Performance Comparison of Machine Learning Models for Spoken Letters Recognition

Raidah S. Khudeyer, Zainab S. Jumaa, Noura H. Ajam

Abstract
Machine learning techniques have been successfully applied to problems in pattern classification, function approximations, optimization and pattern recognition. In this work, we employ different machine learning techniques: two types of neural network structures, Probability neural Network (PNN), Generalized Regression Neural Network (GRNN) and K-Nearest Neighbor Algorithm (KNN) and Support Vector Machine (SVM) to investigate the performance of these techniques on recognition of spoken letters. The data we used in this work taken from UCI machine learning repository. This paper shows the results of comparison among these techniques and the recognition results have indicated that the k-Nearest Neighbor gave good recognition performance of 100%.This work was implemented in MATLAB 7.0 environment.

Keywords: Nearest Neighbor Algorithm, Probabilistic Neural Network, Support Vector Machine, Generalized Regression Neural Network.

1 INTRODUCTION

Machine learning is a set of methods that can automatically detect patterns in data, and then use the uncovered patterns to predict future data, or to perform other kinds of decision making under uncertainty [1]. Machine learning is usually divided into two main types:

- supervised learning approach, the goal of this approach is to learn a mapping from input x to output y, given a labeled set of input-output pairs called a training set, each set represent vector of features or attributes(inputs) with C being the number of classes(outputs).
- Unsupervised learning approach, here only given inputs, and the goal is to find interesting patterns in the data [1].

Another type of machine learning is reinforcement learning. The training information provided by the environment is in the form of scalar reinforcement signal that constitutes a measure of how well the system operates. The learner is not told which actions to take, but rather must discover which actions yield the best rewards, by trying each action in turn [2].

In the area of supervised learning which deals with much classification, there are many types of algorithms [3]:

- Linear Classifier(Logical Regression, Native Bayes Classifier, Perceptron, Support Vector Machine)
- Quadratic Classifier
- K-Means Clustering
- Boosting
- Decision Tree
- Neural networks
- Bayesian Networks
- K-Nearest Neighbor

In this work we used four different machine learning techniques: two type of neural network, k- Nearest Neighbor and Support Vector Machine as linear classifier and neural networks.

KNN is a supervised learning algorithm for many applications for personalization tasks for medical diagnosis, credit scoring, pattern recognition and classification.

Also SVM is supervised machine learning technique used in medical diagnosis, classification and recognition. SVM model is a representation of examples as points in a space, mapped so that the examples of the separates categories are divided by a clear gap that is wide as possible [1]. The goal of SVM is to produce a model (based on training data) which predicts the target value of the test data given only the test data features [2].

A GRNN and PNN are an implementation of a statistical algorithm called kernel discriminate analysis in which the operations are organized into a multilayered feed forward network with four layers: Input layer, Pattern layer, Summation layer and Output layer, it uses a supervised training set to develop probability density functions within a pattern layer . This is a model based on competitive learning with a winner takes all attitudes and the core concept based on multivariate probability estimation.

This paper is structured as follows: section 2 is briefly explains concepts of k Nearest Neighbor Algorithm. Section 3 explains Support vector Machine technique while the section 4 is describe the Probabilistic Neural Network and Generalized Regression Neural Network techniques. Dataset description is described in section 5, while the work description and results are discussed in section 6 and section 7 shows the conclusions.

2 K- NEAREST NEIGHBOR ALGORITHM (KNN)

KNN is a simple algorithm which stores all available cases and classifies new cases based on a similarity measure [4]. KNN algorithm has been used since 1970 in many applications such as statistical estimation, pattern recognition, etc.

This algorithm depends on the choice of K training instances which represent the number of neighbors chosen to assign the class to the new Instance and the distance functions.Fig1. illustrates the behavior of KNN.

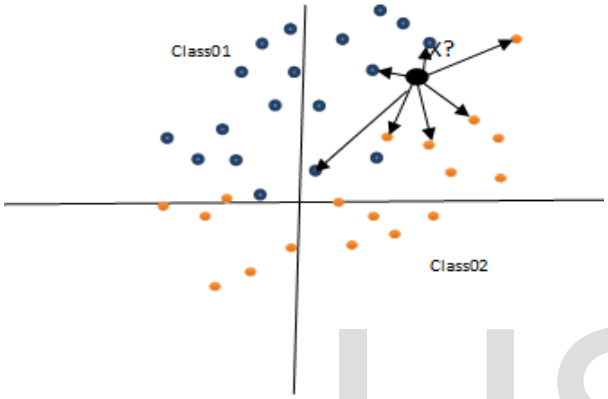

Fig1. KNN Structure

KNN Algorithm:

step1: Choose a value for the parameter K.

step2: input training data set: (X1, Y1), (X2, Y2)… (Xn,Yn), where:
Xi = (Xi(1) , Xi(2) , …,Xi(d)) is a d-dimensional feature vector of real numbers ,for all i.
Yi = [Y1……..Yc], for all I,

step3: Determine the K nearest neighbor of Xj to Xnew by calculating the distance functions.

step4: Classify by YKNN = Yj

In mathematics, a distance is an application that formalizes the idea of the distance which is the length between two points. The distance will allow us to group the individuals which are similar and separate those that do not resemble. The distances between two points can be defined in many ways [5] as follows:

Cityblock distance

Called also Manhattan distance, this distance is associated to the 1- norm, for two vectors Xi and Xj the Manhattan distance is defined:

$$d(Xi , Xj) = \sum_{i=1}^{n} |X - X| \qquad .....(1)$$

Euclidean distance

Euclidean distance is the most universal between two vectors Xir and Xjr, the Euclidean distance is defined as follow:

$$d(Xi,Xj)= \sqrt{\sum_{i=1}^{n}(x_{ir} - x_{jr})^2} \qquad .....(2)$$

Minkowski distance

The most frequently are the Minkowski distance which is defined as follows:

$$d(Xi , Xj) = \sqrt[p]{\sum_{i=1}^{n} |x_{ir} - x_{jr}|^p} \qquad .....(3)$$

3 SUPPORT VECTOR MACHINE (SVM)

SVM model is a representation of examples as points in a space, mapped so that the examples of the separates categories are divided by a clear gap that is wide as possible [6].

Suppose some given data points each belong to one of two classes, and the goal is to decide which class a new data point will be in. An SVM finds best hyper plane that separates all data points of one class from those of other class. The best hyper plan means the one with the largest margin between the two classes and it is known as maximum-margin hyper plan. Margin means the maximal width of the slap parallel to the hyper plane that has no interior data points. The support vectors are the data points that are closest to the separating hyper plan; these points are on the boundary of the slap. The assumption is made that the larger the margin the lower the risk of misclassifying samples of the test dataset and since it is simultaneously minimize the empirical classification error and maximize the geometric margin, so svm is called maximum margin classifier[8,9]. Fig2 . shows the Structure of SVM.

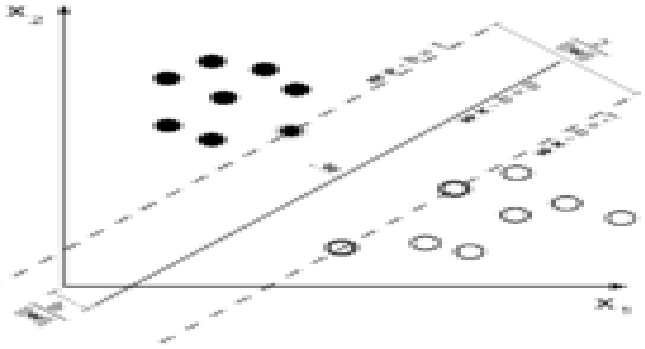

Fig 2. SVM structure

We consider labeled training data set as data points of the form:
{(x1, y1), (x2, y2), (x3, y3) ... (xn, yn)}
Where yn = 1/−1 A constant that denotes the class to which that point xn belongs. n = number of data samples. Each xn is a p-dimensional real vector. The scaling is important to guard against variable (attributes) with large

variance. We can view this training data , by means of dividing(or separation) hyper plan , which takes :

$$w.x + b = 0 \quad \ldots..(4)$$

Where bscalar and w is p-dimensional vector. The vector w points perpendicular to the separating hyper plan. The offset parameter b allows us to increase the margin. Since our interest is to find the "maximum-margin hyper plan "that divides the points having yi = 1 from those having yi = −1, we can select two parallel hyper planes if the training data linearly separable. The distance between these hyper plans should be large as possible.

These hyper planes can be described by the equations:

$$w.x - b = 1 \quad , \quad w.x - b = -1$$

Geometrically, the distance between these two hyper planes is $\frac{2}{||w||}$ so to maximize the distance between the plans we want to minimize $||w||$. Also we have to prevent data points from falling into the margin, we need to insure that for all i either

$$w.xi - b \geq 1 \qquad \text{if } yi = 1$$
$$\text{or} \quad w.xi - b \leq 1 \qquad \text{if } yi = -1$$

This can be rewritten as :

$$yi\,(w.x - b) \geq 1 \qquad \text{for all} \qquad 1 \leq i \leq n \quad \ldots\ldots(5)$$

SVM can perform non-linear classification by using what is called "kernel trick", implicitly mapping their inputs into high – dimensional feature space. It allows constructing the classifier without explicitly knowing the feature space. $(xi, xj) \equiv \emptyset(xi)^T \emptyset(xj)$ is called kernel function. There are many kernel functions in svm and choosing a good one is an important step to train an svm . It mappes the training vectors xi into a higher (may be infinite) dimensional space by the function $\emptyset$ [10]. Then the SVM finds a linear separating hyper plan with the maximal margin in this higher dimension space . Some popular kernel functions listed below: [7]

Linear kernel: $k(xi, xj) = xi^T xj$

Polynomial kernel: $k(xi, xj) = (\gamma xi^T xj + r)^d$, $\gamma > 0$

Radial Basic kernel: $k(xi, xj) = \exp\left(-\gamma ||xi - xj||^2\right)$, $\gamma > 0$

Sigmoid kernel: $k(xi, xj) = \tanh(\gamma xi^T xj + r)$

Here γ, r, d are kernel parameters.

4 GENERALIZED REGRESSIONS NEURAL NETWORKS (GRNN) / PROBABILISTIC NEURAL NETWORK (PNN)

GRNN and PNN networks are a type of supervised learning model based on Radial basis function (RBF) which can be used for regression, classification, and time series prediction [11].One of the property of GRNN ,it does not require an iterative training procedure. A GRNN can be viewed as a standardized RBF network in which there is a hidden unit centered at every training case. These RBF units are known as kernels and are usually probability density functions. The hidden to output weights are the target values, so the output is a weighted average of the target values of training cases close to the given input case [13, 16]. PNN is a special case of Radial Basis network (RBN), but it does not carry weights in its hidden layer. Each node of hidden layer acts as weights of an example vector. The hidden node activation is defined as the product of example vector and input feature vector [11]. GRNN and PNN have similar architecture except there is a fundamental difference. General regression neural networks perform regression where the target variable is continuous, whereas Probabilistic networks perform classification where the target variable is categorical [17].Fig3. Illustrates the architecture diagram. All GRNN / PNN have four layers: input layer, hidden layer, Pattern layer / Summation layer, decision layer. These layers are described in [12].

Spread selection of radial basis function is an important issue, the larger the spread, the smoother the function approximation. To fit data very closely, use a spread smaller than the typical distance between input vectors. To fit the data more smoothly, use a larger spread. PNN requires us to specify a value for spread of radial basis functions. Generally we test different values from 0-1, to determine the best value of spread.

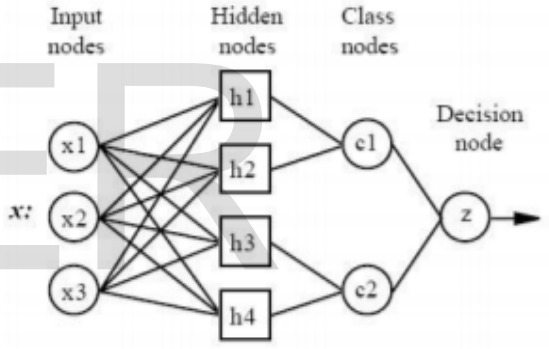

Fig3. GRNN / PNN architecture

5 DATA SET DESCRIPTION

This data set was consists of 150 subjects spoke the name of each letter of the alphabet twice. Hence, we have 52 training set from each speaker. The speakers are grouped into sets of 30 speakers each. Then the number of training set equal to 6238 while the number of test set is equal to 1559 sets.

Each data set has 617 features represent the inputs for each type of techniques that are used in this work. These features are described in the paper by Cole and Fanty[14]. The number of output is equal 1 class take values as: 1 for letter A , 2 for letter B, ... 26 for letter Z [15].

6 WORK DESCRIPTION AND RESULTS

This work described in the following points and fig.4 :

1- Prepare training - testing data: which consist of 6238 patterns for training and 1559 patterns for testing. Each pattern consists of 617 features.

2- Apply and record the results on four techniques:

In PNN and GRNN neural network: The choice of the spread value is very important. When spread value is small only a few samples plays a role, then the performance is less while if it is large, the performance will be increased as shown in the table(1) and fig5.

3- In KNN: We used Minkowski distance to find the distance between two point and kNN requires us to specify a value for k, generally for more accuracy we try with different values of k, after many experiments we find that when we increase k increase accuracy but the best k is 19.

4- In SVM: generate Fuzzy Inference System structure from data using FCM Clustering; the number of clusters is selected randomly.

5-In order to evaluate our methods, the performance (testing rate) is calculated as follows:

Testing rate= (correct patterns/1559)*100

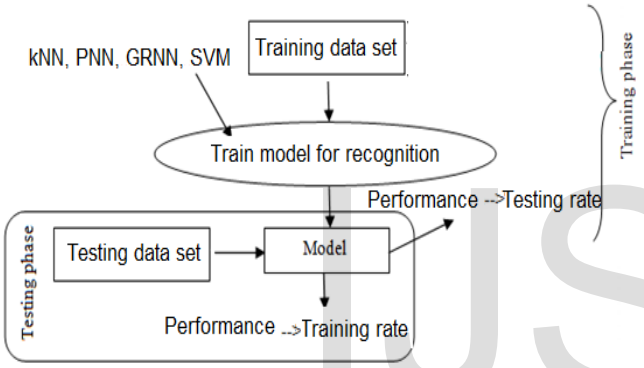

Fig4. Description diagram

The simulation process of the systems: SVM, PNN and GRNN techniques were carried out using MATLAB (R2008a), while kNN technique was programming in Matlab language.

After train PNN and GRNN, table (1) and fig5 shows the results for different values of spread.

Table (1) Performance (Testing rate) with different values of spread in case GRNN and PNN

Spread value	Testing rate %	
	GRNN	PNN
0.1	0.0	3.85
0.2	0.0	20.59
0.3	0.0	87.04
0.4	0.0	88.58
0.5	0.06	88.58
0.6	6.41	88.65
0.7	31.62	88.71
0.8	62.80	88.84
0.9	78.45	88.97
1.0	81.65	89.03

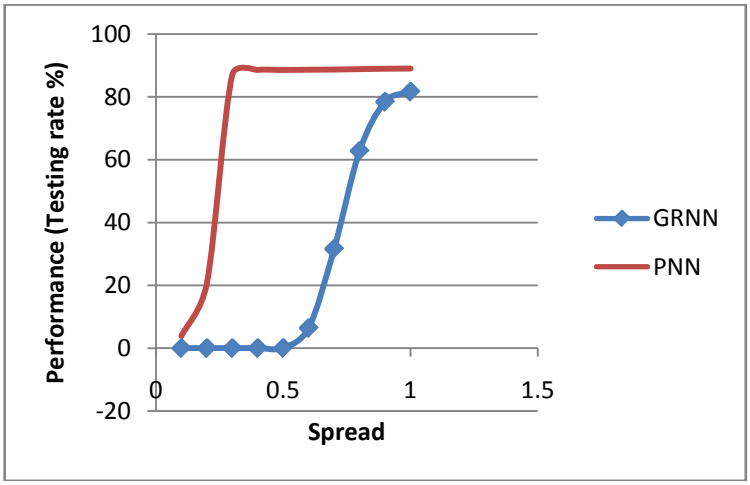

Fig5. Comparison performance with different values of spread in case GRNN and PNN

After train KNN and SVM, Table (2) and fig 6 shows the result of training and testing rate.

Table (2) Performance of kNN and SVM techniques

Tech. Name	kNN	SVM
Training rate	100%	100%
Testing rate	100%	93.01%

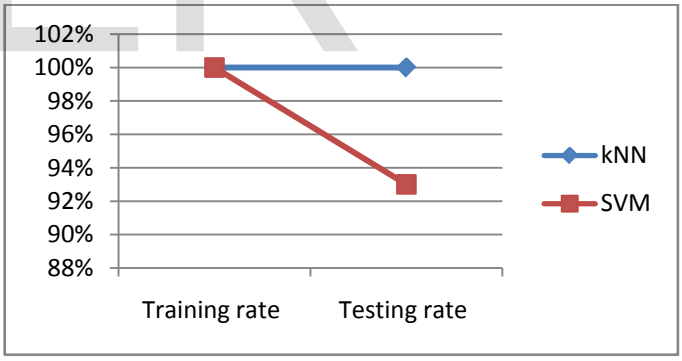

Fig 7. Training and testing rate in case KNN and SVM

From Table (3), the results show that KNN technique give high recognition comparing with other techniques.

Table (3) demonstrates the comparative results (testing rate) of four different techniques used for Spoken letter recognition

Tech. Name	kNN	SVM	PNN	GRNN
Testing rate	100%	93.01%	89.03	81.65%

7 CONCLUSIONS

Machine learning models can be used in recognition and classification problems; in this work we used many different techniques: GRNN, SVM, kNN and PNN for spoken letter recognition system. The recognition accuracy for trained and testing rate are recorded each techniques, The results show that kNN has better accuracy comparing to GRNN, PNN and SVM because it is able to classify all spoken letter. This illustrates that kNN acts more successfully in recognition.

REFERENCES

[1] Kevin P. Murphy, Machine Learning a probabilistic perspective, The MIT Press . London, England.

[2] S.B., Supervised machine learning : A review of classification techniques, Kotsiantis, 31(2007) P-249.

[3] Taivo O. Ayodele,"New Advance in Machine Learning", http// Intecopen.com/books/new-advances-in-machine-learning,university of Portsmouth,UK .

[4] Saed Sayad , "K Nearest Neighbors" , University of Toronto", http//chem.eng.utoronto/"datamining/ 2010.

[5] S.A. Medjahed , T.A. Saadi,A.Benyettou,"Brest Cancer Diagnosis by using K-Nearest Neighbor with Different Distances and Classification Rukes" ,International Journal of Computer Applications (0975-8887), Vol. 62, No. 1, January 2013.

[6] V.Anuja Kumarl et.al. , Classification of Diabetes Disease using Support vector machine , IJERA, V. 3, Issue 2, 2013 P- 1798.

[7] Chih. Wei Hsu et.al. , A practical Guide to Support vector Classification, Department of Computer Science, National Taiwan University, Taipei, Taiwan.

[8] Alexey Nefedov, Support vector machine,A simple tutorial.

[9] Durgesh K. et. al. , Data Classification using Support Vector machine , JTAIT.

[10] Piyush Rai, Kernel methods and nonlinear classification, CS5350/6350, Machine learning, 2011.

[11] Namrata Bilurkar, Ajay Prakash , GRNN and PNN Models for Cancer Prognosis and Prediction, Computational Systems for Health & Sustainability, 17-18, April, 2015 - by R.V.College of Engineering, Bangalore,Karnataka,PIN-560059,INDIA.

[12] R.Bharathi and Dr.Divya Satish, IMPROVING MRI BRAIN TUMOR CLASSIFICATION AND SEGMENTATION BASED ON COMPARISON TECHNIQUE, IJETCSE , ISSN: 0976-1353 Volume 13 Issue 1 –MARCH 2015.

[13] S. Swathi, S. Rizwana, G. Anjan Babu, P. Santhosh Kumar and P.V.G.K. Sarma, CLASSIFICATION OF NEURAL NETWORK STRUCTURES FOR BREAST CANCER DIAGNOSIS, International Journal of Computer Science and Communication, Vol. 3, No. 1, January-June 2012, pp. 227-231.

[14] Ron Cole and Mark Fanty, Department of Computer Science and Engineering, Oregon Graduate Institute, Beaverton, OR 97006. cole@cse.ogi.edu, fanty@cse.ogi.edu

[15] http://www.ics.uci.edu/pub/ml-repos/ machine- learning-databases/, 2003.

[16] Essam Al-Daoud , 2009. A Comparison Between Three Neural Network Models for Classification Problems. Journal of Artificial Intelligence, 2: 56-64.

[17] Shreepad S. Sawant, Preeti S. Topannavar, Introduction to Probabilistic Neural Network – Used For Image Classifications, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 5, Issue 4, 2015