OFW-ITS-LSSVM: Weighted Classification by LS-SVM for Diabetes diagnosis

Ontological Feature weights (OFW) and intensified Tabu Search (ITS) for optimization

Fawzi Elias Bekri
Department of Computer Science & Engineering
JNTU, Hyderabad- 500 085, Andhra Pradesh, India

Dr. A. Govardhan
Professor of Computer Science & Engineering
Principal College of engineering, JNTUH, Jagityal,

***Abstract*—In accordance to the fast developing technology now a days, every field is gaining it's benefit through machines other than human involvement. Many changes are being made much advancement is possible by this developing technology. Likewise this technology is too gaining its importance in bioinformatics especially to analyse data. As we all know that diabetes is one of the present day deadly diseases prevailing. So in this paper we introduce LS-SVM classification to understand which datasets of blood may have the chance to get diabetes. Further, considering the patient's details we can predict where he has a chance to get diabetes, if so measures to cure or stop it. In this method, an optimal Tabu search model will be suggested to reduce the chances of getting it in the future.**

Keywords-machine learning; SVM; Feature reduction; feature optimization; tabu search.

I. INTRODUCTION

In the present situation we can say that diabetes has no cure. In real, it happens due to the lack of insulin which has to do along with glucose in our body. It has to be supplied into our body during loss conditions externally. Indirectly, I is the main cause for fatal heart, kidney, eye and nerve diseases, which can be overcome or prevented by good food habits and body exercises[1].

Over all this, the difficult thing is to differentiate between disease diagnosis and interpretation of diabetes data. For doing this, we are with Support vector Machine(SVM) which was developed by Vepnik[2]. Its work has been tested in many ways[3][12][4]. The utmost advantage with it is that it can even work good with nonlinear functions and it contains Radial basis Functions(RBF) which is even more precise than polynomial and linear kernel functions. The comparison of this SVM with other methods like Combined Neural Networks (CNNs), Mixture of Experts (MEs), Multilayer Perceptrons (MLPs), Probabilistic Neural Networks (PNNs) also revealed that svm methods are perfect.

In estimating the diabetes features, Feature Selection is applied. By the above analysis, if we are left with 8 features, we can come down to 4 by this feature selection. It can take out the factors not conserned with the feature set.[5][6][7][8].

PCA (Principal Component analysis) is one of the feature selection method recently gaining its importance being used in image recognision, signal processing, face recognition etc.

By applying SVM to disease datasets, it can grab a large circumference of data sets even relevant or not relevant to the diagnosing the disease. But by such features with variation, the diagnosing will not be perfect and so weighted factors are to be developed. And they were contributed by Zhichao Wang [9] giving those weights by their ontological relevance.

The LS-SVM technique at last works with 2 parameters for accurate results. Out of many datasets and values came from SVM, the choosing of 2 parameters is very important, If very high features are chosen, some datasets will be missed and if chosen with utter accuracy and care, leads to under-fitting[6,13]. So, 2 optimised solutions are to be found out possibly by Intensified Tabu Search (ITS)[14].

The working of this ITS involves 3 phases. PCA, discussed above, is to get rid of irrelevant evidences given by SVM. Then OFW is to calculate the weight of each factor which PCA thought relevant. Then comes ITS which can find out the best possible 2 parameters for SVM so that it may not be under fit or over fit.

To have a quick look on what paper contains, we shall see the initial data sets of diabetes in section II, then our 1st step of PCA reduction in section III, to weighted preferences in section IV, then OFW in section V followed by experimental results in later sections.

II. DATASET OVERVIEW

The initial data sets are gathered form UCI Machine Learning Repository[16]. It contains almost 8 categories on a whole and 768 sub categories which is really a very large database. The attributes are choose from these large data sets may be either discrete or continuous with an interval[17]. The large data base, provided now is from the following:

- Pregnant: Number of times of pregnant
- Plasma-Glucose: Plasma glucose concentration measured using a two-hour oral glucose tolerance test. Blood sugar level.
- BMI: Body mass index (w in kg/h in m)
- DPF: Diabetes pedigree function
- TricepsSFT: Triceps skin fold thickness (mm)
- Serum-Insulin: 2-hour serum insulin (mu U/mt)

- DiastolicBP: Diastolic blood pressure (mmHg)
- Age: Age of the patient (years)
- Class: Diabetes onset within five years (0 or 1)

III. FEATURE SELECTION

The 1st method which runs for reducing the data base is feature selection. The complexity of data can be reduced so that we can be left eith less datasets and can be more precise. Then comes PCA helping the classification to happen further with the help of statistical measures. The simplification of data by PCA is as follows:

D n-dimension dataset.

M principle axes a1,a2,…. These are orthogonal axes… then, covariance matrix is:

$$s = \left(\frac{1}{L}\right)\sum_{k=1}^{L}(x_k - p)^T (x_k - p)\ \ x_k \in D \quad (1)$$

Where m is the average of samples, and L is the number of samples. Therefore

$$sv_k = \lambda_k v_k \quad k \in 1,....,n \quad (2)$$

Where λ_k is the k^{th} largest Eigen value of s . The m principal components of a given sample $x_k \in D$ are given in the following

$$q = [q_1, q_2, ..., q_n] = A^T x_k \quad A = [a_1, a_2, ..., a_n]: \quad (3)$$

where $q_1, q_2, ..., q_n$ are the principal components of x_k .

LS-SVM: Of all the paper, we discussed the key idea of using SVM brought up by Vapnik[2]which plays a main role in collecting the wide database for our problem. It also has its use in solving pattern recognition and classification problems. The methods present in SVM other than polynomial and linear are its greatest assets which made it to lead global models containing structural risk minimization principle[19]. Though SVM sounds easy due to its extended results, finding the solution is difficult and what all can do is to find sparse solutions. Its difficulty arises from finding nonlinear equations. So as a solution, Suykens and Vandewalle [20] introduced least-squares SVM which results out linear equations. For the new type of SVM also the further proceeding like PCA, OFW and its usage in quantification and classification are applicable and reported in some works[23,24].

In calculation of linear equation, (y=wx+b), we use the 2 axes like regression(x) and dependent variable (y). And the best minimised cost function is

$$Q = \frac{1}{2} w^T w + \frac{1}{2}\gamma \sum_{i=1}^{N} e_i^2 \quad (3)$$

Subject to: $y_i = w^T \phi(x_i) + b + e_i \qquad i = 1,....N \quad (4)$

The formula's two parts are weight decay the 1st to generalize weights and regression error of training data is the second, whereas the parameter indicated by g is to be optimized by the user.

For a better generalization model, the most important criteria are the proper selection of features for the RBF kernel and polynomial kernel.

IV. ONTOLOGY-BASED FEATURE WEIGHTING CALCULATION

A. Feature Weight Calculation

The process of computing domain ontology feature and ontology feature weight is as follow:

a. Characteristic of the information is individually treated as a semantic category and is considered as an ontology semantic peer. The characteristics are grouped based on their semantic principles.
b. The whole relevancy of a feature is used to calculate weight of the characteristic in the ontology tree.

B. Domain Ontology-feature Graph

We construct ontology-feature graph w.r.t. a particular column of information in order to represent the domain knowledge model. There are three layers in the graph. They are:

i. Concept layer
ii. Attribute layer
iii. Data-type layer

Let us discuss them in detail,

Concept layer:

First layer has all the concepts of the ontology called ontology concept. It is explained by the attribute nodes and remaining elements of the concept layer. It can be represented as:

Ontology-Concept = {Cpt.1, Cpt.2, …., Cpt.n_{cpt}}.

For each layer, an object is considered as a node.

Attribute layer:

Second layer, Attribute layer explains the nodes in the concept layer i.e. ontology attribute following the regulations of the characteristic set.

Ontology- Attribute={Ab.1, Ab.2, ..., Ab.n_{ab}}.

Data Type Layer:

This layer explains the node of the Attribute layer following the regulations of the metadata layer i.e. Ontology-Data type.

Ontology- Data type={Dt.1, Dt.2, …, Dt.n_{dt}}

In the figure 1, the solid line shows the relative characteristics of the concept semantic layer and attribute layer whereas, dotted lines show the data type layer nodes individually.

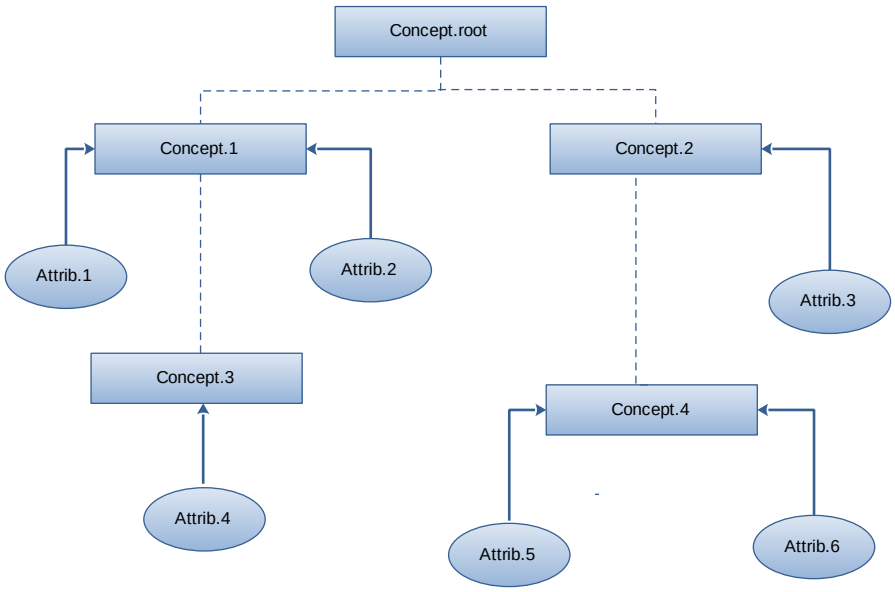

Figure 1. Ontology Feature Graph

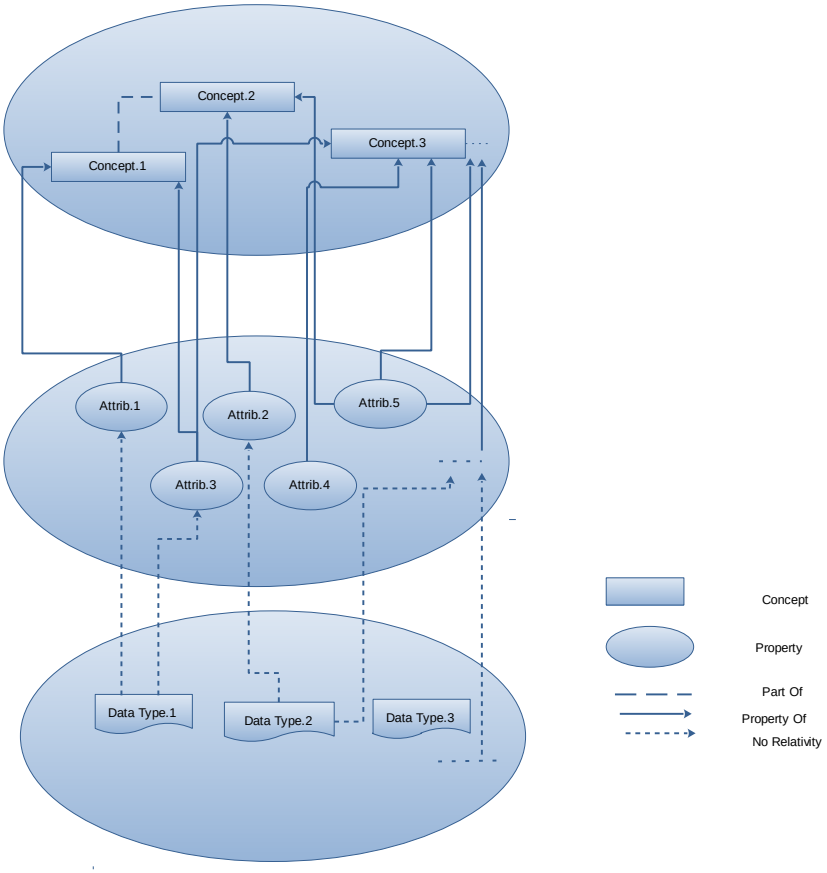

Figure 2. Ontology Feature Tree

The characteristics of the information source and database storage logical model in the domain ontology feature graph are used to compose the data object ontology node. The latter one is used to construct the nodes of the concept layer of the domain ontology characteristic graph based on its design pattern. Generally, the data object ontology node is constructed based on the remaining of the data source of the ontology graph. The database is composed by using the ontology principle as the

primary rule and the principle layer nodes of the graph which is constructed based on database.

The ontology attribute layer is constructed from the elements of the attributes which are the characteristics of the concept layer. The data type layer is used to convert the data types of the attributes to the semantic extension type. The correction between the concept layer and the attribute layer nodes can be described and computed based on the considered s of the principle layers. The communication and computation between the principle and attribute layer nodes is as following:

Semantic Ontology Correlation: The two ontology topics follow the format with predicate suchas as <Subject, Predicate, Object>. Here, Predicate represents the group of predicates,

Predicate= { partOf ,propertyOf },

It is mainly utilized in explaining the ontology predicate. Ontology relations, Concept-Concept and Concept-Attribute, are refered to as the CC and CA following the predicate set explanation.

CC stands for CC =<Concept.i, partOf, Concept.j>

CA stands for CA = <Concept.i, propertyOf, Attribute.k>

Concept.i,Concept.j ∈ Ontology-Concept;

Attribute.k ∈ Ontology-Attribute.

$$i, j < n_{cpt}, k < n_{ab}$$

Basic assumption: (Ab.i1, Ab.i2, … , $Ab.in_i$), Concept.j (Ab.j1,Ab.j2, … , $Ab.jn_j$).

$$n_i, n_j < n_{ab}.$$

The primary relation between Concept.i and Concept.j is Correlation (Concept.i,Concept.j):

$$correlation(concepti, conceptj) = \frac{|\ concepti(Ab.in_i) \bigcap conceptj(Ab.j1,\ j2,....Ab, in_j)\ |}{|concepti(Ab.in_i) \bigcup\ conceptj(Ab.j1,\ j2....Ab, in_j)|}$$

Domain Ontology Feature Tree:

It is mainly used to refer to the relationship among the nodes of the attribute layer, concept layer and their characteristics (represented in the domain ontology characteristic graph). We also make use of it in the computation of correlation between ontology-concepts and ontology-attributes.

Domain ontology characteristic tree can be referred to with the triples as

Ontology-Tree, Ontology-Tree = {<Cpt.root,partOf,Cpt.1>, <Cpt.root,partOf,Cpt.2>, <Cpt.1,propertyOf, Ab.1>, ……}

Here, in the tree, the final node i.e. the leaf node is one of the characteristics of the domain ontology whereas the branches can be represented by the concept. Thus, the discussed correlation can be calculated as:

$$Correlation(Ab.i, Ab.j) = \frac{Height(Ab.i) + Height(Ab.j)^{*}\alpha + Datatype(Ab.i, Ab.j)^{*}\beta}{(Dis\tan ce(Ab.i, Ab.j) + \alpha) + 2^{*} MAX(Height(Ab.i), HeightAb.j)) + \beta}$$

Where,

Height(Ab.i),Height(Ab.j) refer to characteristics hierarchy Ab.i and Ab.j of the concerned tree.

Boolean function Data Type(Ab.i, Ab.j)is used to compare data types of features Ab.i and Ab.j.

Distance(Ab.i, Ab.j) is the shortest path to the elements.

Max(Height(Ab.i),Height(Ab.j)) refers to the maximum length of the tree.

α,β represent the variable parameters with 0< α,β<1.

α maintains the Height and Distance ratio;

β is used in type conversion.

Here, in figure 2, the relation between Attrib.4 and Attrib.3 is computed. Height(Ab.4) = 3.

Height(Ab.3) = 2.

As we can see that both Ab.4 and Ab.3 differ in data type, Data Type(Ab.4,Ab.3)=0. Distance(Ab.4, Ab.3) = 5.

$$Correlation(Ab.4, Ab.3) = (5^{*}\alpha)/(5+\alpha+2^{*}3+\beta) = (5^{*}\alpha)/(11+\alpha+\beta)$$

Similarly, the relation between Ab.4 and Ab.2 is also computed.

Height(Ab.4) = 3;

Height(Ab.2) = 2;

As we can see that both Ab.4 and Ab.3 differ in data type, Data Type(Ab.4,Ab.2)=0. Distance(Ab.4, Ab.2)=3.

*So, Correlation(Ab.4, Ab.2) = (5*α)/(3 +α+2*3+β) = (5*α)/(9+α+β).*

Thus, we can say that Correlation(Ab.4, Ab.2)> Correlation(Ab.4,Ab.3). Hence, Ab.4 and Ab.2 are more similar. The results might change with the variables but the actual one doesn't change.

This shows that, the values of the arbitrary parameters remain unaffected over the relation among attributes. However, we can better the situation by choosing proper parameters through various tests. Thus, the formula to compute the weight of a characteristic can be drawn from the above co relations as,

$$Weight(Ab.k) = Average\sum_{i=1}^{m} correlation(Ab.k, Ab.i)$$

OFW-LSSVM

According to the Conventional LS-SVMs, the given function is performed by the equal contributions from all the

characteristics. But, actually, the various characteristics play different roles with various weights. Thus, different contributions from different characteristics can be performed by using the theory proposed by Zhichao Wang [9].

Given,

$\{x_i, y_i\}_{i=1}^{N}$

Represents coaching group and

$\alpha \in R^d$, where α represents the weighted vector.

$$\sum_{i=1}^{d} \alpha_i = 1, \quad \alpha_i \geq 0 \quad \text{-----(8)}$$

Now, the equation (3) can be used to provide optimal solution to the problem (4), which is as follows:

$$\min \frac{1}{2} \| w \|^2$$

$$s.t. y_i (w.diag(\alpha))$$

Where,

$$diag(\alpha) = \begin{pmatrix} \alpha_1 & 0\ldots & 0 \\ 0 & \alpha_2 \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & 0\ldots & \alpha_d \end{pmatrix}$$

Substituting (8) and (9) into (5), yields the following new optimization problem:

$$\min_{w,\xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{N} \xi_i$$

$$sty_i (w.diag(\alpha) x_i + b) \geq 1 - \xi_i, \geq 0, i = 1, 2, ..., n$$

$$\sum_{i=1}^{d} \alpha_i = 1, \alpha_i \geq 0$$

Thus, we can write the categorization decision method is:

$$f(x) = sign(\sum_{i=1}^{N} L_i y_i K'(X_i, X_j) + b)$$

$K'(X_i, X_j)$ represents the weighted characteristic of the RBF kernel as

$$K'(X_i, X_j) = \exp(-\gamma \sqrt{\sum_{k=1}^{d} \alpha_k (x_{ik} - x_{jk})^2})$$

Intensified Tabu search (ITS) for characteristic selection

As we know, BDF chooses the characteristics based on the betterment of the recognition rate. After considering we came to know that BDF increases the count of support vectors according to the size of the problem [26]. This feature seems to be interesting in producing quick and better decision method, but it applies only if it is connected to the betterment in the recognition rate.

The support vectors and some more characteristics care mainly used to provide a quick and better SVM BDF. Due to this reason, in order to solve the conflict between the complexity and performance, Decision Function Quality (DFQ) criterion is used in association with regularization theory. Thus, SVM makes sure to coach right from the basic i.e. tiny dataset St', where it stands for the primary coaching group St. It will reduce the ambiguity related to the BDF. Even the primary set is also further optimized by using LBG algorithm based on certain assumptions. The basic assumption is to consider parameter k as a variable of problem in choosing model. It is so, because k may not be able to handle all kinds of prototypes generated by LBG algorithm during the process.

Hence, the value of k (i.e. the range of optimization), the characteristic subgroup β, the regularization constant C along with attributes of the kernel such as (σ with gaussian kernel) must be selected for every kernel method K using the model selection method. If we consider θ as a model, k_θ, β_θ, C_θ and σ_θ respectively will be the representatives of the attributes discussed so far. Moreover, q(θ) represents the DFQ criterion for a model θ (c.f. Section 3.1).

The Section 3.3 deals with the presumption of DFQ criterion along with a learning set S_l showing q(θ) ≡ SVM-DFQ(θ, S_l) which is to be optimized for model θ. The optimizing θ^* for q(θ) not being tractable, we decide to define a TS function for choosing a model with optimal intensification and diversification methodologies.

Decision Function Quality(DFQ):

For smooth calculation of the equation, we need DFQ for the theta we have. It can be known by the recognition rate RR with the help of complexity CP of decision function hu. Here comes the q(θ) = $R_R(h_\theta) - C_p(h_\theta)$ be the DFQ[25].

Here, the correct and accurate result from equation can be calculated by using smoothness term and fitting term in terms of recognition rate (RR).Cp indicates the smoothness term. The model complexity of a SVM BDF[25] is given by

$$C_p(h_\theta) = C_{p1} \log_2(\eta sv) + C_{p2} \log_2(\cos t(\beta)) \quad (5)$$

To discuss the parameters of the function, cp1 and cp2 are tradeoff between classification rate improvement and complexity reduction. Beta is a Boolean vector with n size of represented features. Ki is to represent cost for ith feature cost (beta) combined to the subset of selected features is:

cost (β) = $\sum \beta_i k_i$. When those costs are unknown, k_i = 1 is used for all features.

Simplification Step:

Reducing training set size is the simplest way to reduce complexity of SVM. This LBG algorithm [25] is used to simplify the dataset . The simplification details are in the below table and can be used in the further discussion:

Simplification(S,k)
S′ ⇐ ∅
FOR c ∈ {−1, +1}
T = {x | (x, c) ∈ S}
IF 2^k < |T| THEN T′ ⇐ LBG(T, k)
ELSE T′ ⇐ T
S′ ⇐ S′ ∪ {(x, c) | x ∈ T′} ENDFOR
RETURN S′

TABLE I. SYNOPSIS OF SIMPLIFICATION STEP

DFQ estimation

The Decision Function Quality (DFQ)[25] criterion of a particular model θ is calculated from a attained dataset Sl. we can observer the elocation of values from the details given in the Table 3 .Let s_t , s_v represents the datasets produced in a random split (Split function in synopsis SVM-DFQ) with | s_t | = $\frac{2}{3}S_l$, | s_v | = $\frac{1}{3}S_l$. s_t , s_v will be signifying the databases utilized to train SVM (training dataset) and to identify rate consideration (validation dataset). This dissociation is important in order to overcome the risk of over fitting when empirical estimation is used. The SMO algorithm version of the Torch library [31] is used to realize SVM training step. When SVM training is per-formed with unbalanced class datasets, it is more suitable to use Balanced Error Rate (BER) instead of classical Error Rate for the estimation of recognition rate. Recognition rate formulation (noted R_R) in Table 2 corresponds to BER estimation where m_y represents the number of examples in each class(y ∈ {+1,−1}) and $m_y^{correct}$ the number of examples correctly identified. . The kernel functions k_β utilized for training SVM are decided from a distance $d_\beta : d_\beta(x_i, x_j) = \sqrt{\sum_{l=1}^{n} \beta_l (X_i^l - x_j^l)^2}$. Utilizing d_β in the kernel function, the feature selection problem is embedded in the model selection problem. In the present study Gaussian kernels $K_\beta^G = \exp(\frac{-d_\beta^2}{\lambda_1^2})$ are utilized.

SVM-DFQ(θ,Sl)
(s_t , s_v) ⇐ Split(S_l)
$S_t^{'}$ ⇐ Simplification(s_t ,, k_θ)
h_θ ⇐ Training SVM($S_t^{'}, k_{\beta\theta}, c_\theta, \sigma_\theta$)
($m_{-1}^{correct}$, $m_{+1}^{correct}$) ⇐ Testing BDF(h_θ , S_v)
$R_R \Leftarrow \frac{m_{-1}^{correct}}{2_{m-1}} + \frac{m_{+1}^{correct}}{2_{m+1}}$ c_p ⇐ Complexity(h_θ)
q(θ) ⇐ $R_R - c_p$

TABLE II. SYNOPSIS OF DFQ CALCULATING FOR A DEFINED MODEL θ

V. FEATURE OTIMIZATION

Tabu Search specification

The main function q to be obtained produces the quality of the BDF h_θ . The main issue is to select an optimal model (good sub-optimal solution to be exact) θ^* for a function q when C_{p1} and C_{p2} are affixed. A model θ can be denoted by a set of n′ integer variables θ = (θ_1 , . . . , $\theta_{n'}$) = (β_1 , . . . , β_n , k,C′, σ′). Notations k_θ , β_θ , C_θ , σ_θ correspond respectively to k, (β_1 , . . . , β_n), $\sqrt{2}^{C'}$ and $\sqrt{2}^{\sigma'}$ in that integer representation of θ model. One basic move in our TS method corresponds to adding δ ∈ [−1, 1] to the value of a θ_i , while preserving the constraints of the model which depend on it (i.e.∀i ∈ [1, . . . , n′], θ_i ∈ [min(θ_i), . . . ,max(θ_i)] where min(θ_i) and max(θ_i) respectively denote lower and upper

bound values of θ_i variable). Above all the list of all possible neighborhood solutions is added. Among these possible solutions, the apt DFQ that is not tabu is selected. The set of all θ_{tabu}^{it} solutions θ which are tabu at the it repeated step of TS is defined as follows: θ_{tabu}^{it} = {θ ∈ Ω | ∃ i, t′ : t′ ∈[1, . . . , t], $\theta_i \neq \theta_i^{it-1} \wedge \theta_i = \theta_i^{it-t'}$ } with - the set of all solutions and t an adjustable parameter for the short memory used by TS (for experimental results t = $\sum_{i=1}^{n'} \max(\theta_i) - \min(\theta_i)$. The idea is that a variable θ_i could be changed only if its new value is not present in the short memory. Then, our TS method does not go back to a value of θ_i previously changed in short time, avoiding by that mechanism undesirable oscillation effects. Tabu status of solutions θ_{tabu}^{it} may prohibit some attractive moves at iteration it. Therefore, our TS uses an aspiration criterion which consists in allowing a move (even if it is tabu) if it results in a solution with an objective value better than that of the current best-known solution.

The initialization of model θ with our TS model selection is the following:

- $K_\theta - [\log_2(\max(m_{+1}, m_{-1}))/3)$
- C_θ = 1 and σ_θ = 1,
- ∀ i : β_i = 1.

In the present formula K_θ , m_{+1} and m_{-1} denotes positive and negative classes in binary sub-problems. The value of K_θ permits to begin with enough minimum datasets to get low training times with SVM for the first step.

Using intensification and diversification strategies develops TS methods [30]. The selected model should handle two kinds of problems. The first problem is testing all moves between two repetitions with a great number of features which is time-consuming. Especially, it is a waste of time to investigate moves which are linked to features where real solution is not suitable. Thus, emphasizing on moves which are only linked to SVM hyper parameters or simplification level is better than to discover new solutions. Coming to second problem, it is difficult for TS method to free from deep valleys or big clusters of poor solutions by using the short memory which effect in not tab solutions. Utilizing diversified solutions helps in win over of the problem. This is handled by enlarging step size (δ > 1) of moves and by pointing the use of all types of moves (except feature selection moves for the reason stated above).In present TS method, intensification and diversification strategies are utilized one by one and start with the intensification strategy. Later on we deal about the two strategies.

Intensification strategy

In the intensification algorithm synopsis of Table 4, Extensive Search survey all possible basic moves, whereas Fast Extensive Search explores only eligible basic moves which are not related to feature selection (i.e. changing the value of β). $\eta_{promising}$ Controls when the real solution is seen as enough and this one allows switching between the two functions mentioned.

BestNotTabu correlate to the move procedure chosen in the above part (the best tabu solution is chosen if all moves are tabu). In this synopsis, $\theta_{intensification}$ corresponds to the best solution found into a same phase of intensification, although $\theta_{best-known}$ corresponds to the best solution found in all intensification and diversification steps.

Nmax is the maximum number of intensification redundancy for which no development of the last best intensification solution $(\theta_{intensification})$ are identified as failure of the intensification strategy. Nfailure counts the number of failures of intensification strategy.

If Nfailure is higher than a fixed maximum number of failures max then ITS method stops and returns the solution $\theta_{best-known}$. If a solution in £ next has a QDF which is better than $\theta_{best-known}$, aspiration mechanism is used. That solution is selected as the new $\theta_{best-known}$ and $n_{failure}$ is reset to zero.

Diversification strategy

In the diversification algorithm synopsis of Table 5, suitable variable (one which does not have a link with features) is selected (Select Eligible Variable) by random and a jump of ±δ is performed by modifying the chosen variable in the real solution.

There are only two investigated moves (Two Move) to force the diversification of identified solutions. The jump size enlarges with the number of successive failures ($n_{failure}$) of the intensification strategy to investigate more different regions.

In the process of the diversification redundancy, the best visited solution is saved $\theta_{devertification}$ and chosen as the bening solution for the next intensification step $(\theta^{it}{}_{intensification} = \theta^{it-1}{}_{deversification})$. In the TS investigation, when aspiration is included, the strategy automatically moves to intensification and the number of failures is rearranged ($n_{failure} = 0$).

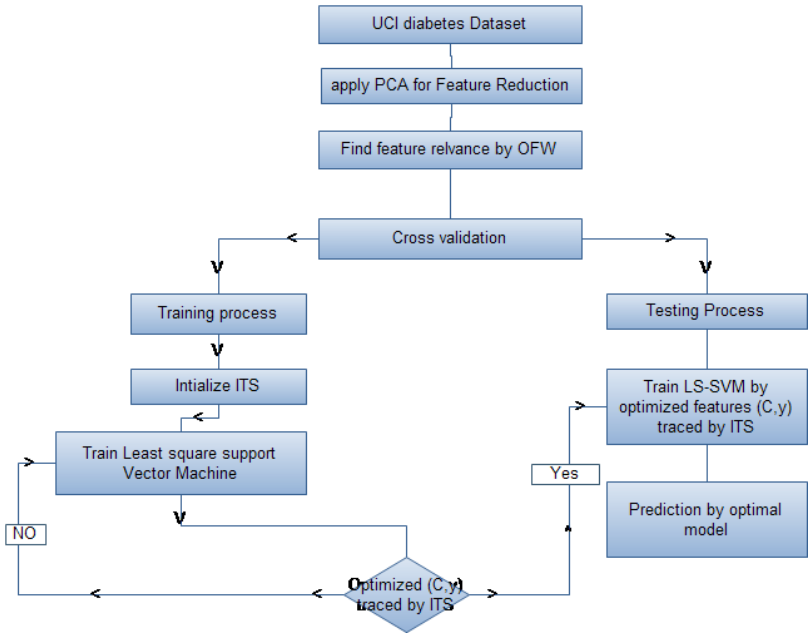

Figure 3. Flowchart of the OFW-ITS-LSSVM

VI. Proposed Method Ofw-Its-Lssvm

This part explains the desired method (OFW-ITS-LSSVM) for the identifying of diabetes diseases (see figure3). Especially the system works in three stages automatically

1) PCA is applied for feature reduction
2) Best feature weights are estimated using OFW
3) ITS is employed for finding the optimal values for C and γ.

At first, PCA method is used to identify four features from diabetes dataset. Thus, in feature choosing stage, only large principal components will be utilized. Then, the OFW-LSSVM is used to classify patients, the feature weights which are received by OFW and at last , the MCS algorithm is used to detect the best value for C and γ parameters of OFW-LSSVM. The description of training procedure is:

1. Set up parameters of ITS and initialize the population of n nests (Algorithm 1)
2. Compute the corresponding fitness function formulated by $\frac{classified}{total}$ (total denotes the number of training samples, and classified denotes the number of correct classified samples) for each particle.
3. 3. Find the best solution using ITS

VII. Experimental Results

The OFW-ITS-LSSVM model was compared with other popular models like LS-SVM, PCA-LS-SVM, PCA-MI-LS-SVM, MI-CS-SVM and PCA-PSO-LS-SVM classifiers. We utilized fold cross validation develop the holdout method. The data set was divided into k subsets, and the holdout method was iterated k times.

Every time, one of the k subsets is utilized as the test set and rest are put together to form a training set. Then the average error across all k trials is computed (Polat and Günes, 2007). This method was used as 10 -fold cross validation in our experiments. We considered the related parameters of PSO in PCA-PSO-LS- SVM classifier as follows: swarm size was set to 50; the parameters C and γ were arbitrary taken from the intervals $[10^{-3}, 200] and [10^{-3}, 2]$, respectively.

The inertia weight was 0.9, acceleration constants C_1 and C_2 was fixed to 2, and maximum number of redundancy was fixed to 70. Classification results of classifiers were shown in a confusion matrix. Like displayed in table 4, each cell has the raw number of examples categories to correspond intergration of real system results.

Output/desired	Non-diabetic	Diabetic	Method
Non-diabetics	44	6	LS-SVM (Polat et al., 2008)
Diabetics	11	17	
Non-diabetics	45	5	PCA-LS-SVM
Diabetics	9	19	
Non-diabetics	44	6	PCA-MI-LS-SVM
Diabetics	4	24	
Non-diabetics	45	5	PCA-PSO-LS-SVM
Diabetics	4	24	
Non-diabetics	48	2	MI-MCS-SVM
Diabetics	3	25	
Non-diabetics	49	1	OFW-ITS-LS-SVM
Diabetics	1	27	

TABLE III. Confusion Matrix

Thus it shows the frequency of disease how a patient is misclassified. Furthermore, Table 5 displays the categories accuracies of OFW-ITS-LSSVM. The present model gets the correct categories accuracy of 95.78% among classifiers on the test set. Determining the test performance of the classifiers is done by addition of specificity and sensitivity that are classified as: Specificity: number of true negative decisions / number of real negative case sensitivity: number of true positive decisions / number of real positive cases.

A true positive decision happens only if the positive expectation of the network mingles with a positive expectation of the physician. A true negative decision happens if the two i.e. network and the physician advice negative expectation.

Methods	Sensitivity (%)	Specificity (%)	Classification accuracy
LSSVM [32]	73.91	80	78.21
LSSVM with PCA [33]	79.16	83.33	82.05
LSSVM with MI and PCA[2]	80	91.66	87.17
LSSVM with PCA and PSO[2]	82.75	91.83	88.46
SVM with PCA, IM and MCS[2]	92.59	94.11	93.58
OFW-ITS-LSSVM	94.96	97.76	95.78

TABLE IV. THE VALUES OF THE STATISTICAL PARAMETERS OF THE CLASSIFIERS

As per the Table 6, it is observed that utilizing the LSSVM classifier with OFW and ITS, it is easy to get the correct classification accuracy compared to other methods. Hence it is apt to say that this method gives a high rate of accuracy in identifying of Diabetes disease. The method can also combine with software to help the physicians to take final decision confidently.

Method	Classification accuracy
QDA	59.5
C4.5 rules	67
RBF	68.23
C4.5 (5xCV)	72
Bayes	72.2
Kohonen	72.8
ASR	74.3
DB-CART	74.4
Naïve Bayes	74.5
CART DT	74.7
BP	75.2
SNB	75.4
NB	75.5
kNN	75.5
MML	75.5
RBF	75.7
LVQ	75.8
Semi-Naïve Bayes (5xCV)	76
MLP + BP	76.4
FDA	76.5
ASI	76.6
SMART	76.8
GTO DT (5xCV)	76.8
BFGS quasi Newton	77.08
LM	77.08
LDA	77.5
GD	77.6
SVM (5xCV)	77.6
GDA-LS-SVM	79.16
GRNN	80.21
LDA-MWSVM	89.74
MI-MCS-SVM	93.58
OFW-ITS-LSSVM	95.78

TABLE V. CLASSIFICATION ACCURACY: COMPARING OFW-ITS-LSSVM WITH OTHER METHODS FROM LITERATURE

VIII. CONCLUSIONS

Over all the work propose a new automatic method to diagnose Diabetes disease depend on Feature Weighted Support Vector Machines and Modified Cuckoo Search. For discarding the other features, Principal Component Analysis was utilized. Later Mutual Information was used to the chose features to weight them depend on their related task of classification. Outcome proves that it devises the accuracy of the method. In addition to, Modified Cuckoo Search is utilized that allows the quick change of the algorithm and locate the correct values for parameters of SVM. The outcome has proved that the present model is faster and significantly more reliable than other models.. The method can also combined with software to help the physicians to take final decision confidently in order to diagnose Diabetic disease.

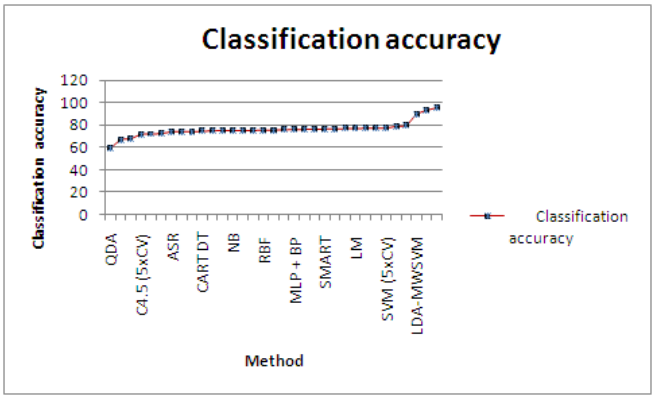

Figure 4. Line chart representation of comparing OFW-ITS-LSSVM with other methods from literature

REFERENCES

[1] [1] Polat, K., Güneş, S., 2007. An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. Digital Signal Processing 17, 702-710.

[2] Vapnik, V., 1995. The Nature of Statistical Learning Theory, New York.

[3] Çalişir, D., Doğantekin, E., 2011. An automatic diabetes diagnosis system based on LDA-Wavelet Support Vector

[4] Übeyli, E.D., 2007. Comparison of different classification algorithms in clinical decision-making. Expert Systems 24, 17-31.

[5] Acır, N., Özdamar, Ö., Güzeliş, C., 2006. Automatic classification of auditory brainstem responses using SVM-based feature selection algorithm for threshold detection. Engineering Applications of Artificial Intelligence 19, 209-218.

[6] Lin, M., Oki, T., Holloway, T., Streets, D.G., Bengtsson, M., Kanae, S., 2008. Long-range transport of acidifying substances in East Asia—Part I:

model evaluation and sensitivity studies. Atmospheric Environment, in press, doi:10.1016/j.atmosenv.2008.04.008.

[7] Valentini, G., Muselli, M., Ruffino, F., 2004. Cancer recognition with bagged ensembles of support vector machines. Neurocomputing 56, 461-466.

[8] Zhang, Y.L., Guo, N., Du, H., Li, W.H., 2005. Automated defect recognition of C- SAM images in IC packaging using Support Vector Machines. The International Journal of Advanced Manufacturing Technology 25, 1191-1196.

[9] Lei Zhang , Zhichao Wang "Ontology-based clustering algorithm with feature weights",2010Journal of Computational Information Systems 6:9 (2010) 2959-2966.

[10] Karabatak, M., Ince, M.C., 2009. An expert system for detection of breast cancer based on association rules and neural network. Expert Systems with Applications 36, 3465-3469.

[11] Mehmet Fatih, A., 2009. Support vector machines combined with feature selection for breast cancer diagnosis. Expert Systems with Applications 36, 3240-3247.

[12] Polat, K., Güneş, S., Arslan, A., 2008. A cascade learning system for classification of diabetes disease: Generalized Discriminant Analysis and Least Square Support Vector Machine. Expert Systems with Applications 34, 482-487.

[13] Pardo, M., Sberveglieri, G., 2005. Classification of electronic nose data with support vector machines. Sensors and Actuators B: Chemical 107, 730-737.

[14] Fred Glover, Tabu search fundamentals and uses, http://leeds-faculty.colorado.edu/glover/TS%20-%20Fundamentals&Uses.pdf, 1995

[15] Xing, H.-j., Ha, M.-h., Hu, B.-g., Tian, D.-z., 2009. Linear feature-weighted support vector machine. Fuzzy Information and Engineering 1, 289-305.

[16] Asuncion, A., Newman, D. J. (2007) Pima Indians Diabetes Data Set, UCI Machine Learning Repository, http://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabets, Irvine, CA: University of California, School of Information and Computer Science.

[17] Cios, K. J., Pedrycz, W., Swiniarski, R.W., Kurgan, L. A. (2007) Data Mining: A Knowledge Discovery Approach, New York: Springer.

[18] Vapnik, V.; Statistical Learning Theory, John Wiley: New York, 1998.

[19] Sun J, Xu W, Feng B, A Global Search Strategy of Quantum- Behaved Particle Swarm Optimization. In Proc. of the 2004 IEEE Conf. on Cybernetics and Intelligent Systems, Singapore: 291 – 294, 2004.

[20] Suykens, J. A. K.; Vandewalle, J.; Neural Process. Lett. 1999, 9, 293.

[21] Suykens, J. A. K.; van Gestel, T.; de Brabanter, J.; de Moor, B.; Vandewalle, J.; Least-Squares Support Vector Machines, World Scientifics: Singapore, 2002.

[22] Zou, T.; Dou, Y.; Mi, H.; Zou, J.; Ren, Y.; Anal. Biochem. 2006, 355, 1.

[23] Ke, Y.; Yiyu, C.; Chinese J. Anal. Chem. 2006, 34, 561.

[24] Niazi, A.; Ghasemi, J.; Yazdanipour, A.; Spectrochim. Acta Part A 2007, 68, 523.

[25] Varewyck, M.; Martens, J.-P.; , "A Practical Approach to Model Selection for Support Vector Machines With a Gaussian Kernel," Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on , vol.41, no.2, pp.330-340, April 2011 doi: 10.1109/TSMCB.2010.2053026

[26] I. Steinwart. Sparseness of support vector machines - some asymptotically sharp bounds. In NIPS, pages 169–184, 2004.

[27] A. Tikhonov and V. Arsenin. Solution of Ill-posed Problems. Winston & Sons, 1977.

[28] A. Tikhonov and V. Arsenin. Solution of Ill-posed Problems. Winston & Sons, 1977.

[29] A. Tikhonov and V. Arsenin. Ill-Posed Pproblems: Theory and Applications. Kluwer Academic Publishers, 1994.

[30] F. Glover and M. Laguna. Tabu search. Kluwer Academic Publishers, 1997.

[31] R. Collobert and S. Bengio. SVMTorch: Support vector machines for large-scale regression problems. In Journal of Machine Learning Research, volume 1, pages 143–160, 2001.

[32] Least Squares Support Vector Machines for Classification and nonlinear modelling PASE 2000 (2000) by J. A. K. Suykens posted to classification lssvm pattern_recognition regression svm by Borelli on 2006-01-18

[33] Davar Giveki, Hamid Salimi, GholamReza Bahmanyar, Younes Khademian, Automatic Detection of Diabetes Diagnosis using Feature Weighted Support Vector Machines based on Mutual Information and Modified Cuckoo Search, arXiv:1201.2173v1, ARXIV, 01/2012

AUTHORS PROFILE

Fawzi Elias Bekri

He studied B.Sc IT and M.Sc IT at sikkim manipal University, Manglore. He did his M.Phil at JNTU, Hyderabad. Now he is doing Ph.D at Jawaharlal Nehru Technological University (JNTU), Hyderabad, A. P., India. His areas of interest include Data mining, KDD in healthcare sector, Software Engineering, Databases and Object Oriented Technologies.

Dr.A.Govardhan

Received Ph.D. degree in Computer Science and Engineering from Jawaharlal Nehru Technological University in 2003, M.Tech. from Jawaharlal Nehru University in 1994 and B.E. from Osmania University in 1992. He is working as a Principal of Jawaharlal Nehru Technological University, Jagitial. He has published around 108 papers in various national and international Journals/conferences. His research of interest includes Databases, Data Warehousing & Mining, Information Retrieval, Computer Networks, Image Processing, Software Engineering, Search Engines and Object Oriented Technologies.