ISSN:0975-9646

Performance Evaluation of Gene based Ontology Using Attribute Selection Methods

Ch. Uma Shankari[#1], T. Sudha Rani[*2]

[#]*M. Tech. Student, Department of Computer Science and Engineering, Aditya Engineering College,*
Surampalem, Peddapuram, East Godavari Dist., Andhra Pradesh, India

[*]*Sr. Assistant Professor, Department of Computer Science and Engineering, Aditya Engineering College,*
Surampalem, Peddapuram, East Godavari Dist., Andhra Pradesh, India

Abstract— **Senescence is the gradual degeneration of function characteristic of most complex life forms. Bioinformatics is a multidisciplinary research that combines biology, computer science, mathematics and statistics into a general field that will have erudite percussion on all fields of life sciences. In this paper, considering the senescence data gathered from four model organisms which are lumbricina (worm), saccharomyces (yeast), diptera (fly), and mus (mouse). By applying the classification methods like Naïve Bayes, 1-Nearest Neighbour, it gives less efficiency. To increase the efficiency, new attribute selection methods are proposed which uses the Support Vector Machine (SVM) algorithm to organize the senescence-related data. The intention of these hierarchical attribute selection methods which uses the SVM algorithm, to organize organism genes into either anti-longevity or pro-longevity genes that gives the better results than with the use of Naïve Bayes algorithm.**

Keywords— **Senescence, classification, attribute selection, naive bayes, 1-nearest neighbour, support vector machine**

I. Introduction

Senescence can be defined as the process by which organism's proceed through a physical deterioration of the body. The data mining methods are applied on senescence data, but it has taken a large enough of time to discover the discrete patterns. Classification is the data mining task that assigns objects to target categories or classes. When the numbers of instances are increased, attribute selection methods are applied to the input data before the classification task is performed. Attribute selection methods are used to select the most compatible and non-repetitious attributes from the input data. So, it gives better efficiency to the classification algorithm.

In this paper, the classification task is performed on four major organisms based on the attribute values indicating whether the gene belongs to with Gene Ontology (GO) term or not, where the term refers to biological process. Pro-longevity genes extend the lifespan of an individual that promotes better use of preventive medicine. Anti-longevity genes are those which function to shorten the lifespan in wild type organisms [2].

GO terms are classified into a hierarchical structure in which the terms are arranged according to their importance, where the ancestors of each GO term are considered as a general terms and its descendants as a specialized terms.

This paper introduces two new attribute selection methods by taking GO terms as attributes and repetitious among GO terms, reduce the repetitious in the selected GO terms and obtains the higher predictive efficiency. These attribute selection methods works well with the Naıve Bayes and 1-NN (nearest neighbour) classifiers in the context of "lazy learning", where a set of instances is submitted for classification for identifying best attributes.

II. Naive Bayes

To organize the model organism's genes into anyone of the category, the Naïve Bayes algorithm is used [4]. Naïve Bayes classifier is based on Bayes theorem with independence assumptions between predictors, but it is not work well if the data is non separable and also it gives less efficiency when organizing the genes. Due to these drawbacks, we move on to another algorithm.

III. Attribute Selection Methods with SVM

In this paper, proposed three types of attribute selection methods which are Hierarchical Information-Preserving (HIP) GO terms, Most Relevant (MR) GO Terms and Hierarchical Information-Preserving and Most Relevant (HIP-MR) GO Terms. All these three methods are proposed in our recent paper [3]. In this paper, the proposed methods use the classification algorithm SVM to organize the genes. The advantage of using a SVM algorithm is, it has a technique called the kernel trick that can work well even if the data is not separable and it gives more efficiency. It has various options like "linear", "rbf" (default value), "poly" etc.. Among these "rbf" and "poly" are useful for non-linear hyper-plane. With this non-linear hyper-plane, the genes are classified either any one of the class that is to be predicted. By the use of SVM, these selection methods are used to gain the two goals: reduce the repetitious between GO terms and select the instances which are having higher relevance for class prediction. These methods follow lazy learning approach i.e. select the attributes for each testing instance.

Consider the below Fig. 1, contains the genes used as attributes. Each attribute associate with the value '1' or '0' to indicate whether the corresponding gene of that instance is associated with the corresponding GO term or not.

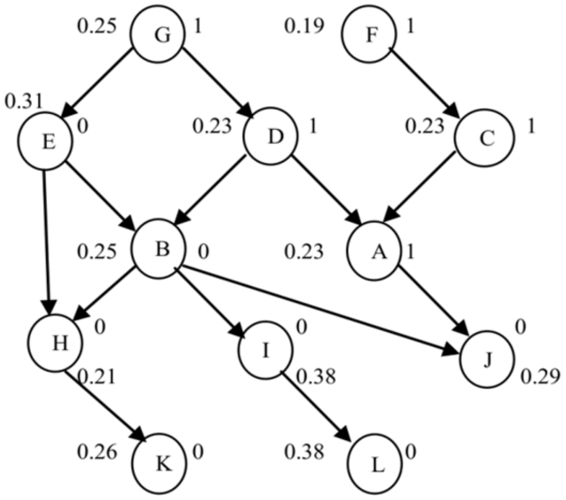


Fig 1. Example of DAG

The Attribute Selection methods are described below.

A. Hierarchical Information Preserving (HIP) GO Terms

By this method, there is a compression of repetitious among the selected GO terms.

The HIP method algorithm is shown below,

ALGORITHM1. HIERARCHICAL INFORMATION PROCESSING (HIP) GO TERMS

```
Construct DAG by taking all GO terms in
  DS;
Load $DS_{\langle Train\rangle}$ ;
Load $DS_{\langle Test\rangle}$ ;
for each $GO_j$ term in DAG do
    Load $A(GO_j)$ from DAG;
    Load $D(GO_j)$ from DAG;
    Load $S(GO_j)\leftarrow$"Selected" ;
end for
for each $I_{\langle m\rangle}\in DS_{\langle Test\rangle}$ do
    for each $GO_j\in DAG$ do
        if $V(GO_{j,m})=1$ then
        for each $Anc_{jk}\in A(GO_j)$ do
        $S(Anc_{jk})\leftarrow$"Deleted";
        end for
        else
        for each $Dec_{jk}\in D(GO_j)$ do
        $S(Dec_{jk})\leftarrow$"Deleted";
        end for
        end if
    end for
    $I_{\langle s\rangle}\leftarrow\{GO_j:S(GO_{j,m})=$"Selected"$\}$;
    $SVM(DS_{\langle Train\rangle}, I_{\langle s\rangle})$;
    Re-assign $\forall GO_j:S(GO_j)\leftarrow$"Selected";
end for
```

Where $DS_{\langle Train\rangle}$ and $DS_{\langle Test\rangle}$ designate the train dataset and test dataset. $A(GO_j)$ and $D(GO_j)$ represent the set of general terms and specialized terms of the j^{th} GO term. $S(GO_j)$ designate the selection status which is either "Selected" or "Removed" of the GO_j term. $I_{\langle m\rangle}$ designate the current instance that belongs to $DS_{\langle Test\rangle}$ to organize. $V(GO_{j,m})$ represents the value of GO_j term (1/0). Anc_{jk} designate the k^{th} general term of the GO_j and Dec_{jk} represents the k^{th} specialized term of GO_j . $I_{<s>}$ designate the dataset contains the selected GO terms.

The above algorithm shows the loading the DAG by taking all GO terms, finding all general terms and specialized terms of each GO term, assigns the status as "Selected" for each GO term and performs the attribute selection by using lazy learning approach. Consider the GO term GO_j , the algorithm checks the value of GO_j term with the ancestors and descendants values. If GO_j term value is "1", it compares with all ancestor values. In those, which ancestors having the value "1" consider them as repetitious and set the status to be "Deleted" and which are having the value other than 1 (ex: 0) consider them as compatible attributes. If GO_j term value is "0", it compares that GO_j term value with all of its descendant values. The descendants which are having the value "0" consider them as repetitious and set the status to be "Deleted" and which are having the value other than 0 (ex: 1) consider them as compatible attributes.

For the understanding of an algorithm, explained with an example.

From the above fig 1, When the term A is processed, the terms G,D,F,C selection status will be "Deleted" which is assigned by the HIP algorithm because they had the values similar to A(1). When E is processed, the terms B, H, I, J, K, and L selection status will be "Deleted" because they had the values similar to E (0).

Processing of all the terms for the GO term t is completed in the above example DAG, the terms A, E are selected. The current testing instance is minimized only having the attributes whose status is "Selected" and performs the SVM (Support Vector Machine) on the minimized instance. Finally, for all the GO term attributes set the status as "Selected", for the formation of further testing instance.

B. Most Relevant (MR) GO Terms

This method selects the attributes based on the relevance value of each GO term and also the repetitious among the GO terms. In MR method, if the GO term t has a value (1), consider the general terms that had the path from the root node to the term t. If the GO term t has a value (0), consider the specialized terms that had the path from the GO term t to the leaf node. MR compares the relevance value of t with relevance value of each term in the identified paths for GO term t and removes all the terms except the term which is having the most compatible value. In the path if there exists more than one GO terms having the same maximum relevance values, then take the inmost term among the set of terms for the value “1” or take the generic term among the set of terms for the value “0”.

The MR method algorithm is shown below,

ALGORITHM2. MOST RELEVANT (MR) GO TERMS

```
Load DAG by taking all GO terms in DS;
Load $DS_{\langle Train \rangle}$;
Load $DS_{\langle Test \rangle}$;
for each $GO_j$ in DAG do
    Load $A_+(GO_{j,l})$ from DAG;
    Load $D_+(GO_{j,l})$ from DAG;
    Load $S(GO_j) \leftarrow "Selected"$;
    Calculate $R(GO_j)$ in $DS_{\langle Train \rangle}$;
end for
for each $I_{\langle m \rangle} \in DS_{\langle Test \rangle}$ do
    for each $GO_j \in DAG$ do
    if $V(GO_{j,m})=1$ then
    for each $Path_l$ from $GO_j$ to root in DAG do
        Discover MRT in $A_+(GO_{j,l})$;
        for each $Anc_{j,k,l+}$ except
        MRT do
        $S(Anc_{j,k,l+}) \leftarrow "Removed"$;
        end for
    end for
    else
    for each $Path_l$ from $GO_j$ to leaf in DAG do
        Discover MRT in $D_+(GO_{j,l})$;
        for each $Dec_{j,k,l+}$ except MRT do
        $S(Dec_{j,k,l+}) \leftarrow "Removed"$;
        end for
    end for
    end if
    end for
    $I_{\langle s \rangle} \leftarrow \{GO_j : S(GO_{j,m}) = "Selected"\}$;
    $SVM\ (DS_{\langle Train \rangle}, I_{\langle s \rangle})$;
    Re-assign $\forall GO_j : S(GO_j) \leftarrow "Selected"$;
end for
```

Where $R(GO_j)$ denotes the relevance value for j^{th} GO term. $A_+(GO_{j,l})$ and $D_+(GO_{j,l})$ designate the set of GO terms containing the j^{th} GO term and also its ancestors or descendants in the l^{th} path. $Anc_{j,k,l+}$ and $Dec_{j,k,l+}$ designate the k^{th} term in $A_+(GO_{j,l})$ and $D_+(GO_{j,l})$ .MRT designates the most compatible term among the set of GO terms in $A_+(GO_{j,l})$ or $D_+(GO_{j,l})$.

From the above algorithm (Algorithm 2), it shows the construction of DAG. For each GO term, the ancestors and descendants at each path will be loaded. Calculates the relevance (R) value of each GO term and performs the attribute selection by using lazy learning approach.

The following algorithm is explained with an example by considering the fig 1. When processing the term A (value=1), the GO term has two paths from the root. First path contain the terms G, D and A, where G is having the high relevance value and it is selected as a MRT. Second path contains the terms F, C, A. Among these, the terms C and A are having the high relevance values but the term A is selected as MRT because A is the deeper than term C in that path. Hence after processing the term A, set the status of all GO terms contained in two paths to “Removed” except the term G.

When the processing of another term E (value=0), the GO term has four paths. First path contains the terms E, H and K and Second path contains the terms E,B,H and K. Third path contains the terms E,B and J. In these three paths, E is having the high relevance value and selected as the MRT. Fourth path contains the terms E,B,I and L. Among these, I and L having the maximum relevance values but I is selected as the MRT because I is shallower than L. Hence after processing the term E, set the status of all GO terms contained in four paths to “Removed” except term I.

Processing of all the terms for GO term t is completed; the terms G and I are selected and the testing instance has reduced which contains the attributes whose status is “Selected”. SVM algorithm is performed on the reduced instance. Finally, for all the GO term attributes set the status as “Selected”, for the formation of further testing instance.

C. Hierarchical Information Preserving and Most Relevant (HIP-MR) GO Terms

This method will select the attributes by considering both repetitious between the selected core terms and their relevance values.

When the GO term t in the instance is processed, HIP-MR method first identifies the set of GO terms based on the value of GO term t. If the value of the term t is “1”, consider all general terms having the value 1. If the value of the term is “0”, all specialized terms having the value 0 has to be considered. HIP-MR removes the general terms of t (value=1) whose relevance value are less than or equal to the relevance value of t and also removes the specialized terms of t (value=0) whose relevance values are less than or equal to the relevance value of t.

The HIP-MR algorithm is shown below,

ALGORITHM3. HIERARCHICAL INFORMATION PROCESSING AND MOST RELEVANT (HIP-MR) GO TERMS

```
Load DAG by taking all GO terms in DS;
Load $DS_{\langle Train \rangle}$;
Load $DS_{\langle Test \rangle}$;
for each $GO_j$ in DAG do
    Load $A(GO_j)$ from DAG;
    Load $D(GO_j)$ from DAG;
    Load $S(GO_j) \leftarrow "Selected"$;
    Calculate $R(GO_j)$ in $DS_{\langle Train \rangle}$;
end for
for each $I_{\langle m \rangle} \in DS_{\langle Test \rangle}$ do
    for each $GO_j \in DAG$ do
    if $V(GO_{j,m}) = 1$ then
        for each $Anc_{jk} \in A(GO_j)$ do
        if $R(Anc_{jk}) \le R(GO_j)$ then
        $S(Anc_{jk}) \leftarrow "Ejected"$;
        end if
        end for
    else
        for each $Dec_{jk} \in D(GO_j)$ do
        if $R(Dec_{jk}) \le R(GO_j)$ then
        $S(Dec_{jk}) \leftarrow "Ejected"$;
        end if
        end for
    end if
    end for
    $I_{\langle s \rangle} \leftarrow \{GO_j : S(GO_{j,m}) = "Selected"\}$;
    $SVM(DS_{\langle Train \rangle}, I_{\langle s \rangle})$;
    Re-assign $\forall GO_j : S(GO_j) \leftarrow "Selected"$;
end for
```

This algorithm shows the construction of DAG, the general terms and specialized terms of each GO term is identified, and calculates the relevance value of each GO term. It selects the attributes by considering the repetitious and relevance values.

The above algorithm is explained in detail by taking the above example fig 1, when processing the GO term A which has the value ‘1’, its relevance value is compared with all ancestor terms C, D, F, G relevance values. The ancestor terms C, D, F are removed because their relevance values are less than or equal to the relevance value of A.

When processing the GO term E (value=0), its relevance value is compared with all descendants terms B, H, I, J, K, L relevance values. The descendant terms B, H, J, K are removed because the relevance values of those terms are less than or equal to the relevance value of E. After processing of GO term t has been completed, the selected GO terms are G and I. Now, the testing instance will contain the attributes whose status is “Selected” and performs the SVM on that instance. Finally, for all the GO term attributes reassigned the status as “Selected”, for the next testing instance preparation.

The advantage of these attribute selection methods with the use of SVM, it gives the clear margin of separation and it is effective in high dimensional areas. The main advantage is provides the superior performance in organizing the genes.

IV. EXPERIMENTAL RESULTS

By the use of SVM algorithm, most of the compatible attributes are ejected and genes are classified into either extended lifespan or reduced lifespan with a separable line. Specificity is calculated by using (1)

$$Specificity = \frac{TruePositiveGenes}{PositiveGenes} \qquad (1)$$

Sensitivity is calculated by using (2)

$$Sesnsitivity = \frac{TrueNegativeGenes}{NegativeGenes} \qquad (2)$$

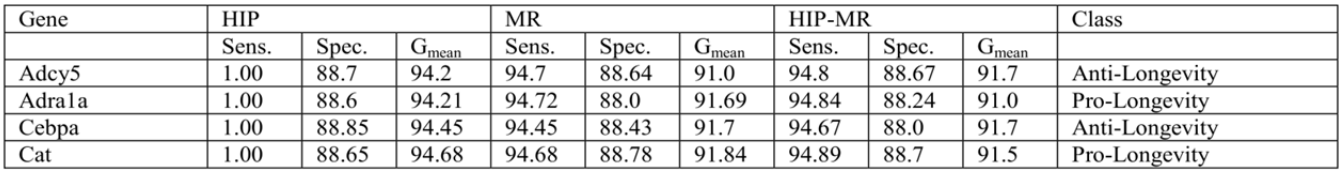
Evaluating the performance of the classifier by the value of G_{mean} and it is calculated by using (3)

$$G_{mean} = \sqrt{Sensitivity} * \sqrt{Specificity} \qquad (3)$$

V. ANALYSIS FOR ATTRIBUTE SELECTION METHODS WITH SVM

The analysis for the mus (mouse) genes are classified by the three attribute selection methods with SVM as shown in the below Table 1.

TABLE I. SENSITIVITY (%), SPECIFICITY (%), AND G_{MEAN} (%) OF ATTRIBUTE SELECTION METHODS WITH SVM FOR GENES MUS (MOUSE)



Gene	HIP			MR			HIP-MR			Class
	Sens.	Spec.	G_{mean}	Sens.	Spec.	G_{mean}	Sens.	Spec.	G_{mean}	
Adcy5	1.00	88.7	94.2	94.7	88.64	91.0	94.8	88.67	91.7	Anti-Longevity
Adra1a	1.00	88.6	94.21	94.72	88.0	91.69	94.84	88.24	91.0	Pro-Longevity
Cebpa	1.00	88.85	94.45	94.45	88.43	91.7	94.67	88.0	91.7	Anti-Longevity
Cat	1.00	88.65	94.68	94.68	88.78	91.84	94.89	88.7	91.5	Pro-Longevity

VI. CONCLUSION

The experimental results shown that the attribute selection methods (HIP and MR) with the use of SVM give more efficiency to organize the model organism's genes into any one of the category, than with the use of Naïve Bayes. These methods make use of repetitious among the GO terms and reduce the repetitious in the set of selected attributes.

ACKNOWLEDGMENT

I would like to express my gratitude to my Guide Mrs. T. Sudha Rani, M. Tech.,(Ph.D), Sr. Assistant Professor, without her guidance, this paper is not possible. Her understanding, encouragement and personal guidance have provided the basis for this paper.

REFERENCES

[1] Attribute Selection for Knowledge Discovery and Data Mining. Norwell, MA, USA: Kluwer, 1998 by H. Liu and H. Motoda.

[2] "Human senescence genomic resources: Integrated databases and tools for the biology and genetics of senescence," Nucleic Acids Res., vol. 41, no. D1, pp. D1027–D1033, Jan. 2013 by R. Tacutu, T. Craig, A. Budovsky, D. Wuttke, G. Lehmann.

[3] "Predicting the Pro-Longevity or Anti-Longevity Effect of Model Organism Genes with new Hierarchical Feature Selection Methods" IEEE/ACM Transactions on computational biology and bioinformatics, vol. 12,no. 2, March/April 2015 by Cen Wan, Alex A. Freitas, and Joao Pedro de Magalhaes.

[4] "Human senescence genomic resources: Integrated databases and tools for the biology and genetics of senescence," Nucleic Acids Res., vol. 41, no. D1, pp. D1027–D1033, Jan. 2013 by R. Tacutu, T. Craig, A. Budovsky, D. Wuttke, G. Leh"Prediction of the pro-longevity or anti-longevity effect of Caenorhabditis Elegans genes based on Bayesian classification methods," in Proc. IEEE Int. Conf. Bioinformatics. Biomed. Shanghai, China, Dec. 2013, pp. 373–380 by C. Wan and A. A. Freitas.mann.

[5] "Attribute selection for the naive Bayesian classifier using decision trees," Appl. Artif. Intell. vol. 17, no. 5-6, pp. 475–487, Nov. 2003 by C. A. Ratanamahatana and D. Gunopulos.

[6] "Lazy attribute selection: Choosing attributes at classification time," Intell. Data Anal., vol. 15, no. 5, pp. 715–732, Aug. 2011 by R. B. Pereira, A. Plastino.

AUTHORS PROFILE

Ch. Uma Shankari received the B. Tech. Degree in Computer Science and Engineering from Pragati Engineering College permanently affiliated to J.N.T.U. Kakinada, Andhra Pradesh, India. Presently working for M. Tech. Degree in Computer Science and Engineering at Aditya Engineering College, affiliated to J.N.T.U. Kakinada, Andhra Pradesh, India.

T. Sudha Rani working as Sr. Assistant Professor in Aditya Engineering College, Surampalem, Andhra Pradesh, India. She completed M. Tech., working for Ph.D Degree and doing research on data mining and bio-informatics.