Data Mining Techniques for Text Mining

[1]Mr. N.Senthil Vel Murugan, [2]Dr. V.Vallinayagam, [3]Dr. K. Senthamarai Kannan
[1]Department of Mathematics, ROHINI College of Engineering & Technology, Anjugramam
[2]Department of Mathematics, St.Joseph's College of Engineering, Chennai – 119.
[3]Department of Statistics, Manonmaniam Sundaranar University, Tirunelveli.
[1] senthil_msu@yahoo.com , [2] vngam05@yahoo.co.in , [3]senkannan2002@gmail.com,

Abstract :-Text mining is a variation on a field called data mining that tries to find interesting patterns from large databases. Classification is a major data mining task. It is often referred to as supervised learning because the classes are determined before examining the data. This research work deals with several classifiers including k-Nearest Neighbor (k-NN), Radial Basis Function (RBF), Multilayer Perceptron (MLP), and Support Vector Machine (SVM) which are used as trained classifiers for performing classification of data into relevant and non-relevant data. This study intends to compare the efficiency of the various existing classification algorithms with the proposed classification algorithms on the basis of runtime, error rate and accuracy. The aim of this paper is the classification algorithms are applied to classify the intrusion detection data sets like Signature Verification.

Keywords: *k-NN, RBF, MLP, SVM, Comparative Cross Validation*

*****

I. INTRODUCTION

Data mining can help reduce information overload and improve decision making. This is achieved by extracting and refining useful knowledge through a process of searching for relationships and patterns from the extensive data collected by organizations. The extracted information is used to predict, classify, model, and summarize the data being mined. Data mining technologies, such as rule induction, neural networks, genetic algorithms, fuzzy logic and rough sets are used for classification and pattern recognition in many industries. With the advancement and expansion of data mining, there is a large scope and need of an area which can serve the purpose of various domains. Fusion of techniques from data mining, language, information process retrieval and visual understanding created an interdisciplinary field called text mining.

Text data mining, referred to as text mining is a process of extracting the information from an unstructured text. In order to obtain high text information, a process of pattern division and trends is done. For an efficient text mining system, the unstructured text is parsed and attached or removed some level of linguistic feature, thus making it structured text. A standard text mining approach will involve categorization of text, text clustering, and extraction of concepts, granular taxonomies production, sentiment analysis, document summarization and modeling. Text mining involves a two stage processing of text. In the first step a description of document and its content is done. This process is called categorization process. In the second step called as classification, the document is divided into descriptive categories and an inter document relationship is established. Text mining has been useful in many areas, i.e. security applications, software applications, academic applications etc.

k-nearest neighbor is a supervised learning algorithm where the result of new instance query is classified based on majority of k-nearest neighbor category. The purpose of this algorithm is to classify a new object based on attributes and training samples.

A radial function or a radial basis function (RBF) is a class of function whose value decreases (or increases) with the distance from a central point. An RBF has a Gaussian shape, and an RBF network is typically a Neural Network with three layers. The input layer is used to simply input the data. The Gaussian activation function is used at the hidden layer, while a linear activation function is used at the output layer. The objective is to have the hidden nodes learn to respond only to a subset of the input, namely, that where the Gaussian function is entered. This is usually accomplished via supervised learning.

The simple feed forward Neural Network that is, actually called a multilayer perceptron. An MLP is a network of perceptions and used for classifying the height. The neurons are placed in layers with outputs always flowing toward the output layer. If only one layer exists, it is called a perceptron. If multiple layers exist, it is an MLP.

The support vector machine (SVM) is a training algorithm for learning classification and regression rules from data, for example the SVM can be used to learn polynomial, radial basis function (RBF) and multi-layer perceptron (MLP) classifiers (Osuna, *et al.*, 1997). SVM can be applied for classification and regression problems. Classification algorithms are increasingly being used for problem solving. The efficiency of algorithms has been compared on the basis of runtime, error rate, accuracy using Weka machine learning tool.

Holdout, random sub sampling, cross-validation and bootstrap are common techniques for accessing accuracy based on randomly sampled partitions of the given data. The use of such techniques to estimate accuracy increase the overall computation time yet is useful for model selection. Apart from these techniques in this case, a new technique, "comparative cross validation" is proposed which involves accuracy estimation by either stratified k-fold cross-validation or equivalent repeated random subsampling. The goal is to calculate the expectation of the classification accuracy, as given by either Stratified k-fold cross-validation or repeated random sub sampling (Jiawei Han, Micheline Kamber 2003).

II. REVIEW OF LITERATURE

Many researchers have investigated the technique of combining the predictions of multiple classifiers to produce a single classifier. The resulting classifier is generally more accurate than any of the individual classifiers making up the ensemble. Fang b, et al., (2003) proposed, two methods to track the variations in the signature patterns written by the same person. The variations can occur in the shape or in the relative positions of the characteristic features. Given the set of training signature samples, the first method measures the positional variations of the one-dimensional projection profiles of the signature patterns; and the second method determines the variations in relative stroke positions in the two-dimension signature patterns.

Songbo Tana (2006) proposed a new refinement strategy, which is called as Drag Pushing, for the KNN Classifier. The experiments on three benchmark evaluation collections show that Drag Pushing achieved a significant improvement on the performance of the KNN Classifier. Sandhya Peddabachigari, Ajith Abraham, Crina Grosan, Johnson Thomas (2007) presents two hybrid approaches for modeling IDS. Decision trees (DT) and support vector machines (SVM) are combined as a hierarchical hybrid intelligent system model (DT–SVM) and an ensemble approach combining the base classifiers and it is concluded that the proposed hybrid systems provide more accurate intrusion detection systems.

Rachid Beghdad (2008) present a critical study about the use of some neural networks (NNs) to detect and classify intrusions. The aim of research is to determine which NN classifies well the attacks and leads to the higher detection rate of each attack. This study focused on two classification types of records: a single class (normal, or attack), and a multiclass, where the category of attack is also detected by the NN. Five different types of NNs were tested: multilayer perceptron (MLP), generalized feed forward (GFF), radial basis function (RBF), self-organizing feature map (SOFM), and principal component analysis (PCA) NN. In the single class case, the PCA NN performs the higher detection rate.

III. DATABASE

Data collection plays an important role in the data mining problems. In this paper, the dataset used for online signature verification is obtained from UCI repository of machine learning databases. It has been examined and reprocessed by http://www.ics.uci.edu/~mlearn/MLRepository.html.

IV. PROPOSED PROCEDURES

4.1 Signature Verification

The most commonly used protection mechanisms today are based on either what a person possesses (e.g. an ID card) or what the person remembers (like passwords and PIN numbers). However, there is always a risk of passwords being cracked by unauthenticated users and ID cards being stolen, in addition to shortcomings like forgotten passwords and lost ID cards (Huang & Yan, 1997). To avoid such inconveniences, one may opt for the new methodology of Biometrics, which though expensive will be almost infallible as it uses some unique physiological and/or behavioral (Huang & Yan,1997) characteristics possessed by an individual for identity verification. Examples include signature, iris, face, and fingerprint recognition based systems.

Generally online signature verification methods display high accuracy rates (closer to 99%) than off-line methods (90-95%) in the case of all the forgeries. This is because in off-line verification methods, the forger has to copy only the shape (Jain & Griess, 2000) of the signature. On the other hand , in the case of online verification methods, since the hardware used captures the dynamic features of the signature as well, the forger has to not only copy the shape of the signature but also the temporal characteristics (pen tilt, pressure applied, signing velocity etc.) of the person whose signature is to be forged. In addition, he has to simultaneously hide his own inherent style of writing the signature, thus making it extremely difficult to deceive the device in the case of online signature verification.

The online verification system can be classified into the following modules:

- Data Acquisition
- Preprocessing and Noise Removal
- Feature Extraction
- Verification (or Identification)

Data Acquisition

Data acquisition in online verification methods is generally carried out using special devices called transducers or digitizers (Tappert, Suen, & Wakahara, 1990, Wessels & Omlin, 2000), in contrast to the use of high resolution scanners in case of off-line. The commonly used instruments include the electronic tablets, pressure sensitive tablets, digitizers involving technologies such as acoustic sensing in air medium, Surface acoustic waves, triangularization of reflected laser beams, and optical sensing of alight pen to extract information about the number of strokes, velocity of signing, direction of writing, pen tilt, pressure with which the signature is written etc.

Preprocessing

Preprocessing in online is much more difficult than in off-line, because it involves both noise removal (Plamondon & Lorette, 1989) and segmentation in most of the cases. The other preprocessing steps that can be performed are signal amplifying, filtering, conditioning, digitizing, resampling, signal truncation, normalization, etc. However, the most commonly used include:

- **External Segmentation**: Tappert, Suen and Wakahara (1990) define external segmentation as the process by which the characters or words of a signature are isolated before the recognition is carried out.
- **Resampling:** This process is basically done to ensure uniform smoothing to get rid of the redundant information, as well as to preserve the required information for verification by comparing the spatial data of two signatures. According to Jain and Griess (2000), here, the distance between two critical points is measured, and if the total distance exceeds a threshold called the resampling length (which is calculated by dividing the distance by the number of sample points for that segment), then a new point is created by using the gradient between the two points.
- **Noise Reduction:** Noise is nothing but irrelevant data, usually in the form of extra dots or pixels in images (in case of off-line verification methods) (Ismail & Gad, 2000), which do not belong to the signature, but are included in the image (in case of off-line) or in the signal (in case of online), because of possible hardware problems (Tappert, Suen, & Wakahara, 1990) or presence of background noises like dirt or by faulty hand movements(Tappert, Suen, & Wakahara, 1990) while signing.

Feature Extraction

Online signature extracts both the static and the dynamic features. Some of the static and the dynamic features have been listed below.

- **Static Features:** Although both static and dynamic information are available to the online verification system, in most of the cases, the static information is discarded, as dynamic features are quite rich and they alone give high accuracy rates.
- **Dynamic features**: Though online methods utilize some of the static features, they give more emphasis to the dynamic features, since these features are more difficult to imitate.

4.2 k-Nearest Neighbor Classifier

k-nearest neighbor (Margaret H.Dunham**,** 2003) is a supervised learning algorithm where the result of new instance query is classified based on majority of k-nearest neighbor category. The purpose of this algorithm is to classify a new object based on attributes and training samples. The classifiers do not use any model to fit and only based on memory. Given a query point, k number of objects (k=1) are found closest to the query point. The classification is using majority vote among the classification of the k objects. Any ties can be broken at random. k-Nearest neighbor algorithm used neighborhood classification as the prediction value of the new query instance.

4.3 Radial Basis Function

Radial basis function (RBF) networks combine a number of different concepts from approximation theory, clustering, and neural network theory. A key advantage of RBF networks for practitioners is the clear and understandable interpretation of the functionality of basis functions. Also, fuzzy rules may be extracted from RBF networks for deployment in an expert system.

4.4 Multilayer Perceptron

The simplest neural network is called a perceptron. A perceptron is a single neuron with multiple inputs and one output. The original perceptron proposed the use of a step activation function, but it is more common to see another type of function such as a sigmoidal function. A simple perceptron can be used to classify into two classes. Using a unipolar activation function, an output of 1 would be used to classify into one class, while an output of 0 would be used to pass in the other class.

An MLP is a network of perceptrons. The neurons are placed in layers with outputs always flowing toward the output layer. If only layer exists, it is called a perceptron. If multiple layers exist, it is an MLP. Although the backpropagation algorithm can be used very generally to train neural networks, it is most famous for applications to layered feedforward networks, or multilayer perceptrons.

Multilayer perceptrons with L layers of synaptic connections and L + 1 layers of neurons are considered. This is sometimes called an L-layer network, and sometimes an L + 1-layer network. A network with a single layer can approximate any function, if the hidden layer is large enough. This has been proved by a number of people, generally using the Stone-Weierstrass theorem. So, multilayer perceptrons are representational powerful.

Let's diagram the network as

$$x^0 \xrightarrow{w^1,b^1} x^1 \xrightarrow{w^2 b^2} \ldots. \xrightarrow{w^L,b^L} x^L$$

where $x^l \in R^{n_1}$ for all $l = 0.....,L$ and W^l is an $n_l x n_{l-1}$ matrix for all $l = 1,.....,L$

There are L+1 layers of neurons, and L layers of synaptic weights. It is supposed to change the weights W and biases b so that the actual output x^L becomes closer to the desired output d.

The backpropagation algorithm consists of the following steps.

1. Forward pass. The input vector x^0 is transformed into the output vector x^L, by evaluating the equation

$$x_i^l = f(u_i^l) = f(\sum_{j=1}^{n_{l-1}} W_{ij}^l x_j^{l-1} + b_i^l) \qquad [4.1]$$

for l= 1 to L

2. Error computation. The difference between the desired output d and actual output x^L is computed.

$$\delta_i^L = f'(u_i^L)(d_i - x_i^L) \qquad [4.2]$$

3. Backward pass. The error signal at the output units is Propagated backwards through the entire network, by evaluating

$$\delta_j^{l-1} = f'(u_j^{l-1}) \sum_{i=1}^{n_l} \delta_i^l W_{ij}^l \qquad [4.3]$$

from l = L to 1.

4. Learning updates. The synaptic weights and biases are updated using the results of the forward and backward passes,

$$\Delta W_{ij}^l = \eta \delta_i^l x_j^{l-1} \qquad [4.4]$$

$$\Delta b_i^l = \eta \delta_i^l \qquad [4.5]$$

These are evaluated for l = 1 to L. The order of evaluation doesn't matter.

4.5 Support Vector Machine

SVM were first suggested by Vapnik in the 1960s for classification and have recently become an area of intense research owing to developments in the techniques and theory coupled with extensions to regression and density estimation. SVM deliver the state of art performance in real world applications such as text categorization, hand-written character recognition, image classification, financial forecasting and so on (Bao, 2003). The support vector machine (SVM) is a training algorithm for learning classification and regression rules from data, for example the SVM can be used to learn polynomial, radial basis function (RBF) and multi-layer perceptron (MLP) classifiers (Osuna, *et al.*, 1997). Support vector machine is a new machine-learning paradigm that works by finding an optimal hyperplane as to solve the learning problems.

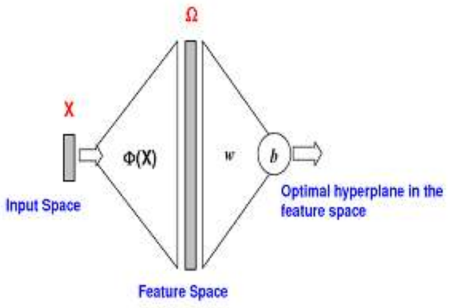

Figure 4.3: Support Vector Machine

4.6 Comparative Cross Validation

Comparative cross validation technique is proposed which involves accuracy estimation by either stratified k-fold cross-validation or equivalent repeated random sub sampling. As per cross validation initial dataset (S) is divided into parts - training [Str] and test [Stst]. Subsequently, k-fold cross validation should divide data [Str] into a secondary training set [(k-1) folds] and a validation set [1 fold]. After training with cross validation, the overall accuracy for Str was always significantly higher than that of Stst. By increasing the size of the Str dataset so that it is more representative of the dataset as a whole (S). That is increasing the number of training vectors, Much more similar training / test accuracy results can be obtained.

The goal is to calculate the expectation of the accuracy, as given by either Stratified k-fold cross-validation or repeated random subsampling (Jiawei Han, Micheline Kamber 2003). The accuracy obtained using Stratified k-fold cross-validation or repeated random sub sampling where $|S|T| = N/K_S$

N - Size of S (|S|),	c(x) - The class label associated with x
C - Number of class labels in S	Ni - Number of elements in class i.
Ni = |fx : c(x) = {i}|	k - Number of folds in k-fold cross validation (CV).

Let D = $(d_1, d_2 \ldots d_{ks})$ be a partition of S for Stratified k-fold cross-validation

The error rate is calculated using mean square error (MSE) evaluated by comparative cross validation. The formula for MSE is as follows:

$$MSE = \frac{\sum_{i=1}^{n}(a_i - p_i)^2}{n}$$, Where: a_i = actual output at time I

a_i = actual output at time I p_i = predicted output at time I n = number of data.

4.7 Validation of the Methods

Natural performance measures for classification problems:

- Run Time: Training Time
- Success: instance’s class is predicted correctly
- Error: instance’s class is predicted incorrectly
- Error rate: proportion of errors made over the whole set of instances
- Accuracy: proportion of correctly classified instances over the whole set of instances

4.8 Experimental Results

Weka is an open source data mining software that contains java implementations of many popular machine learning-algorithms including some popular classification algorithms. The algorithms require the data to be in specific formats.

Table 4.1 : Signature Verification Dataset

Signature Verification	Instances	Attributes
Bob – alive	122	12 2
Bob – changing (mind)	84	12
Alies – alive	99	12 2
Alies – Changing (mind)	127	12

Table 4.2: Parameters of existing and proposed k-NN classifiers

Signature Verification	Existing k-NN			Proposed k-NN		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	0.40	0.00	100	0.20	0.0003	99.99
Bob – changing (mind)	0.37	0.00	100	0.07	0.0004	99.99
Alies – alive	0.50	0.00	100	0.24	0.0003	99.99
Alies – Changing (mind)	0.25	0.00	100	0.15	0.0001	99.99

Table 4.3: Parameters of Proposed Bagged and Pruned Bagged k-NN Classifiers

Signature Verification	Proposed Bagged k-NN			Proposed Pruned Bagged k-NN		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	0.11	0.00	100	0.02	0.00	100
Bob – changing (mind)	0.6	0.00	100	0.2	0.00	100
Alies – alive	0.7	0.00	100	0.2	0.00	100
Alies – Changing (mind)	0.6	0.00	100	0.1	0.00	100

Table 4.4: Parameters of Existing and proposed RBF classifiers

Signature Verification	Existing RBF			Proposed RBF		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	1.62	0.00	100	0.61	0.0002	99.99
Bob – changing (mind)	0.31	0.00	100	0.09	0.0006	99.99
Alies – alive	0.29	0.00	100	0.12	0.0001	99.99
Alies – Changing (mind)	0.85	0.00	100	0.31	0.0002	99.99

Table 4.5: Parameters of Proposed Bagged and pruned Bagged RBF classifiers

Signature Verification	Proposed Bagged RBF			Proposed Pruned Bagged RBF		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	4.15	0.00	100	0.70	0.00	100
Bob – changing (mind)	0.81	0.0002	100	0.16	0.0001	100
Alies – alive	0.80	0.00	100	0.11	0.00	100
Alies – Changing (mind)	2.42	0.00	100	0.25	0.00	100

Table 4.6: Parameters of existing and proposed MLP classifiers

Signature Verification	Existing MLP			Proposed MLP		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	18.19	0.00	100	16.75	0.0003	99.99
Bob – changing (mind)	9.69	0.00	100	9.01	0.0005	99.99
Alies – alive	10.22	0.00	100	10.71	0.0002	99.99
Alies – Changing (mind)	17.18	0.00	100	16.86	0.0001	99.99

Table 4.7:Parameters of Proposed Bagged and pruned Bagged MLP classifiers

Signature Verification	Proposed Bagged MLP			Proposed Pruned Bagged MLP		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)

Bob – alive	178.39	0.0002	100	16.12	0.0001	100
Bob – changing (mind)	89.2	0.0003	100	28.78	0.0002	100
Alies – alive	102.5	0.0001	100	6.46	0.00	100
Alies – Changing (mind)	172.85	0.00	100	10.55	0.00	100

Table 4.8: Parameters of existing and proposed SVM classifiers

Signature Verification	Existing SVM classification			Proposed SVM classification		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	9.90	0.00	100	8.35	0.0009	99.99
Bob – changing (mind)	5.31	0.00	100	2.26	0.0005	99.99
Alies – alive	2.80	0.00	100	2.47	0.0002	99.99
Alies – Changing (mind)	9.59	0.00	100	8.19	0.0008	99.99

Table 4.9: Parameters of Proposed Bagged and pruned Bagged SVM classifiers

Signature Verification	Proposed Bagged SVM classification			Proposed Pruned Bagged SVM classification		
	Run Time (Seconds)	Error Rate (%)	Accuracy (%)	Run Time (Seconds)	Error Rate (%)	Accuracy (%)
Bob – alive	36.03	0.0008	100	2.14	0.0007	100
Bob – changing (mind)	21.06	0.0004	100	0.22	0.0004	100
Alies – alive	11.40	0.0001	100	0.47	0.00	100
Alies – Changing (mind)	66.66	0.0007	100	1.37	0.0006	100

V. CONCLUSION

The study has attempted to develop a new technique called comparative cross validation for data mining problems. The method evaluates the error rate, accuracy and run time for base classifiers. This research paper presents comprehensive empirical evaluation of four different approaches namely k-Nearest Neighbor, radial basis function, Multilayer perceptron, Support vector machine with Signature Verification. Weka data mining software is used to compare the various algorithms and the results have been reported.

SVM ensembles have verified the signature of two different persons as forged and genuine and the verification rate was found to be extremely high (around 99-100%) as expected with the online verification methods. The run time is also found to be significantly reduced.

VI. REFERENCES

[1] Berk. R. A. (2004) “Data Mining within a Regression Framework”, in Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers, oded Maimon and Lior Rokach (eds.), Kluwer Academic Publishers.

[2] Breiman, L. (1996c). Stacked Regressions, Machine Learning, 24(1):49-64.

[3] Breiman, L. (2000). Randomizing outputs to increase prediction accuracy. Machine Learning, 40(3): 229-242.

[4] Fang, b., Leung, c. H., Tang, y. Y., Tse, k. W., Kwok, p. c. k., Wong, y. k. (2003), “Off-line signature verification by the tracking of feature and stroke positions”, Pattern recognition 36:91-101.

[5] Fayyad, U., Piatetsy-Shapiro, G., Smyth, P., and Uthurusamy, R. (1996). From data mining to knowledge discovery. In Advances in Knowledge Discovery and Data Mining.

[6] Friedman, J. H. (1997). On bias, variance, 0/1 loss and the curse of dimensionality. Data Mining and Knowledge Discovery, 1:55–77.

[7] Hansen, L., and Salamon, P. (1990). Neural Network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12: 993-1001.

[8] Ian H.Witten and Eibe Frank, (2005). “Data Mining-Practical Machine Learning Tools and Techniques”, Elsevier, 177-178.

[9] Jiawei Han , Micheline Kamber, (2003). “ Data Mining – Concepts and Techniques” Elsevier, 359-366.

[10] Margaret H.Dunham, (2003), “Data Mining-Introductory and Advanced Topics”, Pearson Education, 90-113.

[11] Govindarajan, RM.Chandrasekaran, (2008) “Support vector Machine with Polynomial Kernel based on Intrusion Detection Data”, International Journal of Computer Science and System Analysis, 2(2),Pages 11-16.

[12] Govindarajan, RM.Chandrasekaran, (2008) “Optimal Design of Radial Basis Function for Intrusion Detection Data”, Asian Journal of Information Technology, 7 (11), pages 489-493.