International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 1, July 2014)

Text Classification Using PSO & Other Technique

Tanya Taneja[1], Balraj Sharma[2]
M. Tech. Scholar, BITS, BHIWANI, INDIA.
A.P. in CSE Department, BITS, BHIWANI, INDIA.

Abstract **- Categories for classification of text are predefined according to these categories all text data is classified. We require classifying text to manage and search any data in database. There are many techniques available in market to classifying the text. Now days every website has overloaded text in database as like customer support websites, news website etc.so in this type of websites text need to classify. In news websites it's necessary to maintain record of old and each news in database. The news can be classifying on the basis of predefined categories of type crime news, sports news, election news, entertainments etc. every technique that exists in real like SVM, Naïve Bayes, PSO and Neural classifiers, working well at a level with some limitations. In this paper we are going to discuss about these techniques and conclude with the comparison of results find out which technique can perform well.**

Keyword**- Text Classification, SVM, Naïve Bayes, Neural, News, Mining, PSO**

I. INTRODUCTION

Text classification is a supervised learning process. In this process a task is assign on text data or document for classify this text according to predefined categories or classes according to their contents. For a long time it is very classical problem in information access field, recently this field is attracted due to over loaded amount of text document available in digital form. Some systems are based on text classification like routing, data access, classification, and filtering. There are many numbers of documents available in digital form and day by day availability of documents increasing. These documents represent some information that can be access easily. So to access information from huge amount of documents is very difficult and more time consuming. Organizations that need to access information from huge amount need a technique to solve this difficulty and more work in less time. Data is automatically classified according categories of their contents. There are many algorithm are available to deal with automatic text classification [1].

So as we come to online data available in database, as time passes the amount of data in database increasing. Website is daily visited by internet users and that user's login on website.

To login on website each user firstly fills some details about him/her, the information regarding each user store in database. So as like this amount of data in database increasing in news websites. Daily news updated on website. We can also see old news of any day. So each news store in database. This database have huge amount of available, then the difficulty arise when data store in mix stage. Difficulty of this removes by using classification of data. Firstly news is classified on the day basis. A record of particular date store together. Due to which a user can choose particular date category and got data. After that news classified on types of news. So there can be categories like crime news, sports news, election news, entertainment news, govt. news etc. due to this type of categories it's easy for a user to search particular news. Like crime news available in crime news categories.

II. COMMON METHOD FOR TEXT CLASSIFICATION

Common method is followed by each technique for text classification is shown in fig. 1. After some common steps we apply algorithm which is suitable for us.

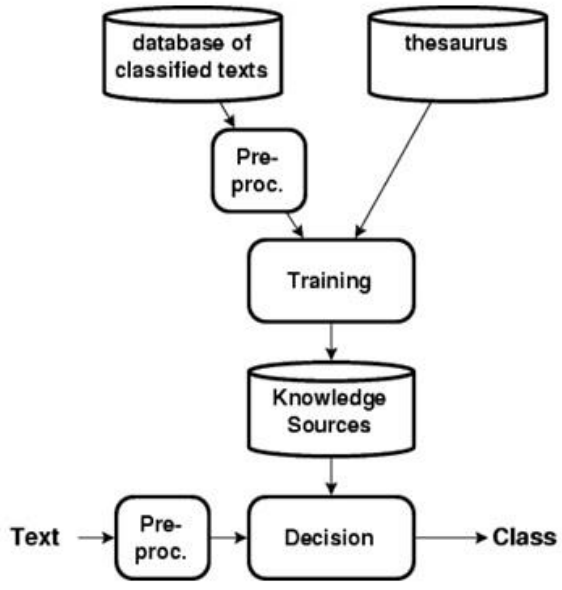

Fig 1 architecture of text classification

As shown in fig we need training dataset to start the process of text classification. The flowchart represents decision function where we apply the technique or algorithm to classify data. Before applying algorithm we

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 1, July 2014)

have to pre - process the training dataset. In pre - processing we remove HTML tags, stemming etc. size of input data can be reduced in pre – processing data. Like sentence boundary determination activities involve in it [2].In knowledge source we can identify important words from documents.

III. Evaluation Matrix

Rules for text classification can be evaluated by performance of data retrieval. There are some parameters on the basis we can measure the performance of text classification that are recall, accuracy, error rate, F_1, and precision. Consider a set of N documents, than apply our algorithm one by one then got a table with four cells. Count for true positive (TP), false positive (FP), true negative (TN) and false negative (FN) exist in cell of table respectively. Clearly, N = TP + FP + TN + FN. The metrics for binary-decisions are defined as:

- **Precision** = TP / (TP + FP)
- **Recall** = TP / (TP + FN)
- **Accuracy** = (TP + TN)/N
- **Error** = (FP + FN)/N
- **F1** = 2*Recall*Precision/(Recall + Precision)

Due to the often highly unbalance number of positive vs. negative examples, note that TN often dominates the accuracy and error of a system, leading to miss-interpretation of the results.

IV. Techniques For Text Classification

There are many techniques available in market for text classification like Association Rule, Naïve Bayes Classifier, SVM, and Genetic Algorithm. In this paper we are going to explain technique that is Support Vector Machine (SVM).

A. Support Vector Machine (SVM)

Support vector machines have strong theoretical foundations and excellentempirical successes. They have been applied to tasks such as handwritten digit recognition,object recognition, and text classification. First of all Support-vector machines proposed for numericaldata. Basic idea behind SVM is to determine separators in thesearch space which can best separate the different classes.

A hypothesis h find the structure for idea of minimization risk, with SVM we can expect lowest true error. Its probability for h will make an error on randomly and selected test example. An upper bound can be used to connect thetrue error of a hypothesis h with the error of h on the training set and thecomplexity of H (measured by VC-Dimension), the hypothesis space containingh [3].

SVM is known as a universal learner. Basically linear threshold value is learnt in SVM. Nevertheless, a simple function “plug - in” is used as kernel for learning. Main property of SVM is that it is independent of the dimensionality of the feature space. SVM separate the data not number of features. This is way to get margin in SVM learner. It means we can categories in the presence of features. By setting the parameters in good way we can also get margin. Like we set width of kernel.

Now the question arise that why SVM work well for text classification. There are some reasons behind this as following:

- *High dimensional input space:* When learning text classifiers, one has to face with very many (more than 1000) features. Since, over fitting protection use by SVM, which does not necessarily depend on the number of features, they have the potential to handle these large feature spaces.
- *Document vectors are sparse:* In document vector there only some values that are not zero, for each document. [4] Give both theoretical and empirical evidences for the mistake bound model that additive" algorithms, which have a similar inductive bias like SVMs, are well suited for problems with dense concepts and sparse instances.
- *Mostly text categorization problems are linearly separable:* Categories which are assumed are separate linearly. Main task of SVM is to find that separators which separate linearly.

SVM need two steps to complete process as like other supervised machine learning algorithms. In the first step — the training step — it learns a decision boundary in input space from pre classified training data. In the second step — the classification step — it classifies input vectors according to the previously learned decision boundary. There are two classes which value find by SVM and separate these classes that classes are — a positive class ($y = +1$) and a negative class ($y = -1$). By example we can understand well as shown in fig 2

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 1, July 2014)

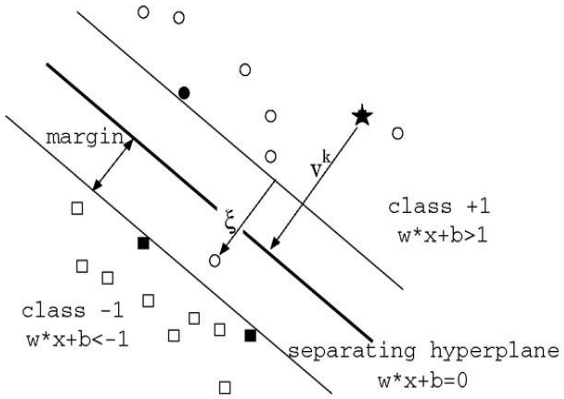

Fig 2 Operating mode of a Support Vector Machine

In the training step the following problem is solved: Given is a set of training examples S_l = $\{(x_1,y_1),(x_2,y_2),\ldots,(x_l,y_l)\}$ of size l from a fixed but unknown distribution $p(x,y)$ describing the learning task. The term-frequency vectors $x_i$represent documents and $y_i \in \{-1,+1\}$ show the indication of document labeled with positive class or not.Here the main target of SVM is find a decision rule h $\mathcal{L}$: x → {-1,+1} that classifies documents as accurately as possible based on the training set S_l.functions $f(x)$ = sign(wx + b) represents hypothesis space, where w and b are parameters that are learned in the training step and which determine the class separating hyper plane. Computing this hyper plane is equivalent to solving the following optimization problem [5] [6].

$$\text{minimize: } W(\alpha) = -\sum_{i=1}^{\ell}\alpha_i + \frac{1}{2}\sum_{i=1}^{\ell}\sum_{j=1}^{\ell} y_i y_j \alpha_i \alpha_j \mathbf{x}_i \mathbf{x}_j$$

$$\text{subject to: } \sum_{\substack{i=1 \\ 0\le\alpha_i\le C}}^{\ell} y_i\alpha_i = 0$$

All training examples with $\alpha_i > 0$ at the solution are called support vectors. The Support vectors are situated right at the margin (see the solid circle and squares in figure 2) and define the hyperplane. The definition of a hyperplane by the support vectors is especially advantageous in high dimensional feature spaces because a comparatively small number of parameters — the αs in the sum of equation (above) is required.

Advantage of the SVM method is that since it attempts to determine theoptimum direction of discrimination in the feature space by examiningthe appropriate combination of features, it is quite robust to high dimensionality.

B. Naive Bayes Technique For Text Classification

Naïve Bayes is used as text classifier because of its simplicity and effectiveness. Simple ("naive") classifica1onmethodbased on Bayesrule [7]. The Bayes rule is applied on document for the classification of text. The rule which is following is:

$$P(c \mid d) = \frac{P(d \mid c)P(c)}{P(d)}$$

This rule is applied for a document d and a class c. probability of A happening given to B can be find with the probability of B given to A. this algorithm work on the basis of likelihood in which probability of document B is same as frequency of words in A.on the basis of words collection and frequencies a category is represented. We can define frequency of word is number of time repetition in document define frequency of that word. We can assume n number of categories from C_0 to C_{n-1}. Determining which category a document D is most associated with means calculating the probability that document D is in category C_i, written $P(C_i|D)$, for each category C_i.

Using the Bayes Rule, you can calculate $P(C_i|D)$ by computing:

$P(C_i|D) = (P(D|C_i) * P(C_i)) / P(D)$

$P(C_i|D)$ is the probability that document D is in category C_i; in document D bag of words is given by probability, which create in category C_i. $P(D|C_i)$ is the probability that for a given category C_i, the words in D appear in that category.

$P(C_i)$ is the probability of a given category; that is, the probability of a document being in category C_i without considering its contents. P(D) is the probability of that specific document occurring. We can classify text with procedure that required using above discussed parameters is as following:

$$c_{MAP} = \underset{c\in C}{\operatorname{argmax}}\, P(c \mid d)$$ MAP is "maximum a posteriori" = most likely class

$$= \underset{c\in C}{\operatorname{argmax}}\, \frac{P(d \mid c)P(c)}{P(d)}$$ Bayes Rule

$$= \underset{c\in C}{\operatorname{argmax}}\, P(d \mid c)P(c)$$ Dropping the denominator

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 1, July 2014)

C. Neural Network Classifiers

The basic unit in a neural network is a *neuron* or *unit*. Each unitreceives a set of inputs, which are denoted by the vector *Xi*, which inthis case, correspond to the term frequencies in the *i*th document. Eachneuron is also associated with a set of weights *A*, which are used inorder to compute a function *f*(·) of its inputs. A typical function whichis often used in the neural network is the linear function as follows:

$$pi = A \cdot Xi \quad (6.14)$$

Thus, for a vector *Xi* drawn from a lexicon of *d* words, the weight vector*A* should also contain *d* elements. Now consider a binary classificationproblem, in which all labels are drawn from {+1, −1}. We assume thatthe class label of *Xi* is denoted by *yi*. In that case, the sign of thepredicted function *pi* yields the class label.

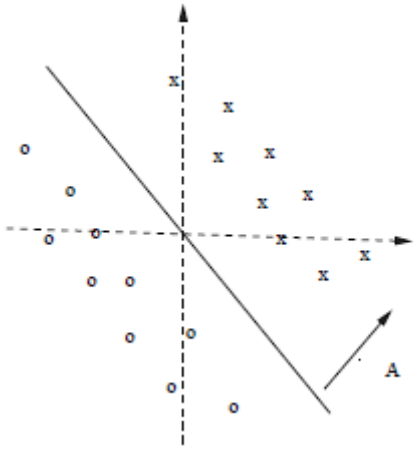

Figure 3The sign of the projection onto the weight vector *A* yields the class label.

In order to illustrate this point, let us consider a simple example ina 2-dimensional feature space, as illustrated in Figure 6.2. In this case,we have illustrated two different classes and the plane corresponding to$Ax = 0$ is illustrated in the same figure. It is evident that the sign ofthe function $A \cdot Xi$yields the class label. Thus, the goal of the approachis to *learn* the set of weights *A* with the use of the training data. Theidea is that we start off with random weights and gradually update themwhen a mistake is made by applying the current function on the trainingexample. The magnitude of the update is regulated by a learning rate μ.This forms the core idea of the *perceptron algorithm*, which is as follows:

Perceptron Algorithm

Inputs: Learning Rate: μ

Training Data (Xi, yi) $\forall i \in \{1 \ldots n\}$

Initialize weight vectors in *A* to 0 or small random numbers

Repeat: Apply each training data to the neural network to check if thesign of $A \cdot Xi$ matches *yi*;if sign of $A \cdot Xi$ does **not** match *yi*, thenupdate weights *A* based on learning rate μ

Until: weights in *A* converge

D. PSO (Particle Swarm Optimization)

PSO is population based technique. Basic idea chosen behind this algorithm is swarm intelligence [8]. PSO designed on social behavior of birds to live in flocks. At initialization of this algorithm there are many solutions we can say population of solution in which any randomly selected that called particle. This algorithm works in iteration manner and moves closer to best solution. Each particle try to move in searching space with dynamic velocity vi that takes particle towards optimal solution at each iteration. At initial point n particles are randomly distributed in search space. On basis of historical behavior itself dynamic adjustment set. Fitness for each particle calculated by using equation 2 and 3 as described in section II for individual classes. In each steps each particle move towards optimal solution with their personal velocity vi changes according to given equation 4 as:

vi(t+1)=w*vi(t)+bp*rand*(pi-ci)+bg*rand*(g-ci)

Where position of each particle represent by pi that is personal for each one particle. g is best solution globally, current position of each particle represent by ci, there is a best constant for each particle that best is bp, inertial constant is w that change with iteration which value varies 0.9 to 0.2, as like personal best constant there is a global best constant bg, where I value varies from 0 to n. movement of each particle calculated by using:

ci(t+1)=ci(t)+vi 5

Where ith particle is represent by xi=(xi1, xi2 …. xiD) and velocity vi=(vi1, vi2 …..viD). so by using above equations fitness function, particle best solution and global best solution can be evaluated. The process continues till termination point not hit.

International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 3, Issue 1, July 2014)

At termination point where process stops we got optimal solution for problem. We have to follow procedure to get optimal solution as following:

Procedure:

Step 1 initialize n particle.

Step 2 repeat the process till termination point hit

a. Calculate particle fitness (fi) using equation 2.
b. Global best position is the best fit particle.
c. All particle moves towards global best position by following equations 4 and 5.
d. Personal best = current position for each particle if condition (fitness of current position < fitness of personal best) satisfy.
e. Update personal best position for each particle.
f. Global best position value calculated.

Step 3 Cluster Center is global best position.

V. Comparative Results Of Algorithms

The behaviors of the classifiers are very similar across the classification tasks. This is illustrated in Fig. 4 which shows theperformance of the three algorithms on two typical binary classification tasksamong the 190. The figure shows how the performance depends on the numberof documents in the train set. Fig. 4 shows that neural classifier is slightly

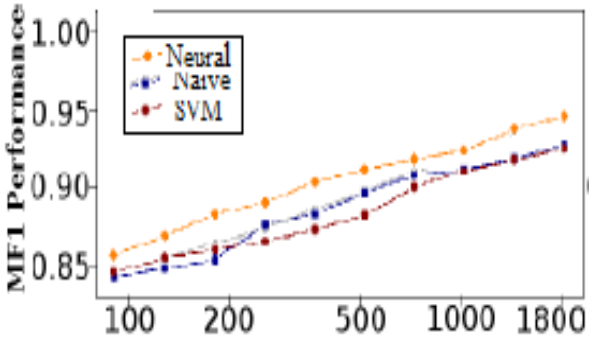

Fig 4 Performance of SVM, Neural, and Naïve Bayes

Better than the other algorithms for all train sizes. However, the differencefrom the worst performer is not very large (about 2 or 3%).

Now we are going to represent the processing time required for the execution of task. With the help of graph in figure 5 we can understand easily.

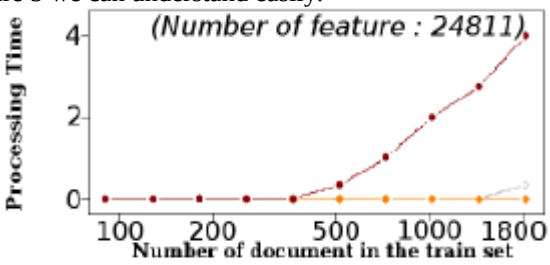

Fig 4 Processing time for SVM, Neural, and Naïve Bayes

Regards the number of features, all three classifiers tends to achieve betterperformance on large feature set. However, the SVMprocessing time can be particularly high if the number of documents is small. a maximum is reached for a relatively small feature set. Then, theperformance decreases until it reverses its tendency again.

VI. Conclusion

As we discuss text classification with existing technique SVM, Naïve Bayes, and Neural. SVM perform well in case of number system and other dataset. But for online text classification there is performance of neural is better than other existing algorithms. We need a concept due to which performance may improve much more as compared to existing. So for this we can choose html as a helping partner of SVM to classify online data like news.In this paper we find conclusion that SVM is most common useful technique from existing supervised learning other techniques.

REFERENCES

[1] CanasaiKruengkrai ,ChuleeratJaruskulchai, "A Parallel Learning Algorithm for Text Classification,"TheEighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2002), Canada, July 2002.

[2] Zhang W., Yoshida T., and Tang X," Textclassification using multi-word features", In proceedings ofthe IEEE international conference on Systems, Man andCybernetics, pp. 3519 – 3524, 2007.

[3] Vladimir N. Vapnik,"The Nature of Statistical Learning Theory", Springer, New York, 1995.

[4] J. Kivinen, M. Warmuth, and P. Auer,"The perceptron algorithm vs. winnow: Linear vs. logarithmic mistake bounds when few input variables are relevant", In Conference on Computational Learning Theory, 1995.

[5] VladimirN.Vapnik, "Statistical Learning Theory", Wiley& Sons, 1998, New York, ISBN 0-471-03003-1.

[6] ThorstenJoachims,"Learning to Classify Text using Support Vector Machines", Kluwer Academic Publishers, Boston, 2002, ISBN 0-7923-7679-X.

[7] Kim S. B., Rim H. C., Yook D. S. and Lim H. S., "Effective Methods for Improving Naïve Bayes Text Classifiers", LNAI 2417, 2002, pp. 414-423.

[8] AjithAbraham,CrinaGrosanVitorino Ramos (Eds.) E- book "Swarm intelligence in Data mining" , (April,2006).