Detection of Francisella Tularensis Pathogen in Soil using Neural Networks

Muhammad Shahbaz[1], Sajida Parveen[1], Fareed Ahmad[1] and Masood Rabbani[2]

[1] Department of Computer Science & Engineering University of Engineering and Technology Lahore, Pakistan
[2]Faculty of Veterinary Sciences University of Veterinary and Animal Sciences, Lahore, Pakistan

Abstract: *Francisella Tularensis is a dangerous bacterium that can cause diseases in both humans as well as animals. The detection of Francisella Tularensis in soil is vital to prevent widespread epidemic of Tularemia disease. Classification of soil samples based on its characteristics can be helpful in initial detection of this pathogen. In this paper, we study the detection of Francisella Tularensis in soil using backpropagation neural networks. As number of neurons and hidden layers along with activation and loss function play an important role in the performance of networks, different experiments were conducted to study their effects. It is concluded that a backpropagation network having one hidden layer with ten neurons performs best for our dataset when activation function is tanh and loss function is absolute. The network with these configurations gives an accuracy of 82.61 % for ten-fold cross validation.*

Keywords: *Francisella Tularensis, Soil Classification, Artificial Neural Network, Backpropagation Algorithm*

1. Introduction

Francisella Tularensis is among the most virulent pathogens ever known [1]. Francisella Tularensis is a hardy organism bacterium with a life of weeks and months at low temperatures in soil, water and animals remaining. The bacterium is highly threatening for humans and animals [2].

Out of the four common subspecies of Francisella Tularensis, Tularensis (Type-A) is the most virulent pathogen in humans as well as animals known in medical science. Even a small dose of bacteria in humans can cause infection. It has been reported that as minimum as 10 to 25 bacteria directly injected into the human body or given through air can be fatal [2] [3]. The natural reservoirs for the bacteria are small mammals such as rabbits, hares, rodents, mice and squirrels. The animals acquire the disease through bites of fleas, mosquitoes and ticks.

Tularemia disease, a zoonotic disease, is caused by Francisella Tularensis. It can cause severe and prolonged fever that can last from weeks to months. This disease is most commonly acquired through bites from arthropods, such as fleas, mosquitoes and ticks. It can also be transmitted through direct contact, inhalation, and ingestion of contaminated animal tissue, soil or water. There is no reported case of Man-to-man transmission of Francisella Tularensis [1].

The detection of Francisella Tularensis in soil can help in prevention of an outbreak of Tularemia disease. The most commonly used methods for detection of Francisella tularemia are Polymerase chain reaction (PCR) [4], enzyme-linked immunosorbent assay (ELISA) [5] and Mass spectroscopy (MS) [6]. These methods require costly equipment and take a lot of time. Even once the equipment is available, it costs between 5-15 thousand dollars for testing each sample. The cost can certainly be reduced by performing tests on only those samples of soil which have more likelihood of having Francisella Tularensis. This initial filtering can be carried out by classifying samples of soil based on its characteristics. In this paper, such classification of samples using neural networks has been studied.

The rest of the paper is structured as follows: Section 2 describes similar research work carried out for detection of the pathogen. Section 3 discusses our methodology to classify soil samples for presence of

https://doi.org/10.17758/URUAE1.E0518002

Francisella Tularensis. Section 4 presents the results of different experiments conducted on our dataset. Finally, Section 5 concludes our discussion.

2. Literature Review

Several machine leaning approaches have been applied to similar problems of soil classification. The study [7] used Self-Organizing Map (SOM) to detect the effects of chemical concentration of some elements for the soil sample in different areas. SOM clustering showed that the concentration of elements in the soil increased near roads and other infrastructures. K-mean and hierarchical clustering has been used in clustering of agriculture soil dataset [8], [9]. They compared the results of both clustering approaches with different distance measures and found that K-mean performed better.

Cone Penetration Testing (CPT) dataset has been classified using Decision Tree (DT), neural network and Support Vector Machine (SVM) [10]. Before classifying the data, they segmented the data into clusters taking into account the contiguity of each cluster from others. They also extracted features based on manual classification by interviewing geological experts. After defining the parameters, they trained their classifier on labeled data marked by experts using DT, ANN and SVM algorithms. Three classification problems were contemplated: binary classification to classify soil data into sandy or not, three-class classification to identify the primary soil classes (Sand, Clay, and Peat), and seven-class classification to determine the appropriate class observed in the area. The overall results showed that neural networperformed better than SVM and DT.

3. Methodology

Classification is a machine learning problem and various algorithms and techniques have been used for it. Tradition- ally, Naive Bayes and Support Vector Machines have been used for classification. However, more complex and robust techniques like neural networks are being used these days due to fast computing machines available. Artificial Neural Network (ANN) is inspired by biological nervous system. It comprises of large number of highly interconnected elements called neurons. ANN s learn by training patterns as is the case with humans. Learning process involves finding out the optimal weights of the connections between neurons.

Neural Networks are generally organized in layers where each layer consists of neurons. Training patterns are fed to the network using an input layer which is either connected to some hidden layer(s) or to the output layer. Fig. 1 shows an architecture of a neural network with one hidden layer. This paper focuses on a certain type of learning called backpropagation.

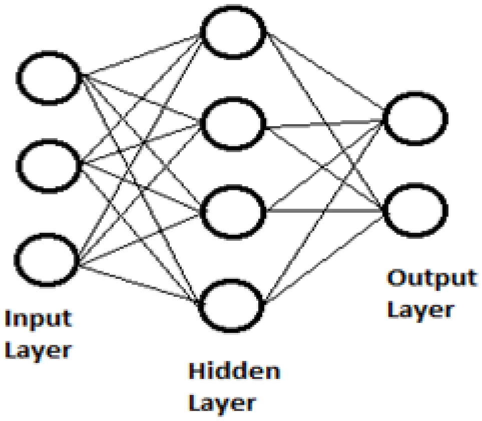

Fig. 1. Architecture of neural networks

3.1. Backpropagation

Please submit your manuscript electronically for review as e-mail attachments. When you submit your initial full paper version, prepare it in two-column format, including figures and tables.

Backpropagation algorithm, originally introduced in 1970s, but became popular after the study which discusses several networks where backpropagation works faster than previously known approaches [11]. Today, backpropagation is the most widely used method of learning in neural networks. The error in such type of Neural Network is propagated from output layer to the input layer passing through any hidden layers. The primary idea behind backpropagation is to divide the process utilizing the chain rule to make it modular so that it can be conveniently used for larger networks.

The two major steps involved in backpropagation are feedforward and backpropagation. In feedforward step, we present a training example to the network, which processes the input and finds out the actual response of the system. Then, an error value is calculated from the known output of that training example and the actual response generated by the network using some loss or cost function. Now, the weights are adjusted minimize the error.

Backpropagation uses gradient descent which requires the calculation of derivative of squared loss/cost function with respect to the weights of the network. Then, each weight is updated according to its contribution in the original output such that loss function is minimized. Fig. 2 shows an overview of backpropagation.

Here, ε_2 is the error propagated to hidden layer h1 while is the error propagated to layer h1, Δw_2 and Δw_1 are the respective weight adjustment values.

$$\varepsilon_2 = \Delta L \; f'(z_2) \qquad (1)$$

$$\varepsilon_1 = \varepsilon_2 \, w_2 \; f'(z_1) \qquad (2)$$

$$\Delta w_2 = \varepsilon_2 \, h_1 \qquad (3)$$

$$\Delta w_1 = \varepsilon_1 \, h_0 \qquad (4)$$

Initially, small and random values are used to initialize network weights. For a given network, the algorithm iterates repeatedly until a stopping condition is found. Stopping condition can be a minimum value of loss function or a specific number of iterations. For each such iteration called epoch, all training examples are presented to the network and their output is computed using feedforward pass. Then, the weights can be updated either after each input pattern or after all input patterns (in this case, cumulative error is used for backpropagation). For our network, stopping condition is 1000 epochs and weights are updated after each input example. For testing a pattern, only forward pass is done to calculate the output.

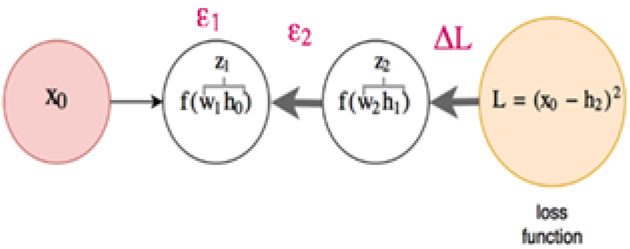

Fig. 2. Overview of backpropagation process

3.2. Dataset

The dataset consists of different soil samples collected from various locations. The dataset has 147 total samples with 74 samples where Francisella Tuileries was present and 73 samples where bacterium was not found. The attributes include pH value, physical characteristics and quantity of different salts and metals in the soil. The list of all 21 attributes is given in Table I.

TABLE I
ATTRIBUTES IN EXPERIMENTAL DATASET

S. No.	Attribute Name	S. No.	Attribute Name
1	pH	12	P(Phosphorous)
2	Sand	13	Ni (Nickel)
3	Silt	14	Cd(Cadmium)
4	Clay	15	Fe (Iron)
5	Soluble Salts	16	Ca (Calcium)
6	Moisture	17	Mg (Magnesium)
7	Organic matter	18	Pb (Lead)
8	Cu (copper)	19	Na (Sodium)
9	Cr(Chromium)	20	Zn(Zinc)
10	Mn (Manganese)	21	K (Potassium)
11	N (Nitrogen)		

4. Experiments and Results

Neural network with backpropagation has been used to classify the above discussed dataset. We have studied different variations of neural network and analyzed their accuracy on the dataset. For baseline classification, naive Bayes classification has been used. It gives an accuracy of 57% which is not much better and there is quite room for improvement.

4.1. Experimental Conditions

We have used IO-fold cross validation mechanism with shuffled sampling for all experiments. The number of epochs used for all experiments are 1000. Weights in the network are updated after presenting each training pattern. Standard backpropagation algorithm updates weights in the direction of maximum decrease in error. According to study [12], this standard approach does not move weights directly towards optimal weights. One proposed method to improve the speed of training is to change the learning rate during training. For classification problems where training examples are fewer, adaptive learning is useful. As the training examples are fewer in our case, we have used adaptive learning.

4.2. Variations of Backpropagation Neural Network

The different variations of backpropagation neural net- work has been studied which are discussed in subsequent subsections.

1) Number of neurons

The number of neurons in a hidden layer can affect the performance of a neural network. Different number of neurons in a single hidden layer has been used and their accuracies are displayed in Fig. 3. It is clear that time taken in all these experiments is almost same. The network seems to operate best for ten neurons in a single hidden layer. In further experiments discussed below, ten neurons have been used in each layer.

https://doi.org/10.17758/URUAE1.E0518002

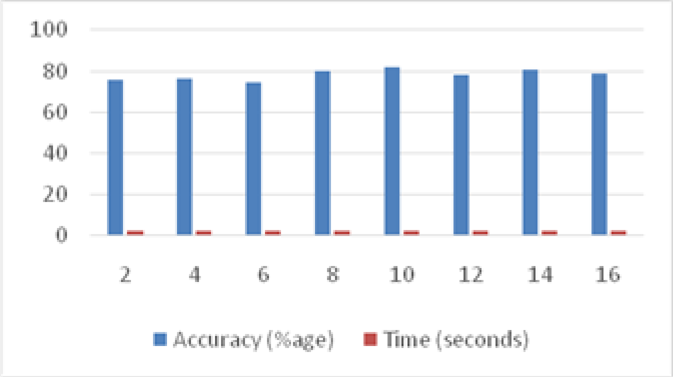

Fig. 3. Accuracy and time against number of neurons

2) Number of hidden layers

The more a neural network has hidden layers, the more complex a network becomes and it can solve more complex problems. But increasing number of hidden layer does not simply improve the accuracy of a network. When number of hidden layers are increased, number of weights between neurons also increases. This requires that enough training patterns are available to update the weights to their optimal values. Usually, one hidden layer is enough for most of the problems. More hidden layers are often used to speed up the training process. The dataset has been trained and tested with different The execution time remains almost same for all experiments. The network seems to perform best for single hidden layer. The reason probably lies in the fact that we have few training patterns and all weights are not updated to their optimal values for multiple hidden layers. In further experiments discussed below, single hidden layer with 10 neurons has been used.

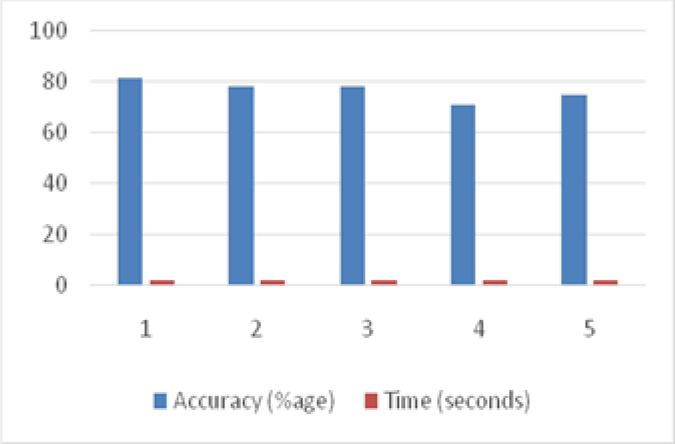

Fig. 4. Accuracy and time against number of hidden layers

3) Activation functions

Activation functions are used to transform the activation level of a neuron into output response. Some of the commonly used activation functions have been used for training and testing for our dataset. The results are displayed in Fig. 5. Tanh and Maxout activation functions perform better than Rectifier and Exponential Rectifier activation functions. The execution time for all activation functions is almost similar. As Tanh performs slightly better than Maxout, tanh function has been used in all the below discussed experiments.

https://doi.org/10.17758/URUAE1.E0518002

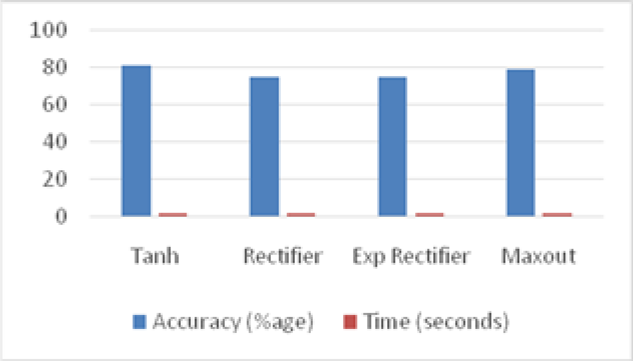

Fig. 5. Accuracy and time against different Activation factions

4) Loss function

The loss function is used to calculate difference between expected output of a training pattern and actual output of network after the forward pass. The value of this loss function is then propagated back in the network to update weights. Different type of loss functions have been used to train the network and their results are shown in Fig. 6. It is evident that absolute and Huber loss functions perform better than Quadratic and Cross Entropy functions in terms of accuracy as well as execution time.

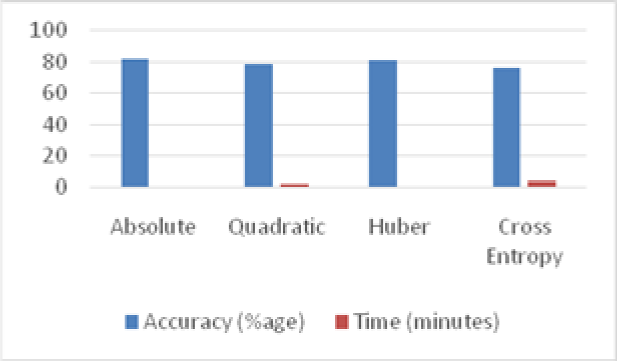

Fig. 6. Accuracy and time against different loss functions

5. Conclusion

Detection of Francisella Tularensis pathogen in soil using neural network techniques has been studied in this paper. The dataset for detection of Francisella Tularensis pathogen was classified using naive Bayes algorithm which gave an accuracy of 57%. In order to improve the accuracy, different combinations of neural networks in terms of number of neurons, number of hidden layers, activation functions and loss functions have been tried out. Out of all experiments conducted, the highest accuracy achieved is 82.61 % by using a single hidden layer with 10 neurons, tanh activation function and absolute loss function. This is quite an improvement from the baseline naive Bayes classifier. In order to further improve the accuracy of network, number of training patterns should be increased.

6. Acknowledgment

We would like to thank Dr. Muhammad Zubair Shabbir from University of Veterinary and Animal Sciences, Lahore for providing us the dataset for analysis.

7. References

[1] Guidelines on Tularaemia, World Health Organization (WHO), 2007.

[2] D. Dennis., T. Inglesby, D. A. Henderson, J. G. Bartlett, M. S. Ascher, E. Eitzen, D. Anne.. "Tularemia as a Biological Weapon: Medical and Public Health Management." Jama vol. 285, no. 21, 2001, pp. 2763-2773.

https://doi.org/10.1001/jama.285.21.2763

[3] F. R. McCrumb Jr "Aerosol Infection of Man with Pasteurella Tularensis." Bacteriological Reviews, vol. 25, no. 3, 1961, pp. 262.

[4] Metzker, L. Michael, and C. T. Caskey. "Polymerase Chain Reaction (PCR)." eLS, 2009.

[5] P.V. Hornbeck, "Enzyme-linked Immunosorbent Assays." Current Protocols in Immunology, 1991, pp. 2-1.

[6] Whitehouse, M. CRAIG, R. N. Dreyer, M. Yamashita, and J. B. Fenn. "Electrospray Ionization for Mass-spectrometry of Large Biomolecules." Science, vol. 246, no. 4926, 1989, pp. 64-71.

https://doi.org/10.1126/science.2675315

[7] S. Dhar and V. Cherkassky. "Application of SOM to Analysis of Minnesota Soil Survey Data." In Neural Networks (IJCNN), The 2011 International Joint Conference on 2011, pp. 633-639.

https://doi.org/10.1109/IJCNN.2011.6033280

[8] Kumar, D. Ashok, and N. Kannathasan. "A Study and Characterization of Chemical Properties of Soil Surface Data using k-Means Algorithm." In Pattern Recognition, Informatics and Mobile Engineering (PRIME), 2013 International Conference on 2013, pp. 264-270.

[9] E. Hot and V. Popović-Bugarin. "Soil Data Clustering by using K-means and fuzzy K-means Algorithm." In Telecommunications Forum Telfor (TELFOR), 2015, pp. 890-893.

https://doi.org/10.1109/TELFOR.2015.7377608

[10] B. Bhattacharya and D. P. Solomatine. "Machine Learning in Soil Classification." Neural Networks, vol. 19, no. 2, 2006, pp. 186-195.

https://doi.org/10.1016/j.neunet.2006.01.005

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning Internal Representations by Error Propagation. No. ICS-8506. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

https://doi.org/10.21236/ADA164453

[12] G. D. Magoulas and M. N. Vrahatis. "Adaptive Algorithms for Neural Network Supervised Learning: A Deterministic Optimization Approach." International Journal of Bifurcation and Chaos, vol. 16, no. 07, 2006, pp. 1929-1950..

https://doi.org/10.1142/S0218127406015805