules
1. Data Structure initialization: TDB, FP-Tree, β_{Tree}
2. For all x ∈ TDB do
3. If wS(x) ≥ wminSupp then
4. frequentItemsList ← x
5. End if
6. End for
7. descendingSorting(frequentItemsList,TDB)
8. FP_Tree.Create ← frequentItemsList
9. While (node ≠ root) do
10. mineRules(Conf)
11. saveRules()
12. end while

Comparison of Different Algorithms

The performance of the algorithms is compared by varying the support percentage. Fig 2 shows the comparison of Apriori, Eclat and FP-Growth algorithms on single level data and Fig 3 shows the performance of GO-WAR algorithm in comparison with Apriori and FP-Growth algorithms. From the figure, it is clear that FP-growth algorithm performs better than Eclat and Apriori. Execution time needed for Apriori is greater than that of Eclat and FP-Growth. FP-growth algorithm requires at most two scans of the database. In Apriori, as the size of dataset increases, the number of database scan needed also increases.

Performance of Apriori algorithm decreases with increase in support percentage. Performance of FP-growth remains unaffected by the variation of support factor. Apriori needs more database scan. Eclat needs only one database scan and it finds next level itemsets by intersecting current level itemsets. FP_Growth uses complex data structure compared to Aprori and Eclat. In cross-ontology mining GO-WAR outperforms Apriori and FP-Growth in execution time and accuracy.

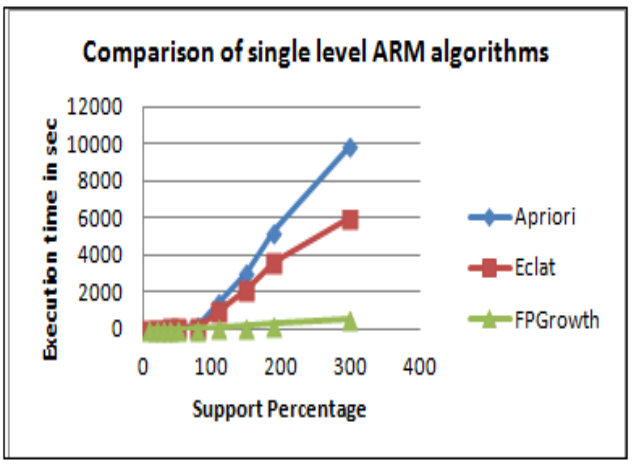

Figure 2: Comparison of single level ARM algorithms

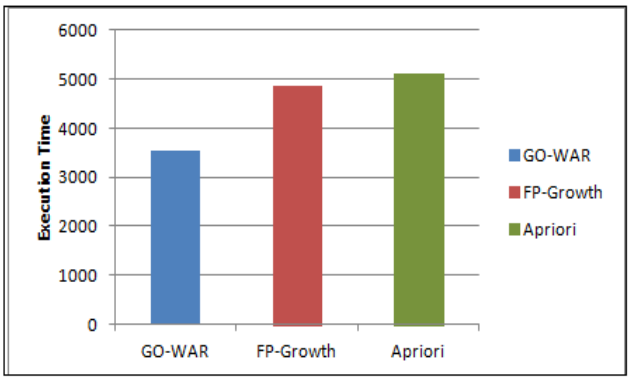

Figure 3: Comparison of GO-WAR algorithm with Apriori and FP-Growth

CONCLUSION

The performances of various algorithms have been compared. GO-WAR is an efficient algorithm for cross-ontology mining. FP-Growth algorithm performs better for single level data mining. Gene Ontology mining at cross ontology levels will help in discovering novel associations between gene ontology terms. This can be useful for drug discoveries, finding genomic and proteomic relationships and other significant bioinformatics applications.

REFERENCES

1. Ashburner M, Ball CA, Blake JA, Botstein D, Butler H, Cherry JM, Davis AP, Dolinski K, Dwight SS, Eppig JT, Harris MA, Hill DP, Issel-Tarver L, Kasarskis A, Lewis S, Matese JC, Richardson JE, Ringwald M, Rubin GM, Sherlock G. Gene ontology: tool for the unification of biology. The Gene Ontology Consortium, Nat Genet. 25(1), 2000 May, 25-9, Doi: 10.1038/75556.
2. Camon E, Magrane M, Barrell D, Lee V, Dimmer E, Maslen J, Binns D, Harte N, Lopez R, Apweiler R. The gene ontology annotation (GOA) database: sharing knowledge in UNIPROT with gene ontology, Nucleic Acids Res. 32 (Suppl.1) (2004) D262–D266, Doi: 10.1093/nar/gkh021.
3. Vadivu G and Hopper SW. Semantic Linking and Querying of Natural Food, Chemicals and Diseases, International Journal of Computer Applications, (0975 – 8887) Volume 11 – No.4, Dec 2010, Doi: 10.5120/1567-2093.
4. Agapito G, Cannataro M, Guzzi PH, Milano M. Using GO-WAR for mining cross-ontology weighted association rules, Journal of Computer methods and programs in biomedicine, Volume 120, Issue 2, July 2015, Pages 113-122, Doi:10.1016/j.cmpb.2015.03.007.
5. Agapito G, Milano M, Guzzi PH, Cannataro M. Extracting Cross-Ontology Weighted Association Rules from Gene Ontology Annotations, IEEE/ACM Transactions On Computational Biology and bioinformatics, Vol. 13, No. 2, March/April 2016, Doi:10.1109/TCBB.2015.2462348.
6. Guzzi PH, Milano M, and Cannataro C. Mining Association Rules from Gene Ontology and Protein Networks: Promises and Challenges, Procedia Computer Science, Volume 29, 2014, Pages 1970–1980, Doi: 10.1016/j.procs.2014.05.181.
7. Kaur M, Kang S. Market Basket Analysis: Identify the changing trends of market data using association rule

mining, Procedia Computer Science, 85: 78 – 85, December 2016, Doi: 10.1016/j.procs.2016.05.180.

8. Kumbhare TA, Chobe SV. An Overview of Association Rule Mining Algorithms, International Journal of Computer Science and Information Technologies, Vol. 5 (1), 2014, 927-930, ISSN: 0975-9646.
9. Nasreen S, Azam MA, Shehzad K, Naeem U, Ghazanf MA. Frequent Pattern Mining Algorithms for Finding Associated Frequent Patterns for Data Streams: A Survey, Procedia Computer Science, 37:109–116, September 2014, Doi: 10.1016/j.procs.2014.08.019.
10. Manda P, Ozkan S, Wang H, McCarthy F, Bridges SM. Cross-Ontology Multi-level Association Rule Mining in the Gene Ontology, PLOS One, October 2012, Vol 7, Issue 10, Doi: 10.1371/journal.pone.0047411.
11. Jayasri D, Manimegalai D. An Efficient Cross Ontology-Based Similarity Measure for Bio-Document Retrieval System, Journal of Theoretical and Applied Information Technology 20th August 2013. Vol. 54 No.2.
12. Idoudia R, Ettabaa KS, Solaiman B, Hamrouni K. Ontology Knowledge mining based Association Rules Ranking, Procedia Computer Science, 96, December 2016, 345 – 354, Doi: 10.1016/j.procs.2016.08.147.
13. Faria D, Schlicker A, Pesquita C, Bastos H, Anto´ nio E. N. Ferreira, Albrecht M, Andre´ O. Falcao. Mining GO Annotations for Improving Annotation Consistency, PLoS One. July 2012, Vol 7, Issue7, Doi: 10.1371/journal.pone.0040519.
14. Zhu P, Jia F. A New Ontology Based Association Rules Mining Algorithm, Journal of Theoretical and Applied Information Technology, Vol. 45, No.1, 15th November 2012, ISSN: 1992-8645.
15. Naulaerts S, Meysman P, Bittremieux W, Vu TN, Vanden Berghe W, Goethals B, Laukens K. A primer to frequent itemset mining for bioinformatics, Briefings in Bioinformatics, 16(2), 2015 Mar, 216-31, Doi: 10.1093/bib/bbt074, Epub: 2013 Oct 26.
16. Shivakumar BL, Porkodi R. Finding relationships among gene ontology terms in biological documents using Association Rule mining and GO annotations, International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: Vol. 2, No.3, June 2012, 2249-955.
17. Wang Y and Chen Y. A New Association Rules Mining Method based on Ontology Theory, IEEE fifth International Conference on Advanced Computational Intelligence(ICACI) October 18-20,2012 Proceedings, Doi: 10.1109/ICACI.2012.6463170.
18. Girotra M, Nagpal K, Minocha S, Sharma N. Comparative Survey on Association Rule Mining Algorithms, International Journal of Computer Applications (0975 – 8887) Volume 84 – No 10, December 2013.
19. Agapito G, Milano M, Guzzi PH and Cannataro M. Improving Annotation Quality in Gene Ontology by Mining Cross-Ontology Weighted Association Rules, Proceedings of 2014 IEEE International Conference on Bioinformatics and Biomedicine, Doi: 10.1109/BIBM.2014.6999374.
20. Manda P, McCarthy F, Bridges SM. Interestingness measures and strategies for mining multi-ontology multi-level association rules from gene ontology annotations for the discovery of new GO relationships, Journal of Biomedical Informatics 46, 2013, 849–856, http://dx.doi.org/10.1016/j.jbi.2013.06.012.
21. Benites F, Simon S, Sapozhnikova E. Mining Rare Associations between Biological Ontologies, PLoS One. Volume 9, Issue 1, January 2014. http://dx.doi.org/10.1371/journal.pone.0084475.
22. Nagar A, Hahsler M, Al-Mubaid H. Association Rule Mining of Gene Ontology Annotation Terms for SGD, Conference Proceedings of IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, 12-15 August 2015, Doi: 10.1109/CIBCB.2015.7300289.
23. Vadivu G and Hopper SW. Ontology Mapping of Indian Medicinal Plants with Standardized Medical Terms, Journal of Computer Science, ISSN 1549-3636, Aug, 2012, Doi: 10.3844/jcssp.2012.1576.1584.
24. Vadivu G, Swaminathan R, Thenmozhi M. Similarity Measure Based On Edge Counting Using Ontology, International Journal of Engineering Research and Development, ISSN (Online):2278-067X, Aug, 2012, ISSN(Print): 2278-800X.
25. Vadivu G, Hopper SW, BharatRam G. Semantic Data Integration and Querying using SWRL, LNCS, Springer-verlag ISSN - 1865-0929, ISBN - 978-3-642-22542-0, Jul, 2011, Page No. 567-574, Doi: 10.1007/978-3-642-22543-7_58.

Source of Support: Nil, **Conflict of Interest:** None.