IJAMML 15:1 (2021) 53-67 September 2021
ISSN: 2394-2258
Available at http://scientificadvances.co.in
DOI: http://dx.doi.org/10.18642/ijamml_710012230

MACHINE LEARNING METHOD TO DIFFERENTIATE ATAXIAS

Gustavo Simões Carnivali

Universidade de Minas Gerais - Belo Horizonte - Brazil, Av. Pres. Antônio Carlos, 6627 - Pampulha, Belo Horizonte - MG, 31270-901, Brazil

Abstract

Spinocerebellar ataxias or SCAs, are a group of more than 37 genetically and clinically heterogeneous known neurodegenerative diseases. This work analyzes the level of genetic similarity between several ataxias, we identified proteins that are associated with more than one ataxia. A decision tree was trained to identify ataxias by identifying whether a new entry disease not yet identified and not classified can be grouped as an ataxia. Altogether 12 proteins from different ataxias were verified, all 12 proteins were analyzed in 500 different trees to better evaluate the method used. Of the 12 proteins tested, the method was correct for 10 different proteins or 83% of correct results. This identifier and the results obtained in the experiments allow a greater characterization of the diseases addressed, it also allows applications such as the reuse of treatments for similar diseases.

Keywords: machine learning, ataxias, decision tree.

[*] Corresponding author.

E-mail address: gustavocarnivali@gmail.com (G. S. Carnivali).

2020 Mathematics Subject Classification: 68.
Submitted by Jianqiang Gao.
Received September 21, 2021; Revised September 30, 2021

1. Introduction

Spinocerebellar ataxias or SCAs are a group of more than 37 known neurodegenerative diseases that are genetically and clinically heterogeneous. The most common type among SCAs has an occurrence of 1 to 5 cases per 100,000 people. They commonly affect the nervous system, causing loss of coordination [1].

This work analyzes the level of genetic similarity between several ataxias by identifying proteins that are associated with more than one of them. Once, it has been shown that ataxias have many proteins in common [2]. In the study of [3], it was shown that ataxias also have more genes related to each other compared to other diseases. This study intends to evolve the studies carried out in [2, 3], using machine learning tools to characterize the ataxias presented.

Studies have shown that genes can be related to other genes, that is, the increase in the action of one gene can lead to an increase or decrease in the phenotypic effects of another gene [4]. From this feature a network of gene or protein interactions can be generated. Thus, a decision tree was trained to identify ataxias in order to establish whether or not an as-yet unidentified and unclassified new entry disease can be grouped as an Ataxia [5].

Some studies such as [16] already use machine learning methods to better analyze diseases, the study presented here can also be used for this purpose, but its main objective, not addressed in [16], as well as not in related works, is the comparison between already known diseases. Studies such as [17, 18, 19] uses machine learning methods to compare diseases and can and were used as a basis for the composition of this study, but they do not present comparative studies on the studied diseases (Ataxias), as well as, do not use genetic characteristics in comparisons. This study therefore generates a new method that can be applied to other diseases because it is precise and uses a quick and simple data structure, but specifically, in this study, it was applied to a set of diseases not yet studied by machine learning methods.

This classification allows for greater understanding of the group of diseases studied and also allows applications such as the reuse of drugs/treatments for similar diseases that were or will be analyzed by the generated tree, as well as an understanding of why this indication is functional based on genetic factors of the disease [6].

2. Methodology

2.1. Protein networks

Protein relationships can occur in different ways in our organism, a common way, which can be mentioned, is the gene co-expression [7]. Gene expression can be interpreted as the process by which DNA nucleotide sequences are transcribed into RNA used by the cell or RNAs that are translated into proteins [8]. A change in the expression of one gene can increase or decrease the expression of other genes [9]. This correlation between genes can be represented by an interaction graph. A graph is a mathematical structure G = (V, E) where V is a non-empty set of vertices that can represent genes or proteins for example and E are edges that connect two vertices and indicate a relationship between them, just as they can represent a co-expression relationship between two genes [1]. An edge, in a genetic or protein network, may have associated with it a weight that corresponds to the confidence of the co-expression of the gene [7].

In this work we use the String-DB database to generate the biological networks. String-DB is a database containing thousands of protein-protein interactions. It also includes a score that associates a degree of confidence in its occurrence to each interaction. String-DB calculates this score, a value between 0 and 1, using different prediction approaches and different databases such as NCBI Gene Expression Omnibus, ProteomeHD, PubMed, Ensembl, SwissProt [7]. We consider this confidence value as the weight of the edges of our graph G. Only connections with a confidence above the threshold defined by 0.5 were considered to increase reliability.

In Figure 1, a biological network obtained by the string platform for the connections of the protein that causes squamous cell ataxia type 1 (ATXN1) is represented.

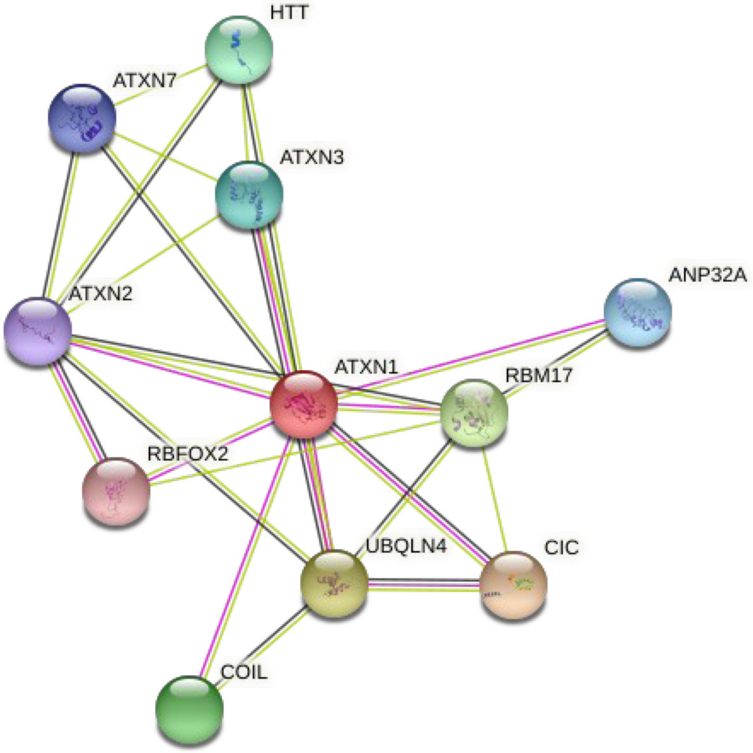

Figure 1. Main connections of protein caused of the Spinocerebellar ataxia type 1 (ATXN1). Edge scores are not shown in the figure, but for example, score between ATXN7 protein and HTT explained by PubMed ([11]): "score 0.810. Putative homologs are mentioned together in other organisms (score 0.092)".

2.2. Representation: Graph as a vector

The machine learning methods used require the input data to be vectors with characteristics. Unfortunately a graph is not naturally synthesized by this format. There are programs that use the graph structure to create feature vectors that can be used as input to machine learning methods [11]. In this study, the program presented in [11] called NBNE (Neighborhood Based Node Embeddings) was used. The NBNE in its own study is compared to other current methods and surpasses them. NBNE creates a vector with 128 features for each vertex of the graph.

Unfortunately, NBNE does not allow an easy interpretation of the created vectors, that is, NBNE creates a vector for each edge of the graph with values that imply complex properties of the network topology, these properties, however, do not allow an easy biological interpretation of the values found, its main focus is on the quality of the representation and not on the interpretation of the results generated. However, NBNE was chosen in this work because it is a current method and presents degrees of efficiency in similar applications that are more suitable than other existing methods [11].

2.3. Machine learning methods

The method used in this work is classified as supervised. This is also possible thanks to the representation presented in Subsection 2.2, which allows the representation of a graph as a numerical vector of features. This class of methods features a training phase and a testing phase. In the training phase, in general, the obtained data are divided and, based on classical methods, a machine is taught to classify new data based on the teaching generated by the previous data [12]. In the case of this work, we want to develop a machine capable of telling whether a new input protein causes an ataxia or not. For this, the training set will contain genes from the input graph that are linked or not to an ataxia. Therefore, the objective is to know whether the data used to develop the network are sufficient to identify this class of diseases and, in parallel, identify the characteristics that differentiate this group of diseases from others.

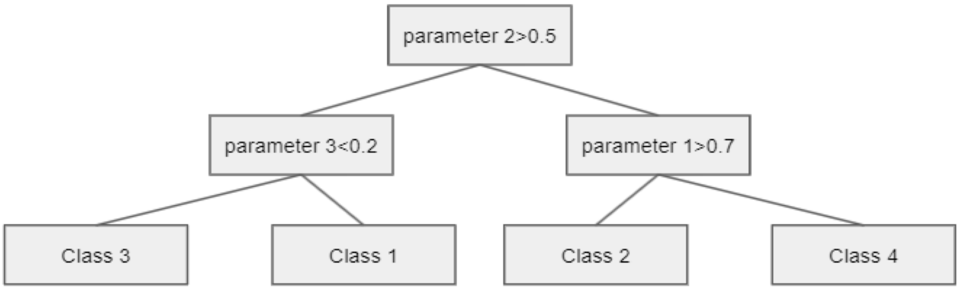

Figure 2. Example decision tree.

A recurrent problem that can occur in machine learning is that of overfitting. In our context, a method that produces overfitting would generate a machine that would memorize the input data, being unable to recognize other ataxias, besides those used as input, which have few significant differences. Therefore, overfitting should be avoided in this study, aiming to reduce this problem in this work, a method that classically generates little overfitting was chosen [12].

The machine learning method used is the decision tree. This method was chosen over the others because it presents an easy biological interpretation of the results obtained, that is, from the characteristics of the tree it can be more easily implicated in the characteristics of the studied diseases. Therefore, in addition to the main objective of this study of classifying diseases, deep interpretations and the generation of the tree will allow for a better understanding of this class of diseases that is still poorly studied and known (i.e., ataxias) [13].

In addition, the tree used has already obtained several improvements. Comparing it to other methods by the accuracy metric that will be described in the future, the tree managed to obtain significant quality results in relation to the other tested methods.

2.4. Decision tree

The decision tree is a tree of choices about stacked decisions. At each node the tree asks a question according to the input data parameters, if the answer is positive the user goes to the left of the tree and if the answer is negative the user goes to the right of the tree. The process is repeated until the user arrives at a node that no longer has children called a leaf [13].

A visual example of a tree can be seen in Image 2. For example, if the input data has a value of parameter 2 greater than 0.5 it will go to the node to the right of the current node, if the value of this parameter is smaller that 0.5 it will go left. As an application example, for this tree, if a data has a parameter 2 value less than 0.5 and a parameter 3 value greater than 0.2 it will be classified as class 1.

The continuous growth of the tree with many decision nodes can generate overfitting because the tree would decorate the input data, preventing its generalizability [13]. An adequate height of the tree must be chosen by the user from the knowledge of his data [13]. It will be seen that in fact the tree generated by this study does not have a great height.

For the creation of the decision tree (i.e., correct positioning of the nodes) a classic method used in this work can be used. Basically, the method calculates the entropy of each question and gives preference to questions with higher entropy to assume higher positions in the tree. Entropy, among other things, considers the number of data that will be directed to the right or left child in a tree [13].

The accuracy metric will be used in this work, it serves to identify the efficiency of the method used. After the method is trained, some test data are delivered to the method, if it manages to classify these data correctly its accuracy increases, if its classification is wrong its accuracy decreases linearly, that is, for example, 100% accuracy means that the algorithm got all the possibilities right, 50% accuracy means that the algorithm got half of all the possibilities right.

2.5. Cross validation

In order to improve probabilistic results about the efficiency of a method on a dataset, cross-validations can be used. Owned input data can be divided into 2 groups: training data and testing data, but how does this division occur? From iterative methods, training and testing groups are created based on the input data and different at each iteration, thus allowing the evaluation of the method in a more independent way from the data used.

In this study, we want to know which data will be present in each set, in order to be able to characterize the Ataxias exclusively. Therefore, no classical cross-validation method will be used, but your idea, for bringing benefits to the experiments, will be used to assemble the test and training sets used. How the sets were assembled will be better described in the next section of Experiments (Section 4).

3. Algorithm

The algorithm that implements the project was entirely made in Python. Python has libraries that allow easy use of machine learning methods, including the decision tree. From sklearn, the code presented by Algorithm 1 was produced and it manages to synthesize all the methodology proposed by this work.

The algorithm (exposed by Algorithm 1) basically reads and stores the input data in lines 1 to 3 using the NBNE. Lines 4 to 6 create and use the tree based on functions provided by the sklearn library. Finally, line 7 calculates the final result. Naturally, Algorithm 1 has been simplified by hiding unnecessary secondary functions for understanding the method.

Algorithm 1: Decision tree

```
X train = <reading the data> % input data read
Y train = <reading the data>
Y pred = <reading the data>
model = Decision TreeClassifier() %tree creation
model = model.fit (Xtrain, Ytrain)
Y pred = model.predict(Xtest) %creation of the prediction variable
result = result + (accurancy_score(Ytest, Ypred))% accuracy calculation
```

Algorithm 1: Proposed method.

4. Experiments and Results

In this study, we used a set of 12 proteins directly linked to some ataxia. The types of ataxia used are considered monogenetic diseases that are caused by the mutation of a single gene, for each ataxia used, at least one protein is known to cause or influence it. The proteins used are described in Table 1 [15]. Another set of 24 proteins not directly bound to ataxias and chosen at random from all the existing set of proteins in the human body was used. Initially, a training group was created containing 9 proteins from the first group and 18 proteins from the second, chosen at random from the whole set, the random subdivision was done 30 times with the aim of bringing more confidence to the results found.

The decision tree was applied to this iterated set of 27 proteins as described above. The remaining pool of 9 proteins was used as a test to evaluate the method used. To assess tree size, 500 trees with sizes ranging from 1 to 500 were generated, one tree for each specified size. For the 500 trees with 30 iterations each, the highest average accuracy obtained for the specified data was 0.89.[1]

Each tree height level obtains a distinct accuracy value. The variance of the calculated accuracy of all 500 trees generated in the algorithm was only 0.0016 showing that: the tree quality varied little with the change of its height, the tree quickly obtained good quality even with few nodes and, finally, the tree did not significantly present the overfitting problem. To facilitate viewing the result, it is shown in Figure 3.

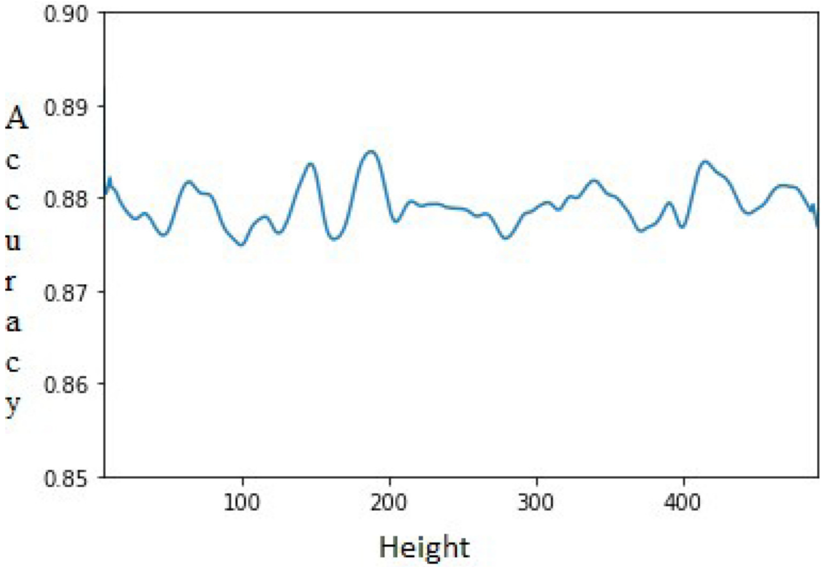

Figure 3. Accuracy value variation in a decision tree with height variation from 1 to 500.

[1] The average was used because each tree generated 30 results, one for each iteration. Measures such as variance or confidence interval will not be presented throughout the work as they obtained low and irrelevant values for the results found.

The characteristic of getting a good result even with low tree height can be a property of ataxias. If the ataxias have more similarities as described in studies [2, 3] the decision tree, even with few vertices, would achieve a good result. Furthermore, based on this result, one can assume the existence of a strong characteristic that approximates the Ataxias that was found by the decision tree and that was used by it, thus, even with the addition of new vertices, new choices did not generate other significant separations. However, all results can also be explained by technical factors such as the low dimensionality and diversity of training and test data.

A second set of tests was performed. In order to determine which of the tested ataxias differs the most from the others in the test set, 12 sets with all 24 chosen proteins plus 12 – 1 = 11 ataxias proteins were created. Twelve experiments were carried out removing in each experiment a protein related to an ataxia. The tree was trained with the height that obtained the best accuracy performance in the previous test. The result for the 12 proteins can be seen in Table 1.

Table 1. As predicted by the previous experiment, this experiment achieved a high hit rate of 83%. Among the 12 ataxias tested, only 2 had misclassification, Ataxias of type 10 and 11 represented by proteins ATXN10 and TTBK2

Table 1. Search results

Disease	SCA2	SCA2	SCA3	SCA17	SCA3	SCA7	SCA36	SCA7	SCA10	SCA11	SCA1	SCA7
Protein	ATXN2	ATXN2L	ATXN3L	TBP	ATXN3	ATXN7L2	NOP56	ATXN7	ATXN10	TTBK2	ZNF674	ATXN7L3
Result	Right	Right	Right	Right	Right	Right	Right	Right	Missed	Missed	Right	Right

5. Conclusion

This work presented a way to classify, from a machine learning method, a set of neurodegenerative and genetic diseases from a set of a biological interaction network, specifically, the set of diseases called ataxias was studied.

The machine learning method used was the decision tree. This method was chosen mainly for its easy interpretability and explainability, but also for obtaining adequate results in relation to other existing methods.

This work performed two sets of experiments on a set of 12 proteins acting on several ataxias. From the first experiment, it was possible to conclude that the ataxias are well separated even by decision trees with few vertices, showing the existence of a possible strong similarity between the Ataxias. As a result of the second set of experiments, a possible lesser similarity was seen between ataxias of types 10 and 11 in relation to the other ataxias tested considering the proteins used.

In the future, the data and methodology presented may support comparative studies of other diseases. The classifier generated, despite having good experimental quality, can be retrained using more proteins not bound to ataxias or, if new studies are carried out that analyze this class of diseases, other proteins also bound to ataxias can be used, as well, to other proteins from diseases close to the ataxias.

6. Acknowledgement

Capes, Fapemig and UFMG.

References

[1] H. L. Paulson, The spinocerebellar ataxias, Journal of Neuro-Ophthalmology: The Official Journal of the North American Neuro-Ophthalmology Society 29(3) (2009), 227-237.

DOI: https://doi.org/10.1097/WNO0b013e3181b416de

[2] J. Lim, T. Hao, C. Shaw, A. J. Patel, G. Szabó, J.-F. Rual, C. J. Fisk, N. Li, A. Smolyar, D. E. Hill, A.-L. Barabási, M. Vidal and H. Y. Zoghbi, A protein-protein interaction network for human inherited ataxias and disorders of Purkinje cell degeneration, Cell 125(4) (2006), 801-814.

DOI: https://doi.org/10.1016/j.cell.2006.03.032

[3] G. S. Carnivali and S. V. A. Campos, Does the ataxia group have genetic similarities?, Anais do XIII Encontro Academico de Modelagem Computacional (2020), 135.

[4] B. Snel, G. Lehmann, P. Bork and M. A. Huynen, STRING: A web-server toretrieve and display the repeatedly occurring neighbourhood of a gene, Nucleic Acids Research 28(18) (2000), 3442-3444.

DOI: https://doi.org/10.1093/nar/28.18.3442

[5] M. De Souto, A. Lorena, A. Delbem and A. de Carvalho, Tecnicas de aprendizado de maquina para problemas de biologia molecular, Sociedade Brasileira de Computacao 1(2) (2003).

[6] M. Fatima and M. Pasha, Survey of machine learning algorithms for disease diagnostic, Journal of Intelligent Learning Systems and Applications 9(1) (2017), 1-16.

DOI: https://doi.org/10.4236/jilsa.2017.91001

[7] D. Szklarczyk, A. Franceschini, S. Wyder, K. Forslund, D. Heller, J. Huerta-Cepas, M. Simonovic, A. Roth, A. Santos, K. P. Tsafou, M. Kuhn, P. Bork, L. J. Jensen and C. Von Mering, STRING v10: protein–protein interaction networks, integrated over the tree of life, Nucleic Acids Research 43(D1) (2015), 447-452.

DOI: https://doi.org/10.1093/nar/gku1003

[8] A. Brazma and J. Vilo, Gene expression data analysis, FEBS Letters 480(1) (2000), 17-24.

DOI: https://doi.org/10.1016/s0014-5793(00)01772-5

[9] S. Horvath and J. Dong, Geometric interpretation of gene coexpression network analysis, PLoS Computational Biology 4(8) (2008); e1000117.

DOI: https://doi.org/10.1371/journal.pcbi.1000117

[10] G. Goeckenjan, H. Sitter, M. Thomas, D. Branscheid, M. Flentje, F. Griesinger, N. Niederle, M. Stuschke, T. Blum, K.-M. Deppermann, J. H. Ficker, L. Freitag, A. S. Lübbe, T. Reinhold, E. Späth-Schwalbe, D. Ukena, M. Wickert, M. Wolf, S. Andreas, T. Auberger, R. P. Baum, B. Baysal, J. Beuth, H. Bickeböller, A. Böcking, R. M. Bohle, I. Brüske, O. Burghuber, N. Dickgreber, S. Diederich, H. Dienemann, W. Eberhardt, S. Eggeling, T. Fink, B. Fischer, M. Franke, G. Friedel, T. Gauler, S. Gütz, H. Hautmann, A. Hellmann, D. Hellwig, F. Herth, C. P. Heußel, W. Hilbe, F. Hoffmeyer, M. Horneber, R. M. Huber, J. Hübner, H.-U. Kauczor, K. Kirchbacher, D. Kirsten, T. Kraus, S. M. Lang, U. Martens, A. Mohn-Staudner, K.-M. Müller, J. Müller-Nordhorn, D. Nowak, U. Ochmann, B. Passlick, I. Petersen, R. Pirker, B. Pokrajac, M. Reck, S. Riha, C. Rübe, A. Schmittel, N. Schönfeld, W. Schütte, M. Serke, G. Stamatis, M. Steingräber, M. Steins, E. Stoelben, L. Swoboda, H. Teschler, H. W. Tessen, M. Weber, A. Werner, H.-E. Wichmann, E. Irlinger Wimmer, C. Witt and H. Worth, Prävention, Diagnostik, Therapie und Nachsorge des Lungenkarzinoms, Pubmed Results, Pneumologie 65(8) (2011), e51-e75.

DOI: https://doi.org/10.1055/s-0030-1256562

[11] T. Pimentel, A. Veloso and N. Ziviani, Fast node embeddings: Learning egocentric representations, 2018.

[12] M. C. Monard and J. A. Baranauskas, Conceitos sobre aprendizado de maquina, Sistemas Inteligentes: Fundamentos e Aplicacoes 1(1) (2003), 32.

[13] J. Gama, Arvores de decisao, Palestra ministrada no Nucleo da Ciencia de Computacao da Universidade do Porto, Porto, 2002.

[14] G. Keijzers, D. Bakula and M. Scheibye-Knudsen, Monogenic diseases of DNA repair, New England Journal of Medicine 377(19) (2017), 1868-1876.

DOI: https://doi.org/10.1056/NEJMra1703366

[15] J. M. Stuart, E. Segal, D. Koller and S. K. Kim, A gene-coexpression network for global discovery of conserved genetic modules, Science 302(5643) (2003), 249-255.

DOI: https://doi.org/10.1126/science.1087447

[16] A. Dagliati, S. Marini, L. Sacchi, G. Cogni, M. Teliti, V. Tibollo, Pasquale de Cata, Luca Chiovato, and Riccardo Bellazzi, Machine learning methods to predict diabetes complications, Journal of Diabetes Science and Technology 12(2) (2018), 295-302.

DOI: https://doi.org/10.1177/1932296817706375

[17] S. Uddin, A. Khan, M. E. Hossain and M. A. Moni, Comparing different supervised machine learning algorithms for disease prediction, BMC Medical Informatics and Decision Making 19(1) (2019), 1-16.

DOI: https://doi.org/10.1186/s12911-019-1004-8

[18] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia and J. Gutierrez, A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease, In 2017 IEEE Symposium on Computers and Communications (ISCC) (2017), pp. 204-207.

DOI: https://doi.org/10.1109/ISCC.2017.8024530

[19] M. Brahimi, K. Boukhalfa and A. Moussaoui, Deep learning for tomato diseases: Classification and symptoms visualization, Applied Artificial Intelligence 31(4) (2017), 299-315.

DOI: https://doi.org/10.1080/08839514.2017.1315516

■