Published in final edited form as:
Expert Syst Appl. 2009 July 1; 36(5): 9250–9259. doi:10.1016/j.eswa.2008.12.050.

Associative Classification of Mammograms using Weighted Rules

Sumeet Dua, Ph.D.,
Department of Computer Science, Department of Computer Science, Louisiana Tech University, P.O. Box 10348, Ruston, LA 71270 and with the School of Medicine, LSU Health Sciences Center, 2020 Gravier Street, New Orleans, LA 70112. (Phone: 318-257-2830; fax: 318-257-4922; e-mail: sdua@coes.latech.edu)

Harpreet Singh, and
Department of Computer Science, Department of Computer Science, P.O. Box 10348, Louisiana Tech University, Ruston, LA 71270. (e-mail: hsi001@latech.edu)

H.W. Thompson, Ph.D.
Section of Biostatistics, School of Public Health and Departments of Ophthalmology, Medicine and Neuroscience, LSU Health Sciences Center, 2020 Gravier Street, New Orleans, LA 70112. (e-mail: hthomp2@lsuhsc.edu)

Abstract

In this paper, we present a novel method for the classification of mammograms using a unique weighted association rule based classifier. Images are preprocessed to reveal regions of interest. Texture components are extracted from segmented parts of the image and discretized for rule discovery. Association rules are derived between various texture components extracted from segments of images, and employed for classification based on their intra- and inter-class dependencies. These rules are then employed for the classification of a commonly used mammography dataset, and rigorous experimentation is performed to evaluate the rules' efficacy under different classification scenarios. The experimental results show that this method works well for such datasets, incurring accuracies as high as 89%, which surpasses the accuracy rates of other rule based classification techniques.

Keywords

Image classification; association rule; mammograms

I. Introduction

The design, development, and distribution of computer-aided image classification methods for enhanced clinical care and delivery by physicians has recently gained importance. This new found importance can be attributed, to a large degree, to the healthcare bottleneck in the delivery of services, which results when providers do not have adequate classification and delivery methods to handle the large amount of data made available by advancements in imaging. Data mining methods offer precise, accurate, and fast algorithms for such classification using dimensionality reduction, feature extraction, and classification routines. Consequently,

Correspondence to: Sumeet Dua.

Publisher's Disclaimer: This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final citable form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

association rule mining [6] has gradually emerged as an effective way to develop supervised classification frameworks for enhanced sensitivity and specificity in medical image classification.

In this paper, we present a new association rule based technique for medical image classification. We extract texture features from images to form association rules, which are then employed for classifier building and validation. Rigorous experimentation is performed, and we achieve superior classification accuracy on a previously studied mammogram dataset, demonstrating the efficacy of our technique.

The following sections of the paper are organized as follows. In Section II, we provide a brief introduction to related research in the field of medical image classification and to the use of association rules for image classification. In Section III, we outline the methodology in detail; in Section IV, we present the results of our experiments along with comparisons to existing techniques. In Section V, we present our conclusion.

II. RESEARCH AND RELATED BACKGROUND

The classification of medical images is a difficult and often computationally overwhelming task. Digitized medical images contain labels, noise, and irregularities that must be minimized before computational methods can be used to analyze them. Moreover, these images contain several correlated features (or embedded isomorphisms), often referring to indicators of a pathological or diagnostic state, which, when mined and exploited, can lead to superior classification.

Because automated methods can help physicians make better diagnostic decisions, much research has focused on content-based image retrieval (CBIR) in the medical domain ([10] presents an excellent survey of such methods). Medical image classification, often a prelude to a successful CBIR framework, has only recently attracted independent attention [6]. Mammogram classification has gradually emerged as an appealing domain for the evaluation of design and for the implementation of such classification framework, especially by those employing association rules [3]. For example, Ferreira et al [6] use wavelets to classify mammograms into different categories. Recently, Association Rules (ARs) have attracted interest as a means to achieve multi-class classification of mammograms [5], [11], [13]. Antonie et al [5] use association rules to classify digital mammograms into normal, benign, and malignant classes. However, their technique is time consuming, requires labeling of quadrants with abnormalities, and relies on very low support and confidence values, resulting in the generation of weak rules. Ribeiro et al. [11] use texture features and association rules to classify mammogram images. The major problems with this technique are the ad hoc segmentation of images, the time consuming discretization of segments, and the constraint of keeping the class label on right side of the rule. In [13], Yun et al. use a combination of association rules with a rough set theory for mammogram classification.

In addition to the above, Tseng et al. [12] apply multilevel association rules to hierarchically clustered objects from various images and perform object based segmentation on the image. This technique is not widely applicable for medical images because they usually contain few objects and because the objects may contain different abnormalities, i.e. different stages of cancer in breast mammograms. There are many other techniques present in literature, but, in the interest of space limitations, we restrict our discussion to the ones that are closely related to our area of research.

In this paper, we present a unique technique based on finding associations within an image and then relating those associations with images of the same class, to exploit both intra- and inter-image relationships for classifier building and testing. Our algorithm, Weighted Association

Rule based Classification (WAR-BC), uses a new rule weighting scheme, which can handle unbalanced class data easily. Unbalanced datasets are not an exception, but a norm in diagnostic classes of medical images. Rigorous experimentation is done to evaluate the efficacy of the methods, and the results are reported as measures of accuracy, sensitivity and precision, and confusion matrices.

III. PROPOSED METHODOLOGY

Our methodology consists of five major parts: data preprocessing, segmentation and feature extraction, data preparation, association rule mining, and classifier training and classification (Figure 1).

A. Data Preprocessing

Medical images present in the mammogram database are noisy, often containing deliberately inserted identifiable labels, which need to be eliminated prior to rule discovery and classification. Such artifacts can often lead to redundant and non-informative rules. Furthermore, the images are usually large in pixel-size (1024×1024), and most of the area consists of homogeneously colored background that gives limited useful information. For our studies, we have employed the breast mammogram database made available by the Mammography Image Analysis Society (MIAS)[2]. This dataset is freely available and has been widely used for mammography classification in previous studies (such as in [5]).

During the preprocessing stage, we present a method based on the connected component theory [7] to remove the labels from the image and crop the image to a relevant reduced size. Every intensity image (Figure 2(a)) is transformed into its binary format (Figure 2(b)). Once the image is transformed, we find the connected components present in the image, referring to the labels on the image, the breast part of the mammogram, the black background, and any scattered noise present. The connected component with the largest area is chosen as the segment of the mammogram image containing the breast and is then extracted from the original image (Figure 2(c)).

The segmented image does not have smooth boundaries since it is obtained from a binary image which has abrupt changes in pixel values from 0 to 1 and vice versa. The next step is to smooth the breast boundaries using the following pseudocode.

Algorithm—**B**order **S**moothing and **I**mage **Crop**ping (**BSI-Crop**)

Input: Segmented part of the mammogram, starting pixel and ending pixel points on each line of the segment, original image, number of rows *N*, intensity threshold $\boldsymbol{\alpha} = \mathbf{10}$, border threshold $\boldsymbol{\beta} = \mathbf{20}$

Output: Cropped segmented and border smoothened image called *New_cropped_segmented_image*

Method

```
For every line (row) $j$ in the segmented image $\forall j < N$
    Scan the line
    $Start\_pixel \leftarrow$ starting pixel position of the segment
    $End\_pixel \leftarrow$ ending pixel position of the segment
    Scan the same line on original image
    Read left from $Start\_pixel$
```

```
        New_Start_point(j) ← pixel position when five
        consecutive pixels have intensity < $\alpha$
        Read right from End_pixel
        New_End_point(j) ← pixel position when five
        consecutive pixels have intensity < $\alpha$
        Change every pixel value to the left of
        New_Start_point(j) and to the right of
        New_End_point(j) to zero
    End For
    Tip_Border_left ← minimum (New_Start_point (1-N))
    Tip_Border_Right ← maximum (New_End_point (1-N))

    Left_border ← Tip_Border_left −$\beta$
    Right_border ← Tip_Border_right + $\beta$
    New_cropped_segmented_image ← crop the border
    smoothened image into Left_border and Right_border
    limits
```

In this phase, the segmented image is scanned in a row-major format, and the starting and ending points of the segment on each line (row) are recorded (Figure 2(c)). The tip of left arrow shows the starting point and the tip of right arrow shows the ending point in each row. This starting and ending point tells us that where exactly the segmented breast part begins and ends in each row. The same points are recorded on the original unsegmented image, and this image is read, left from the starting point and right from the ending point (Figure 2(d)). A new starting point is marked on the image when five consecutive pixels have an intensity of less than a threshold (α=10) on the left side of the old starting point. Similarly, a new ending point is marked when five consecutive pixels have an intensity of less than the threshold on the right side of the old ending point. The threshold is selected after careful evaluation of the pixel intensities for the breast part and for the background. Every pixel to the left of the new starting point and to the right of the new ending point on this line is changed to zero. This procedure is applied to each of the following lines of the original unsegmented image, and a new-segmented and label free image with smooth boundaries is formed (Figure 2(e)). The new image formed will not always occupy the entire 1024×1024 image frame. Once the borders have been smoothened, the image is cropped in both directions to within the border threshold (β=20) pixels of the tip of the border (Figure 2(f)). As a final step of preprocessing we perform histogram equalization on the cropped images to improve their contrast (Figure 2(g)).

B. Segmentation and Feature Extraction

In the segmentation and feature extraction step, we first divide each image into several $n \times n$, non-overlapping segments. The size n is selected based on the expected granularity of the feature base. For our experiments, we select 20 as size n. We then segment the images into smaller blocks to capture the local relationships present in the image. Once the image has been segmented into blocks, eight texture features (Table 1) are extracted from each segment. Hence, each segment represents a feature vector length of eight. Each vector is given a unique Segment ID, which, in our case, is the number of the segment from which the features were extracted, e.g. *TID 1 (f1, f2, f3…………………..f8) and TID 2 (f1, f2, f3…………………..f8)*. We use eight of the fourteen Haralick [8] coefficients. The pseudocode for image segmentation and feature extraction is provided below.

Pseudocode for Segmentation and feature extraction

Algorithm: **SE**gmentation and **F**eature **Ex**traction (**SE-FEX**) divides the image into different non-overlapping segments, extracts features from these segments and arranges them in a Transactional Database

Input: Preprocessed images$\boldsymbol{I_1}$**,** $\boldsymbol{I_2}$,…..…, $\boldsymbol{I_N}$, Segment size **nxn**, set of discrete values a feature can take {ν_1, ν_2,……., ν_k}, number of Haralick texture features H

Output: Images $\boldsymbol{I_1}$, $\boldsymbol{I_2}$, ……., $\boldsymbol{I_N}$ in transaction database format where each transaction is a vector representing features extracted from each segment

Method

```
For every Image $I_j$ (1....$r$,1…$c$) $\forall j \in$ (1..$N$), $r$ shows
number of rows and $C$ number of columns
    Number of segments ($N_s$) $\leftarrow$ (r*c)/(n*n)
    For every segment $S_l$ $\forall l \in$ (1..$N_s$)
        $S_l(I_j[F_h]) \leftarrow \nu_t \forall\ t \le k, h \le H, j \le N, l \le N_s$
        extract features from the segment $S_l$
    Endfor
Endfor
```

For our experiment, we used the following notations: $P(i,j)$ is an entry in the co-occurrence/spatial dependence matrix with row number i and column number j; μ_x, and μ_y are the means for rows and columns, respectively, and σ_x, and σ_y are the corresponding standard deviation. Four possible angular nearest-neighbor distances can be used to calculate the co-occurrence matrix: 0°,45°,90°,135°. The value of all the features is calculated in these four directions, and the average value is represented as the actual value of a feature. We use the 1-nearest neighbor distance approach to calculate the co-occurrence matrix.

C. Data Preparation

Once the features have been extracted, we need to preprocess the data so that it can be used for association rule mining. During this phase, noise is removed by eliminating segments that contain 'NaN' values, because this information comes from the image background and is unemployable. Once this step is complete, we then perform Z-Score normalization on the data, which normalizes attribute A based on mean and standard deviation.

Z-Score normalization maps a value ν of A to ν′ using the formula: -

$$\nu' = \frac{\nu - \overline{A}}{\sigma_A}$$

where $\bar{A}$ is the mean, and σ_A is the standard deviation of the attribute.

Next, the continuous feature values are discretized into ten intervals using the equi-width binning method. The training data for each feature f_i over all the classes is combined to find the minimum (min_f_i) and maximum (max_f_i) values for feature f_i. These minimum and maximum values are then used to discretize the continuous values for f_i into ten equi-width intervals. The length of each interval is calculated using the formula l_i = (max_f_i−min_f_i)/10. The data in each bin is now uniquely labeled to substitute for this quantitative value.

D. Association Rule Mining

We represent each image formally as a vector of features. Let I denote the set of all images of a particular class, and let the feature take a set of k discrete values $\{\nu_1, \nu_2, \ldots\ldots, \nu_k\}$. We denote the value of feature F_i for image I_j by $I_j[F_i]$. For each image (I_j) and feature (F_i) and for each set X of images, $X \subseteq I$, $p \in \{l,2,\ldots,k\}$, define the present sets of $I_j \,|F_i\, I_j$ and X: $present(I_j|F_i,p)$:=φ if $\mathrm{I_j[F_i]} \neq \nu_p$ {i} if $\mathrm{I_j[F_i]} = \nu_p$, $present(I_j,p) := \{i|I_j[i] = \nu_p\}$ and $present(X,p) := \bigcup_{I \in X} present(I,p)$. Here $present(X,p)$ represent the frequent itemset X. We also define for some index, set P, and for some features, $\{F_i|i \in P\}$., the present set of X given $\{F_i \mid i \in P\}$ as follows: $present(X|\{F_i|i \in I\}, p) := \bigcup_{i \in P} \bigcup_{I_j \in I} present(I_j|F_i, p)$. For X $\subseteq I$, $p \in \{1,2,\ldots, k\}$, we define p-support of X to be, $s(X,p) := \#present(X \mid \{F_i \mid i \in P\}, p)$. For disjoint subsets X and Y of I, $p \in \{1,2,\ldots,k\}$, we write $X(p) \Rightarrow Y(p)$ to indicate that $X \cap Y = \varnothing$ and $present(X,p) \subseteq present(Y,p)$. We refer to $X(p) \Rightarrow Y(p)$ as an association rule. An association rule has a *support* $s(X(p) \Rightarrow Y(p))$, defined as, $s(X(p) \Rightarrow Y(p)) := \{i \mid present(X \mid \{F_i \mid i \in P\},p) \subseteq present(Y \mid \{F_i \mid i \in P\}, p)$. Finally, we define the *confidence* of $X(p) \Rightarrow Y(p)$ as follows:

$$c(X(p) \Rightarrow Y(p)) := [\, s(X(p) \Rightarrow Y(p))] / s(X, p).$$

These association rules are discovered for every image of every class using the *apriori* method [3]. Figure 3 shows a representative example of some of these rules. The first rule in the figure signifies that when value of 5th feature is 110, the value of 8th feature is 91, and the value of 6th feature is 92, then the value of the 1st feature is 94 with a support of 98 and a confidence of 100%. In the training phase (as described in the next section), these rules will be used to build an associative classifier, which is then used to classify the images in the test class.

E. Classifier Training and Classification

A fixed percentage of data is selected from each of the classes present in the training phase. Once the association rules have been discovered for training images, we combine the rules for the images in each class, based on the amount of training data employed. The rules for images belonging to the same class are combined into a class-level association rule set. From this set, we calculate the *frequency* of each rule for that class (the intra-class weight of a rule). The frequency information refers to the percentage of training images in a particular class in which the rule is present. It is possible that the same rule might be present in the images of other classes; therefore, we assign another frequency weight to each rule based on its presence across multiple classes (the inter-class weight of a rule). For this weight assignment, the class-level rule sets from all the classes are combined to form a global set, which has only unique rules present throughout different classes. Associated with each rule is the frequency of the rule in each class.

The weighing approach is described in the following pseudocode.

Pseudocode for Rule weighting

Algorithm: Rule **Weight**ing (R-Weight) is used to provide Horizontal and Vertical weights to every rule present in the training database

Input: Number of classes C, combined list of training rules for each class L_{C_i}, Number of rules in each class C_j, Total number of rules N

Output: Horizontal and Vertical weight matrices of rules

Method

```
For every rule $R_iC_j \ \forall i \le N, j \le C$
    $Frequency\ (R_iC_j) \leftarrow$ percentage of images in $C_j$, having $R_iC_j$
    $Hrizontal\_weight\ (R_iC_j) \leftarrow Frequency\ (R_iC_j) / \sum_{j=1}^{C} R_iC_j$
End For//Horizontal weighting complete
For every class $j \ \forall\ j \le C$
    $Rank\_rules_j \leftarrow$ Sort rules according to $confidence$ and then $support$ in each $confidence$
    For every rule $R_iC_j \ \forall\ i \le C_j, j \le C$
    $Vertical\_weight\ (R_iC_j) \leftarrow (C_j - Rank\_rules(R_iC_j))$
    End For
    $Normalized\_weight(R_iC_j) \leftarrow$ normalize vertical weights in the range 0–1
End For
```

Two rule measures, *horizontal* weight and *vertical* weight, are assigned to each rule. Horizontal weight is calculated based on the *frequency* of a rule across the different classes. Once the frequency has been calculated for each class, it is divided by the sum of all the frequencies of that rule over different classes to get a relative frequency. This relative frequency is defined as the horizontal weight of a rule for that particular class. For example, suppose there are 100 unique rules present in total across all the classes. Say rule $\boldsymbol{R}_1$ is present in 30% of training images in class 1, 70% in class 2, and 20% in class 3. Then the frequency of $\boldsymbol{R}_1$ is, in class 1, 0.30, in class 2, 0.70, and in class 3, 0.20. The relative frequency/horizontal weight of $\boldsymbol{R}_1$ for class 1 is 0.30/(0.30+0.70+0.20)=0.25, for class 2 is 0.70/1.2=0.583, and for class 3 is 0.20/1.2=0.166.

For vertical weighing, rules are sorted in every class (in class-level rule set) according to the decreasing order of the confidence value of a rule. It should be noted that more than one rule may have the same confidence value. To reduce this occurrence, the rules are further sorted, within each confidence, according to decreasing order of the support value. The rule with the highest confidence/support pair gets the first rank and the highest weight. The highest weight is the number of rules present in that class. For example, if 80 rules are present in a class, then the rule with rank 1 is assigned a weight of 80. The second ranked rule is assigned a weight of one less than the first ranked rule, which, in our example, is 79. The third ranked rule is assigned a weight, of two less than the first ranked rule, and so on. The last ranked rule is assigned a weight of 1. These weights are then normalized in the 0–1 range.

Finally, we take the number of items in a rule into account. The actual weight of the rule is the sum of its horizontal and vertical weight multiplied by the *cardinality* of rule (the number of items present in the rule). If C = Total No. of classes, N = Total global rules, CD_i = Cardinality of rule $\boldsymbol{R}_i$, $\forall\ i \le N$

C_j = Rules in class j, $\forall\ j \le C$

$\boldsymbol{H_jR_i}$ = Horizontal weight of Rule $\boldsymbol{R_i}$ for class

$\boldsymbol{j}, \forall\ (\boldsymbol{i} \le \boldsymbol{N}, \boldsymbol{j} \le \boldsymbol{C})$

$V_j R_i$ = Vertical weight of Rule R_i for class

$\boldsymbol{j}, \forall\ (\boldsymbol{i} \le \boldsymbol{C_j}, \boldsymbol{j} \le \boldsymbol{C})$

Q = No. of rules from query Image which match with global set of rules (N)

Then the weight of a rule R_i for class C_j is defined by the formula: -

$$W_j R_i = (H_j R_i + V_j R_i) \times CD_i$$

Classification: The first step in the classification of a query image is to generate association rules for the image using the same confidence and support as the images in the training data. Then, each rule from the query image is taken and matched with the global rule set to find its horizontal weight. The rule is then matched with class-level rule sets to find its vertical weight in each class. A match is defined as the matching of all the items in a rule body, both on the left and right hand side of the rule. To mark a match, the horizontal and vertical weight of the rule is added and then multiplied by the number of items present in the rule. The resultant is called the score of the rule. The procedure is repeated for each rule in the query image. Finally, the scores of the matching rules are added on a class-by-class basis and a cumulative sum is calculated for each class. The image is classified to the class with the highest cumulative sum. The formula for sum of all the rules for class $\boldsymbol{C_j}$, is defined by the formula: -

$$Tot_j = \sum_{i=1}^{Q} (W_j R_i)$$

Then the output label (predicted class) can be decided by the formula: -

$$ClassLabel \Leftarrow \arg \max_{j=1}^{C} (Tot_j)$$

$\boldsymbol{N_i}$ = No. of query images from class i

$\boldsymbol{Q_i}$ = No. of images in class i which were correctly predicted by the classifier. Then accuracy for class i can be defined by:

$$CAcc_i = \frac{Q_i}{N_i}$$

And the total accuracy is given by: -

$$Accuracy_{total} = \frac{\sum_{i=1}^{C} Q_i}{\sum_{i=1}^{C} N_i}$$

The classification mechanism is shown in Figure 4.

IV. EXPERIMENTS

We tested our methodology WAR-BC on the Mammography Image Analysis Society (MIAS) [2] mammogram database. This database is commonly used for Mammography classification [5]. A total of 322 images from three classes: normal, benign, and malign, are present in the database [5],[2]. Among these images, 208 belong to the normal class, 63 belong to the benign class, and 51 belong to the malign class.

We compare our results with those of [5], [4], and [13]. In [4] the authors use two techniques to classify the mammograms. These are: a three layer (input, hidden, and output) back propagation neural network (BPNN) and an association rule based classifier (ARC-AC). In BPNN, the input layer has 69 nodes, the hidden layer has 10 nodes, and the output layer has 1 node. The node of the output layer classifies a query image. ARC-AC uses association rules to classify images, with an initial support set at 10% and an initial confidence set at 0%. The confidence is increased in the tuning phase, depending on the accuracy of the training data. ARC-BC [4] is an improvement over ARC-AC, where the association rules are formed for each class separately, rather than for the entire dataset, with one support and one confidence. The results using ARC-BC are better than the results using ARC-AC, and hence, we only compare our results with ARC-BC. In [13] an associative classifier is combined with a rough set theory to build a hybrid classifier called JAC. For further information on the workings of these classifiers, we recommend [5], [4], and [13]. The results of our comparison are presented in Table 2. As shown in the table, our approach (WAR-BC) demonstrates better accuracy than previous approaches.

A. Associative Classification

We illustrate our results using three percentage sets of training/testing data 70/30, 80/20, and 90/10. We keep the support value low (4%) and the confidence level high (90%). Then, we mine all the classes for association rules with the same support and confidence.

Our weighting schema can easily handle multiple classes with unbalanced data. This ability to handle unbalanced data is an important improvement to existing association rule techniques which are sensitive to unbalanced data (ARC-AC). Further, we use true association rules, rather than class constrained rules, where the class is kept on the right-hand side of the rule like the ones used by ARC-AC and ARC-EC. Consider a rule $A,B,C,D \Rightarrow C_i$. This rule could be present in both C_1 *and* C_2, and hence, could cause confusion for the classifier. Looking at this rule more carefully A, B, C, D is actually a frequent itemset which might be present in more then one class. However, a rule $A,B \Rightarrow C, D$ or $A \Rightarrow B,C,D$ might be present in only one of the two classes C_1 *and* C_2. Hence, this rule can be used as an important discriminator for classification purposes. Using experimentation, we demonstrate this property and show that classification accuracy with class-constrained rules is lower than non-class constrained rules.

We perform ten cross validations to compare to the results of [5], [4], and [13]. Table 2 shows the comparison of the results obtained using our technique with the results obtained using other methods.

Results for classification with class-constrained rules (WAR-CCBC) are presented in second column of Figure 5(a). In BP NN, ARC-BC, and JAC, results for 90% training and 10% testing data are shown (Table 2). The average accuracy of our technique for the same training/testing data is almost 10% higher than ARC-BC, 8% higher than BP NN, and 12% higher than JAC. Even with WAR-CCBC, our accuracy is higher than all the other techniques. In addition, we experimented with less training data (70% and 80%) and obtained an average accuracy (over

10 splits of data) higher than that of BP NN, ARC-BC, and JAC. Figure 5(b) shows these results.

Further we also calculate the precision and sensitivity as follows.

$$\mathrm{Pr}ecision=\frac{TP}{TP+FP} \qquad Sensitivity/\mathrm{Re}call=\frac{TP}{TP+FN}$$

In this case, TP = images which are normal and are labeled normal by the classifier; FP = images which are abnormal, but are labeled normal; TN = images which are abnormal and are labeled abnormal, and FN = images which are normal, but are labeled abnormal. Figure 6 shows the graphs for the Precision and Sensitivity of three pairs (70/30, 80/20, and 90/10) over ten splits of the data. The graphs show that the precision and sensitivity values are fairly high for all pairs of classification. For 90% training data, the average precision is 91.83%, and the average sensitivity is 96.36%.The graphs show that the precision and sensitivity values are fairly high for all pairs of classification. For 90% training data, the average precision is 91.83%, and the average sensitivity is 96.36%. Figure 7 shows the confusion matrix for the best-case scenario. We can see that none of the normal image is misclassified into other classes, and only one image each from the benign and malign classes are misclassified into the normal class.

From our experiments, we also note that the misclassification is an image issue, and not an issue with the classifier. Regardless of the percentage of data taken for classifier training, some images are repeatedly misclassified into the same class. This directed us to look for classification separately in different density mammograms, such as fatty, glandular, and dense tissue. Details about these experiments will be provided in Section IV E.

B. One vs. All classification

Additionally, we perform experimentation with a different classification model, a multiple classification model vs. the all classifier model previously used. In this classifier, we build a separate binary classifier for each of the three classes. While building the classifier for class Normal, the rules for class Normal is kept as such, but the rules from class Benign and class Malignant are combined to form the second class, named Other. The horizontal and vertical rule weighting procedure is the same as described in Section III E, except that instead of three classes, we now have two. For training the classifier for class Benign, the rules from class Normal and Malignant are combined to form class Other. For training class Malignant, the rules from class Normal and benign are combined to form class Other. Each Classifier is trained to classify an incoming image only into its own class by giving it a certain score. When a new image comes for classification, every classifier gives it a score, and the image is classified into the class with the highest score. For example, classifier Normal gives the new image a score of 0.8, classifier Benign gives it a score of .04, and classifier Malignant gives the image a score of .99. Next, the image is classified to class Malignant as the classifier for this class gives it the highest score for its class. The third column of Figure 5(a) shows these results.

C. Fuzzy-K Nearest Neighbor Classification

In our previous experiments, we employed the Harlick feature for the extraction of association rules and then used those rules for classification purposes. While Harlick features possess discriminative power when applied independently, the association rules are expected to boost classification validity by introducing isomorphisms. Independent evaluation of the efficacy of these features and comparison of the evaluation to our associative classification-based results will provide us with an empirical measure of improvements that we have been able to achieve with association rule discovery.

As a preliminary step, the data representation is modified for use with FKNN based classification. A new data matrix of Harlick features is extracted from segments formed where each row represents an image and each column represents a set of features extracted from each segment. If an image has n segments, then the total number of columns for that image would be $n \times 8$ (because we are experimenting with eight features). Class labels are included for training and excluded for testing.

To make an accurate comparison, we take the same amount of data for training (90%) and testing (10%) as we did for the previous experiments. The results are shown in Table 2. Furthermore, to check the efficacy of the association rules extracted by our algorithm, we do another set of experiments with F-KNN, in which we provide the association rules for an image as input to F-KNN (named this experiment F-KNN2) instead of using raw Haralick features as input.

Class-level association rule sets (see Section III E) were combined to form an aggregate rule set (Figure 8) over a complete database. Unique rules from this aggregate rule set are selected and arranged in a data matrix. Each row in the data matrix represents an image and two consecutive columns represent the support and confidence of a single rule. Since not all the training images have every unique rule, for an image (row) which did not have a particular rule from the aggregate rule set, the corresponding columns were set to zero. The columns containing the confidence and support for a rule R_i can be located by the function (i−1)*2+1. So the rule R_{30} confidence and support for all the images is found in columns 59 and 60, respectively. Figure 5(c) shows the results for these experiments.

D. Hierarchical Agglomerative Clustering using Induced Rules

The associative features that we have extracted in the previous steps have discriminative power that can be evaluated by an independent feature and rule classification algorithm. In pursuit of this evaluation, we have applied a hierarchical agglomerative cluster algorithm (PNC2) that induces rules in the context of direct generation of “if-then” rules for classification tasks [1]. We have performed rigorous experimentation to evaluate the efficacy of the association rules using this method. Initially, the same input data matrix as F-KNN is given as input to PNC2 for learning. The model training time with this data matrix is prohibitively large (25 hours on a single processor AMD opteron 2.39 GHz Machine). However, the testing accuracy was only 53.13%. In an attempt to boost the model training runtime for the 10-fold cross validation, we take four consecutive segments (in row-major format) for each image from original data matrix and average the features to extract the derived aggregated value. This reduces the data size and time for model learning significantly without compromising the training accuracy. The results are shown in Table 2. For further information about the working of FKNN and PNC2 we recommend [1] and [9].

E. Discriminative Classification of Domain Classes

In this set of experiments, we use association rule based classification to classify normal vs benign vs malignant cases, based on the tissue densities. Of the 322 total images, 108 are fatty, 101 are glandular, and 112 are dense. Images in each density class are further divided according to the abnormality present into: Normal, Benign, and Malignant classes. For the fatty dataset, there are 67 Normal, 23 benign, and 18 malignant; for the glandular dataset, there are 65 normal, 20 benign, and 16 malignant; and for the dense dataset, there are 76 normal, 20 benign, and 16 malignant images. Classifier training/testing is performed separately for each density class. Again, we run experiments with three different pairs of data (70/30, 80/20, and 90/10). Here, we perform a 5-fold cross validation. The average accuracy for density in the class fatty over 5 runs (Figure 9) for 70/30 data pair is 77%, for 80/20 is 86.84%, and for 90/10 is 95%. Accuracy in the glandular class is 85%, 84.38%, and 88%. For the dense class, accuracy is 84.23%, 87.7%, and 86.6%.

V. CONCLUSION

Mammogram classification is an overarching image classification problem. In this paper, we have presented a novel framework for the improvement of mammogram classification, which includes a new preprocessing methodology for segmenting, a unique associative rule discovery based algorithm for classification and an evaluation of the efficacy of raw and derived features using fuzzy K-nearest neighbor and agglomerative clustering of associative features. In addition, we have presented a novel framework for the weighting of the rules based on rule presence in different classes to employ intra-class and inter-class similarities. Our approach eliminates the needs of using class constraining rules, which boosts the effectiveness of the discovered rules employed as discriminatory features. Our detailed experimental results demonstrate that our technique is superior to existing techniques for mammogram classification using association rules. The expert classifier results demonstrate the robustness of the derived feature vectors suggesting opportunities for future elucidation and refinement of this work.

Acknowledgments

This research project was made possible by NIH Grant Number P20 RR16456 from the INBRE Program of the National Center for Research Resources. Additional support was made available by NSF Grant Number 0454355. Its contents are solely the responsibility of the authors and do not necessarily represent the official views of NIH or NSF.

References

1. http://www.newty.de/pnc2/PNC2.html.
2. MIAS Database. The PCCV Project: Benchmarking Vision Systems. http://peipa.essex.ac.uk/info/mias.htm
3. Antonie ML, Zaiane OR, Coman A. Application of Data mining Techniques for Medical Image Classification. MDM/KDD 2001:94–101.
4. Antonie, ML.; Zaiane, OR.; Coman, A. LNICS. Vol. 2797. MMCD, Berlin/Heidelberg: Springer; 2003. Associative classifiers for medical images; p. 68-83.
5. Agrawal, R.; Imielinski, T.; Swami, AN. Mining association rules between sets of items in large databases. Proceedings of the 1993 ACM SIGMOD ICMD, ACM; Washington, D.C.. 1993. p. 207-216.
6. Ferreira, CBR.; Borges, DL. Automated mammogram classification using a multiresolution pattern recognition approach. Proceedings of XIV Brazilian Symposium on Computer Graphics and Image Processing, IEEE; Florianopolis, Brazil. 2001. p. 76-83.
7. Haralick, RM.; Shapiro, LG. Computer and Robot Vision, 1. Addison-Wesley; 1992. p. 28-48.
8. Haralick, RM.; Shanmugam, K.; Dinstein, I. Textural features for image classification. IEEE Trans, on SMC ; Piscataway, New Jersey. 1973. p. 610-621.
9. Keller JM, Gray MR, Givens JA. A fuzzy k-nearest neighbor algorithm. IEEE Transaction on Systems, Man and Cybernetics 1985;15 (4):580.
10. Muller H, Michoux N, Bandon D, Geissbuhler A. A review of content-based image retrieval systems in medical applications-clinical benefits and future directions. International Journal of Medical Informatics 2003;73:1–23. [PubMed: 15036075]
11. Ribeiro, MX.; Traina, AJM.; Balan, AGR.; Traina, C., Jr; Marques, PMA. SuGAR: A framework to support mammogram diagnosis. IEEE CBMS 2007; Maribor, Slovenia. 2007. p. 47-52.
12. Tseng, SV.; Wang, M-H.; Su, J-H. A new method for image classification by using multilevel association rules. Presented at ICDE 05; Tokyo. 2005. p. 1180-1187.
13. Yun J, Zhanhuai L, Yong W, Longbo Z. Joining associative classifier for medical images. HIS. 2005

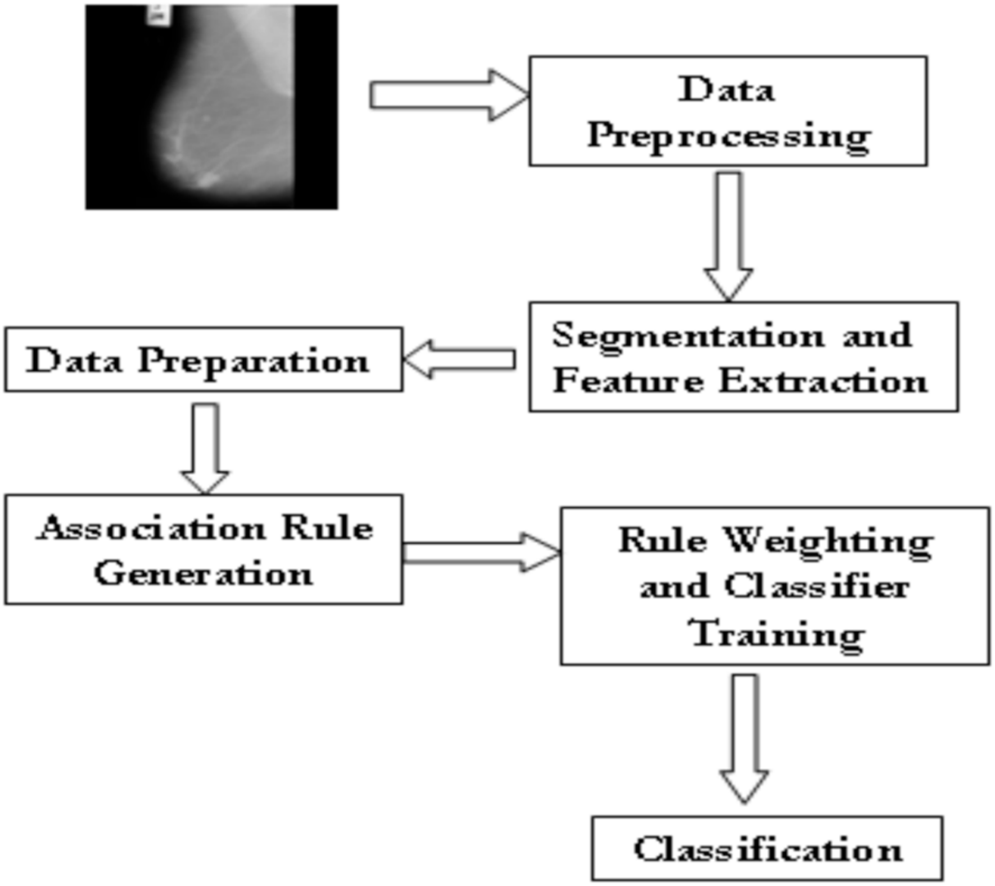

Data Preprocessing
Segmentation and Feature Extraction
Data Preparation
Association Rule Generation
Rule Weighting and Classifier Training
Classification

Fig. 1.

Proposed Methodology: This figure explains the overall computing framework followed in our approach. The methodology is described in detail in the paper.

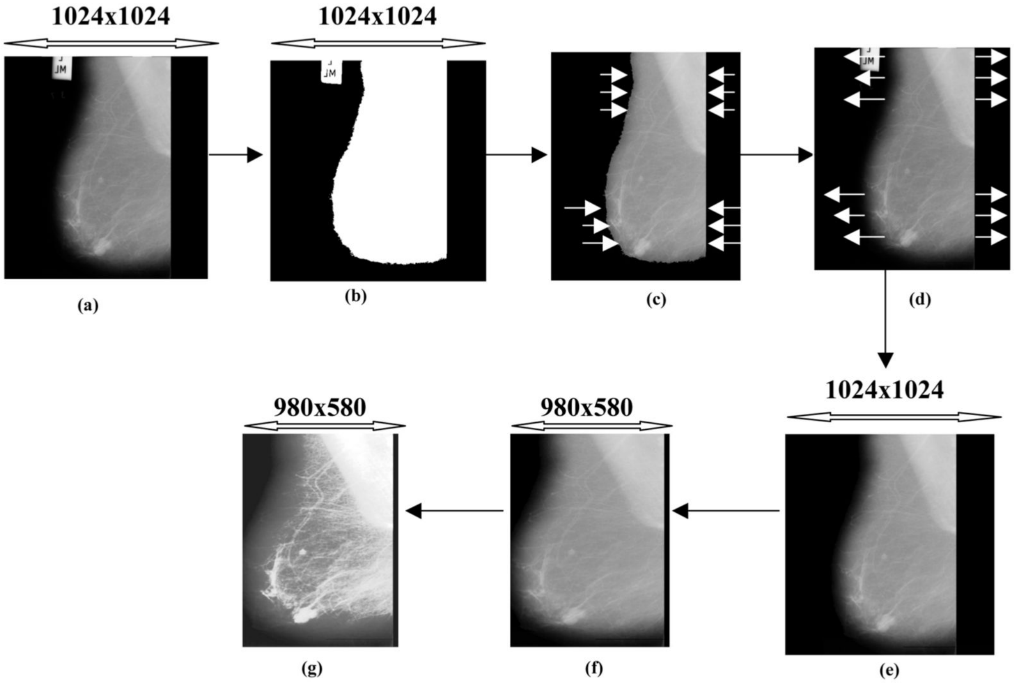
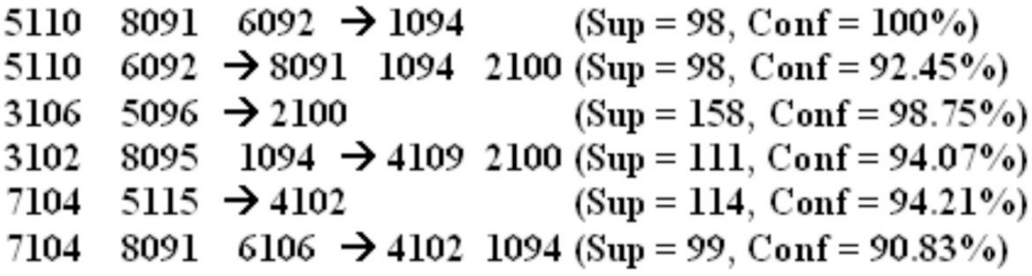


Fig. 2.

(a) Original Image (b) Binary Representation (c) Segmented Breast Part (d) Unsegmented Image with Starting and Ending Point Information (e) Segmented and Border Smoothed Image (f) Cropped Image (g) Normalized Image: This figure presents a diagrammatic description of the various image preprocessing and segmentation procedures employed to prepare the image for the following feature extraction procedures.



```
5110   8091   6092 → 1094          (Sup = 98, Conf = 100%)
5110   6092 → 8091   1094   2100   (Sup = 98, Conf = 92.45%)
3106   5096 → 2100                 (Sup = 158, Conf = 98.75%)
3102   8095   1094 → 4109   2100   (Sup = 111, Conf = 94.07%)
7104   5115 → 4102                 (Sup = 114, Conf = 94.21%)
7104   8091   6106 → 4102   1094   (Sup = 99, Conf = 90.83%)
```

Fig. 3.

Representative Examples of Rules Extracted: The figure represents some sample rules that are extracted using the proposed approach. The first column in each row is an assigned rule-id. The antecedents of rules are before the right arrow and the descendents are following the arrow. Note that some rules have one (level-1) antecedent and some have multiple antecedents (level-2, level-3, etc.). The features participating in rules are represented by numerical labels as a result of discretization of rules. The calculated support and confidence measures of rules are indicated in brackets after the rule.

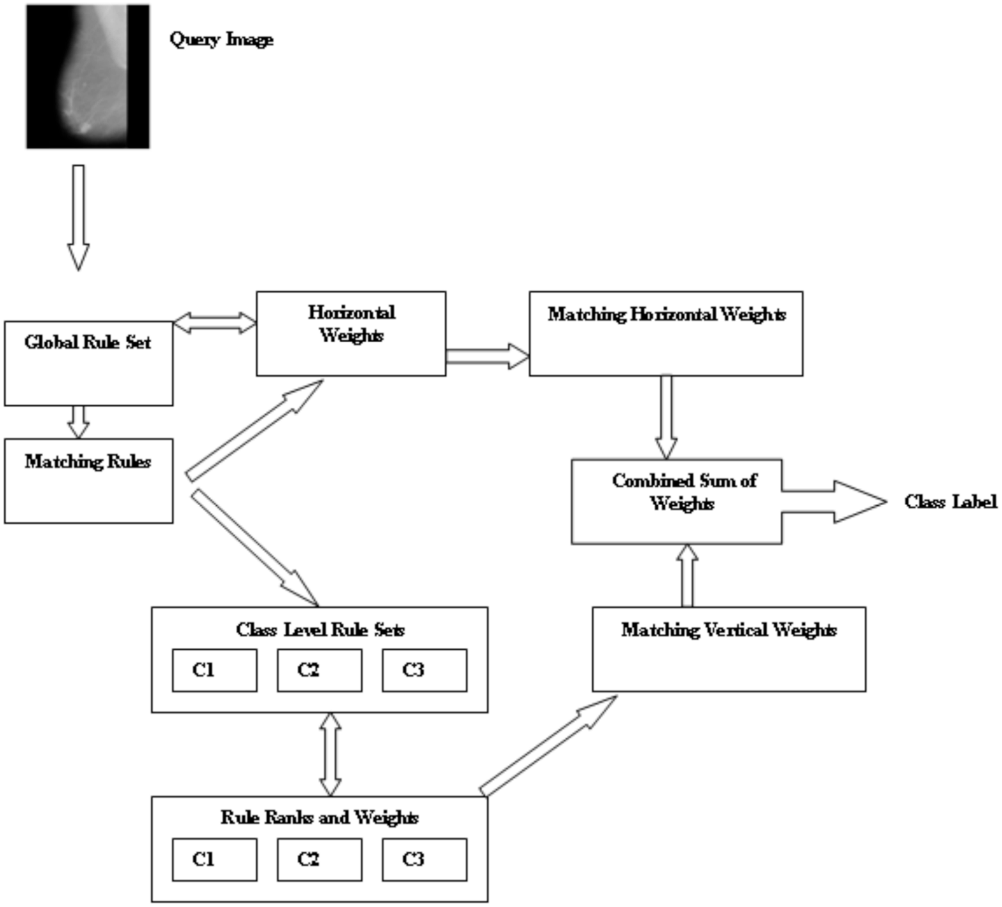
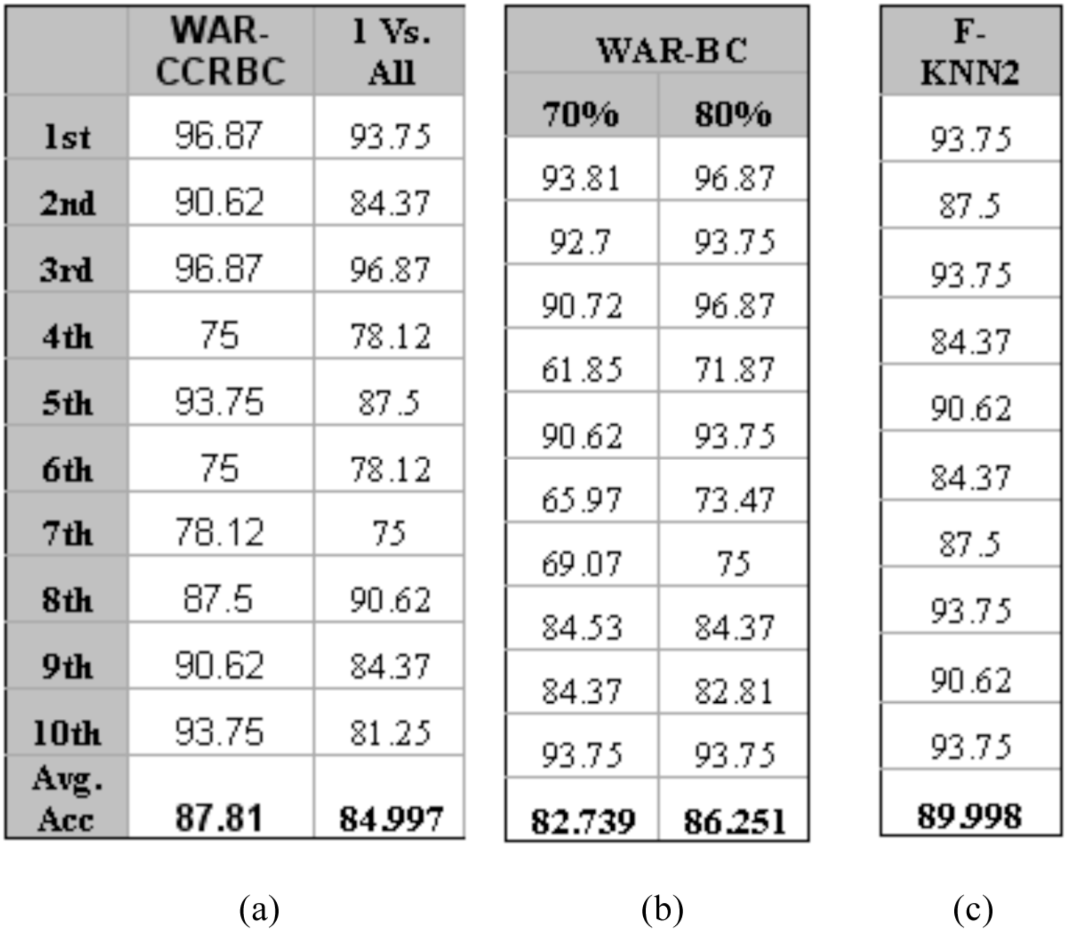


Fig. 4.

Classification Mechanism of the System: The figure explains how a query image with an unknown class label is assigned to a class by comparison with existing rule sets during Associative Classification. Depending on the combined sum of weights for each class, the image is assigned to a particular class.



(a)

	WAR-CCRBC	1 Vs. All
1st	96.87	93.75
2nd	90.62	84.37
3rd	96.87	96.87
4th	75	78.12
5th	93.75	87.5
6th	75	78.12
7th	78.12	75
8th	87.5	90.62
9th	90.62	84.37
10th	93.75	81.25
Avg. Acc	**87.81**	**84.997**

(b)

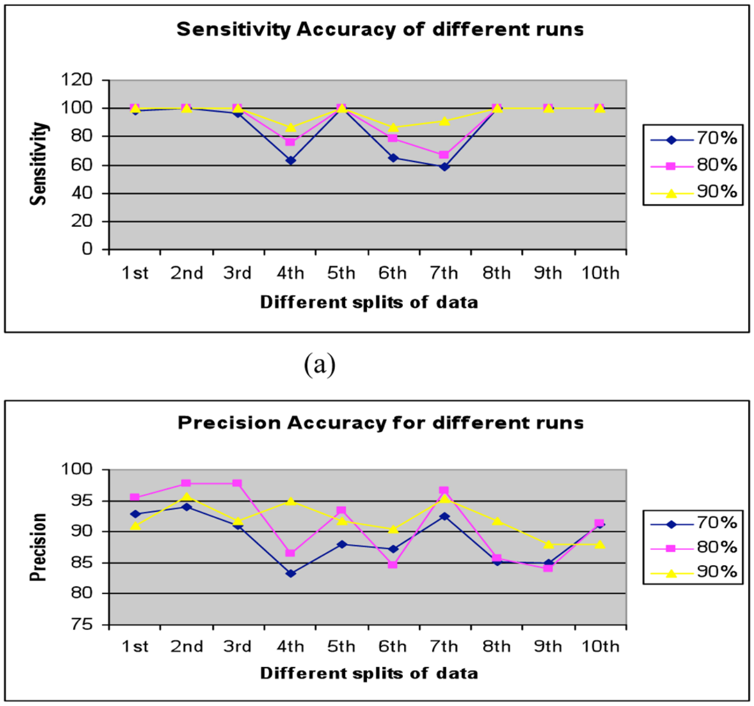
WAR-BC	
70%	**80%**
93.81	96.87
92.7	93.75
90.72	96.87
61.85	71.87
90.62	93.75
65.97	73.47
69.07	75
84.53	84.37
84.37	82.81
93.75	93.75
82.739	**86.251**

(c)

F-KNN2
93.75
87.5
93.75
84.37
90.62
84.37
87.5
93.75
90.62
93.75
89.998

Fig. 5.

(a) Accuracies for Class Constrained Rules and Multiple One vs. All Classifiers (WAR-CCBC and 1 Vs. All) respectively (b) Accuracies for 70% and 80% Training Data (c) Accuracies for using Association Rules as Input Data: The figure explains the accuracy of our proposed method when compared with other approaches for same selection of training and testing datasets in (a) and (c) and different training and testing pairs in (b). A 10 cross validation is performed to reduce any selection bias.

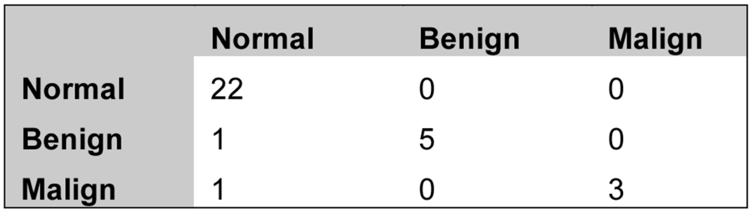


(a)

(b)

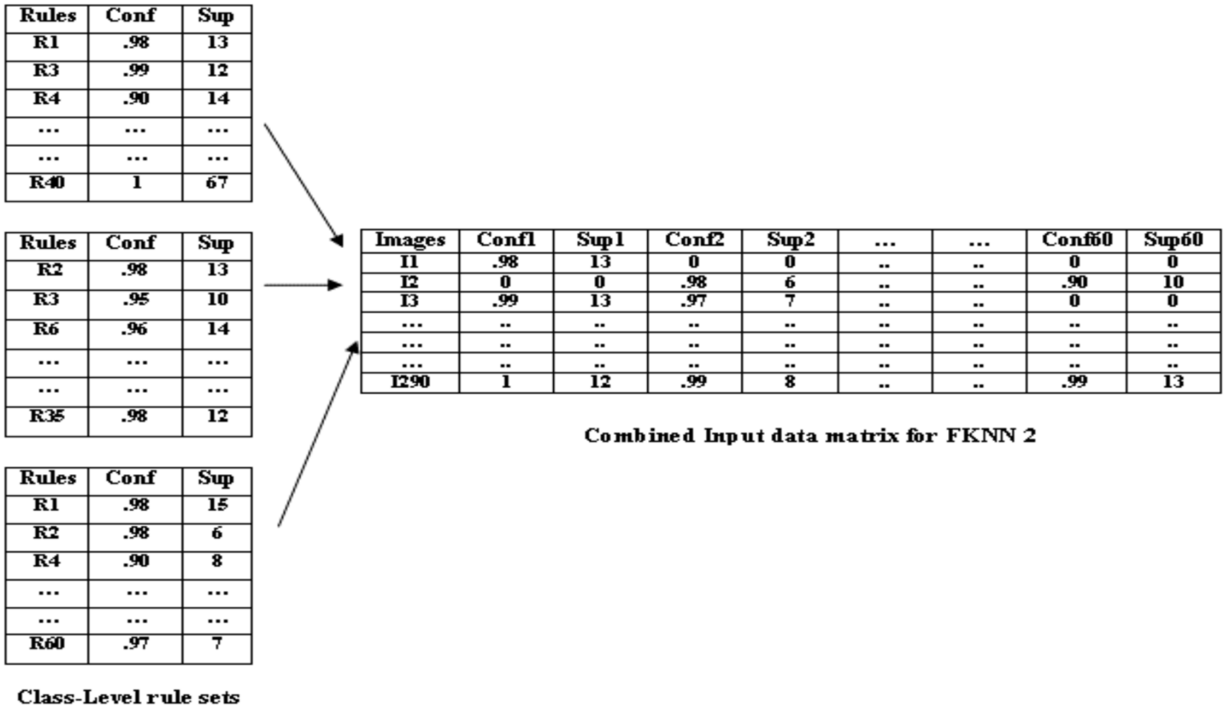
Fig. 6.

(a) Precision over 10 Runs (b) Recall over 10 Runs: The figures explain the change of Precision (a) and Recall (b) with different percentages of training versus testing data. The experiments are repeated ten times each (10-cross validation) to remove any selection bias that might result due to randomly selecting a fixed percentage of training images. Note the boost in accuracy with a high percentage of training data. The fluctuation owing to different runs is reduced with greater percentage for training.



	Normal	Benign	Malign
Normal	22	0	0
Benign	1	5	0
Malign	1	0	3

Fig. 7.
Confusion Matrix for the Best Case Scenario: The figure shows the confusion matrix for three classes considered for classification. The number indicates the number of cases reported. The horizontal rows are true classes, and the vertical columns are reported classes. The Benign and Malign classes are accurately separated from each other, and class Normal causes most misclassification.



Rules	Conf	Sup
R1	.98	13
R3	.99	12
R4	.90	14
...	...	...
...	...	...
R40	1	67

Rules	Conf	Sup
R2	.98	13
R3	.95	10
R6	.96	14
...	...	...
...	...	...
R35	.98	12

Rules	Conf	Sup
R1	.98	15
R2	.98	6
R4	.90	8
...	...	...
...	...	...
R60	.97	7

Class-Level rule sets

Images	Conf1	Sup1	Conf2	Sup2	...	...	Conf60	Sup60
I1	.98	13	0	0	..	..	0	0
I2	0	0	.98	6	..	..	.90	10
I3	.99	13	.97	7	..	..	0	0
...	..	..	..	..	..	..	..	..
...	..	..	..	..	..	..	..	..
...	..	..	..	..	..	..	..	..
I290	1	12	.99	8	..	..	.99	13

Combined Input data matrix for FKNN 2

Fig. 8.

Combining Class Level Rule Sets into Aggregate Set Matrix: To evaluate the efficacy of association rule descriptors as discriminatory features, the data is rearranged in a feature matrix as shown in the figure. This feature set is given as an input to the classifier for classifier building and testing.

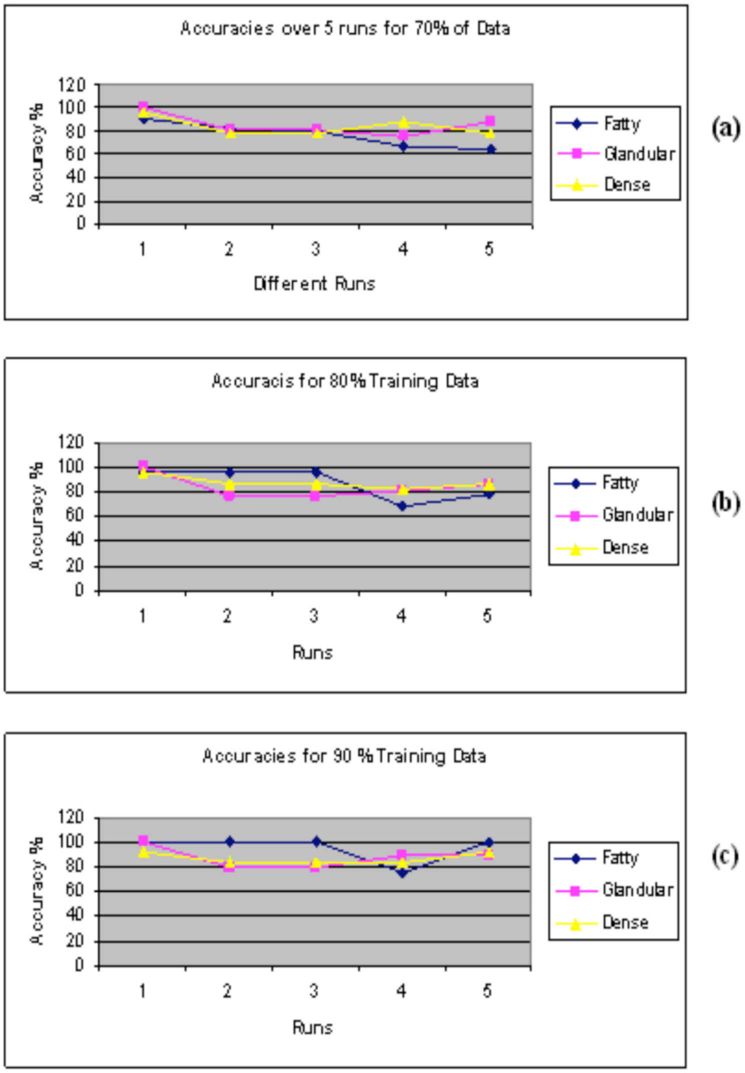


Fig. 9.

Domain Specific Classification of Mammograms for Different Training/Testing Percentage Pairs (a) 70/30 (b) 80/20 (c) 90/10: To evaluate the impact of tissue density we ran experiments separately for each density class of Fatty, Glandular, and Dense. Here X-axis represents the five runs, while the Y-axis represents the accuracy rate.

Table 1

Harlick Texture Features for Feature Representation

Feature Label.	Feature	Calculation
F1	Energy	$\sum_{i=0}^{n}\sum_{j=0}^{n}\{p(i,\ j)\}^2$
F2	Contrast	$\sum_{i=0}^{n}\sum_{j=0}^{n}(i-j)^2 p(i,\ j)$
F3	Local Homogeneity	$\sum_{i=0}^{n}\sum_{j=0}^{n}\frac{p(i,\ j)}{1+(i-j)^2}$
F4	Correlation	$\sum_{i=0}^{n}\sum_{j=0}^{n}((ij)p(i,\ j)-\mu_x\mu_y)\Big/\sigma_x\sigma_y$
F5	Entropy	$-\sum_{i=0}^{n}\sum_{j=0}^{n}p(i,\ j)\log p(i,\ j)$
F6	Cluster Shade	$\sum_{i=0}^{n}\sum_{j=0}^{n}(i-M_x+j-M_y)^3 p(i,\ j)$
F7	Information measure of correlation	H_{XY}-$H_{XY1}/max(H_X,H_Y)$
Ft	Maximum Probability	$\max_{i,j}P(i,j)$

where,

$$M_x=\sum_{i=0}^{n}\sum_{j=0}^{n}ip(i,j)\qquad M_y=\sum_{i=0}^{n}\sum_{j=0}^{n}jp(i,j)$$

$$P_x=\sum_{j=0}^{n}p(i,j)\qquad P_y=\sum_{i=0}^{n}p(i,j)$$

$$H_x=-\sum_{i=0}^{n}P_x(i)\log P_x(i),H_y=-\sum_{j=0}^{n}P_y(j)\log P_y(j)$$

$$Hxy1=-\sum_{i=0}^{n}\sum_{j=0}^{n}P(i,j)\log\{P_x(i)P_y(j)\}$$

Table 2

Comparison of different techniques with our technique

	BP NN	JAC	ARC-BC	F-KNN	PNC2	WAR-BC
1st	96.87	69.342	80	59.37	53.13	93.75
2nd	90.62	86.373	93.33	46.87	56.25	90.62
3rd	90.62	77.586	86.67	56.25	62.5	93.75
4th	78.125	72.912	76.67	71.87	62.5	84.37
5th	81.25	78.224	70	53.12	59.37	93.75
6th	84.375	77.055	76.67	75	20	81.25
7th	65.625	77.691	83.33	65.25	68.75	90.62
8th	75	73.752	76.67	56.25	20	90.62
9th	56.25	82.123	76.67	53.12	68.75	87.5
10th	93.75	79.819	83.33	59.37	12.5	90.62
Avg. Acc	**81.2485**	**77.4877**	**80.334**	**59.647**	**48.375**	**89.685**