An Active Learning Approach to *Egeria densa* Detection in Digital Imagery

Huan Liu and Amit Mandvikar

Department of Computer Science and Engineering

Arizona State University

Tempe, AZ 85287-5406

{hliu, amitm}@asu.edu

Patricia G. Foschi

Romberg Tiburon Center for Environmental Studies

San Francisco State University

Tiburon, CA 94920

tfoschi@sfsu.edu

Keywords: Active Learning, Image Data, Classification, Ensemble Methods, Unlabeled Data

Abstract

This chapter focuses on the development of an active learning approach to an image mining problem for detecting *Egeria densa* (a Brazilian waterweed) in digital imagery. An effective way of automatic image classification is to employ learning systems. However, due to a large number of images, it is often impractical to manually create labeled data for supervised learning. On the other hand, classification systems generally require labeled data to carry out learning. In order to strike a balance between the difficulty of obtaining labeled images and the need for labeled data, we explore an active learning approach to image mining. The goal is to minimize the task of expert labeling of images: if labeling is necessary, only those important parts of an image will be presented to experts for labeling. The critical issues are: (1) how to determine what should be presented to experts; (2) how to minimize the number of those parts for labeling; and (3) after a small number of labeled instances are available, how to effectively learn a classifier and apply it to new images. We propose to use ensemble methods for active learning in Egeria detection. Our approach is to use the combined classifications of the ensemble of classifiers to reduce the number of uncertain instances in the image classification process and thus achieve reduced expert involvement in image labeling. We demonstrate the effectiveness of our proposed system via experiments using a real-world application of *Egeria* detection. Practical concerns in image mining using active learning are also addressed and discussed.

1 Introduction

Multimedia content is rapidly becoming a major research area of data mining. One task of mining information from digital imagery is to discover patterns and knowledge from images for the purpose of classification. The specific problem we address here is the detection of Brazilian waterweed (*Egeria densa*) in images. *Egeria densa* is an exotic submerged aquatic weed causing navigation and reservoir-pumping problems in the Sacramento-San Joaquin Delta of Northern California. As a part of a control program to manage *Egeria*, it is necessary to map the areal extent of *Egeria* in scan-digitized aerial images. The detected *Egeria* regions are calculated and compared with previously detected regions. The analyzed results are then used to determine control strategies and form new solutions. The *Egeria* detection problem can be abstracted to one of classifying massive data without class labels. Relying on human experts to classify *Egeria*, or labeling each region with classes (*Egeria* or not) is not only time-consuming and costly, but also unreliable if the experts are overburdened with too many minute and routine tasks. Massive manual classification becomes impractical when images are complex with many different objects (e.g., water, land, *Egeria*) under varying picture-taking conditions (e.g., deep/shallow water, sun glint). The main objective of the work is twofold: (1) to learn to automatically detect *Egeria,* and (2) to relieve experts from going through all the images and identify regions where *Egeria* exists in each image.

The following desiderata for an image classification system present a unique challenge to data mining research for novel solutions.

1. *Reduced expert involvement.* Classification algorithms that require less expert involvement are essential in real-world applications, because human involvement in decision making forms the most serious bottleneck for efficient processing.

2. *Fewer labeled training images.* Labeled data is required to train automatic classifiers in a supervised fashion. The only source for such labeled data is to label data manually by experts, which is tedious, slow and expensive. Consequently, it is sensible to reduce the

number of images to be labeled. The reduced number of images, however, increases the difficulty for learning.

3. *Classification performance.* An image classification system can produce *certain* and *uncertain* classifications. Uncertain classifications require the intervention of human experts. Reducing the number of uncertain classifications translates directly to the reduction of expert involvement. In addition, classifications deemed certain should also be *correct*. Standard performance measures for detection problems such as accuracy, precision, recall, and F measure can be used in evaluation of correctness.

4. *Generalization.* To generalize, a classifier must perform well with unseen images. This is a central issue in pattern recognition and learning theory. A typical approach to avoid overfitting in training is to *regularize* the structure of the classifier [43]. Another approach is to combine the outputs of an ensemble of several, perhaps weak, classifiers [7, 12].

In this image mining application, we adopt an active learning approach to minimize the need for labeled data in training a classifier, propose the use of *class-specific ensembles* in implementing active learning, and demonstrate that active learning can be extended to *Egeria* detection in images unseen to the active learner. We will show that: (1) by using active learning, we can significantly reduce expert involvement; (2) by having class-specific ensembles, we can conduct active learning effectively with a small number of labeled instances; and (3) by extending active learning to new images in an *iterative active learning* algorithm, we can rely on the learned results to further reduce expert involvement. Thus, we develop a novel data mining approach to assist human experts in efficient *Egeria* detection.

The proposed class-specific ensembles stem from our observation that different types of classifiers are better suited to detecting different objects such as *Egeria*, land, and water. Since it is impractical to train one classifier (or ensemble) for each object (as experts need to provide training instances for all objects), we propose to learn class-specific ensembles for two classes: *Egeria* and

non-*Egeria*. We will explain why the class-specific ensembles approach should outperform the conventional single ensemble approach. We also empirically show that this approach significantly reduces the number of uncertain image regions and is better than a single ensemble for the task of *Egeria* detection.

Being able to learn with limited labeled data during training does not solve the problem of generalization. We also need to show that active learning of class-specific ensembles can reduce expert involvement in classification of new images. This reduction is achieved by applying iterative active learning. With limited interaction with experts, our active learning scheme adapts the ensembles to new images. Because of the scarcity of training images, it is likely that images used in training only partially represent testing data (new images). Iterative active learning allows the ensembles to efficiently work on new image data with limited expert involvement.

Section 2 introduces the problem domain of *Egeria* mining from digital images. Section 3 presents some conventional approaches for image mining and related work. Our approach is described in Section 4. A novel concept of class-specific ensembles and the algorithms to find optimal class-specific ensembles are introduced in Section 5. Section 6 provides empirical evaluation details. Section 7 concludes this work with some immediate extensions.

2 *Egeria* Detection

Egeria, a submerged aquatic weed has grown uncontrolled in the Sacramento-San Joaquin Delta of Northern California for over 35 years and currently covers over 6000 acres of waterways. The presence of this exotic weed has disrupted navigation and recreational uses of waterways, clogged irrigation intake trenches, and caused reservoir-pumping problems [15]. The *Egeria* invasion has also displaced native flora and probably affected native fauna. In 1997, the California Department of Boating and Waterways started developing a control program to manage *Egeria*. At that time, researchers at the Romberg Tiburon Center for Environmental Studies (RTC) [13] were hired to

assess the effects of control protocols on fish and other fauna and to estimate the areal extent of *Egeria*. The RTC team has continued to monitor the areal extent of this weed every year since then, via visual/manual interpretation of color infrared (CIR) aerial photography. This imagery is flown at 1:24,000 scale and then scan-digitized to nominal 2-meter pixels.

Classifying *Egeria* in scan-digitized CIR imagery presents a challenging problem due to a number of variable and unfavorable conditions [15]. These include changes in imaging conditions (e.g., film exposure, vignetting, scanning anomalies), problems associated with water-related subjects (e.g., turbidity, sun glint, surface reflectance due to wind), and other environmental changes (e.g., exposure of *Egeria* at extremely low tide, shadows falling upon the water, algal cover over *Egeria*). Figure 1, a scan-digitized CIR aerial photograph, illustrates the spectral variations in *Egeria* that may occur even within a short distance. The figure also exhibits some problems caused by lack of spectral separation between *Egeria* and other extraneous classes. For example, it shows that dense well-submerged *Egeria* appears black and is confused with shadows on land when *Egeria* exposed during very low tide appears reddish and is confused with terrestrial vegetation. Digital analysis also indicates that subtle changes - for example, in *Egeria* canopy density, film vignetting, or water turbidity - produce overlapping spectral response patterns. Clearly, traditional computer-assisted multispectral classification methods are problematic under these conditions, and visual/manual image interpretation and analysis procedures are time-consuming and costly.

The *Egeria*-all variations figure goes here. (Figure 1)

3 Conventional Approaches

Conventional image classification methods focus on using single classification algorithms to detect the required patterns in images. Major categories of these classification algorithms are listed below.

(a) Decision Tree based algorithms, such as C4.5, Decision Stump, Id3, Alternating Decision Tree;

(b) Rule/Discretization based algorithms, like Decision Tree (PART), One Rule, PRISM, Hyper Pipes, Voting Feature Intervals;

(c) Neural Networks based algorithms, such as Voted Perceptrons, Kernel Density Estimators, Logistic;

(d) Support Vector Machine (SVM) based algorithms, like Sequential Minimal Optimization for SVMs;

(e) Probability Estimators, such as Naive Bayesian Classifier, Naive Bayesian Classifier-simple; and

(f) Instance Based algorithms, such as IB1, Decision Table.

The choice of an appropriate learning algorithm usually depends on the domain. Commonly used algorithms for detecting patterns in images are probability estimators (Bayesian-based), neural networks, support vector machines, decision trees, and their variants. For example, Kitamoto [23] developed a system using k-NN (k-Nearest Neighbors) for predicting the presence of typhoons from satellite images. In a different domain, Antonie et al. [3] used neural networks along with association rule mining to detect breast cancer from medical images. Hermes et al. [20] applied support vector machines [9, 22, 44] to a remote sensing application of classifying satellite images into regions of forests, water bodies, grasslands, etc. Salzberg et al. [39] used CART [8] and C4.5 [33] decision trees to detect cosmic ray hits from Hubble Space Telescope images.

In the above applications, the underlying function to be learned is uniform for all the different images in the task domain. As mentioned previously, *Egeria* detection presents its unique difficulty in *Egeria*'s spectral variations found in different images (as shown in Figure 1). Therefore, different types of classifiers might be better suited to detecting different objects such as *Egeria*, water, land, etc. It is impractical to train separate classifiers for each different object. In order to do so, the experts would need to provide training data to separate each object from all the other objects.

This could overburden the experts who are already overwhelmed with manual labeling. Giacinto and Roli documented that conventional methods do not perform well for such image mining applications [17]. They proposed the use of ensembles of neural networks, wherein classification results from a multitude of neural networks are "merged" by using statistical combination methods. The authors concluded that this is a valid alternative to designing new, *more complex* classifiers. Active learning that is adopted to relieve the experts from tedious manual labeling can be implemented using a similar approach. The subsequent two sections will explain the concepts of *active learning* and *iterative active learning*, as well as the need for and the details of *class-specific ensembles* in *Egeria* detection.

4 An Active Learning approach

Many real-world applications generate massive unlabeled data as in the case of *Egeria*. Manually obtaining labels for massive unlabeled data is not only time consuming but also unreliable. Experts can only process a small portion of the unlabeled images in a given period of time [15]. One goal of our work is to reduce human involvement in the labeling process by applying some learning methodology to automate this process of labeling for a large number of images.

Learning to detect *Egeria* requires an initial set of labeled training instances that differentiate *Egeria* and non-*Egeria*. In Section 2, we presented some difficult problems associated with *Egeria* detection, which indicates that a large number of training instances would be required in training an effective classifier that can work well on new images. As mentioned earlier, obtaining labeled training data is very expensive, while gathering unlabeled data is often straightforward [27]. Active learning is a supervised learning algorithm [10, 36], in which the learner has the freedom to select the data points to be added to the training set. As in the case of labeling, we can rely on active learning to select those critical unlabeled instances for labeling. If we can design effective active learners that can learn from a smaller set of labeled data, we may be able to significantly reduce

the number of critical unlabeled instances that need be labeled. This means we can reduce expert involvement in labeling. The rest of this section discusses issues of active learning in *Egeria* detection. Section 5 is about designing an effective active learner.

4.1 Active Learning

Active learning [40] can help reduce the number of supervised training instances needed to achieve a given level of performance [41, 42]. For example, an active learner can be trained with an initial set (S_0) of labeled data, and then is applied to another set (S) of unlabeled data. If the active learner is confident about its classification of an instance in S (the prediction of its class label), the prediction is *certain*; otherwise, it is *uncertain*. When manual labeling is time-consuming and labor-intensive, as is the case for *Egeria* detection, active learning may be able to help reduce the number of instances that need be labeled. An active learner may begin with a small set of labeled data, and predict class labels for unlabeled instances. The prediction can result in two sets of data: their predictions are either certain or uncertain. The instances with uncertain predictions are presented to human experts to assign class labels. The active learner is then *retrained* with the newly labeled data to improve its prediction. In short, active learning is basically a supervised learning algorithm, and requires an expert to resolve its uncertain classifications. If we can have an effective active learner, we can significantly reduce the number of instances with uncertain predictions. Thus, we will only ask human experts to resolve a small number of such instances.

Active learning has been used widely in classification applications on web documents. Some researchers [25, 40] have described applications of active learning that greatly enhance the generalization behavior of support vector machines [9, 22, 44]. Freund et al. [16] suggested combining selective sampling with the Query-by-Committee algorithm (QBC) [41] for active learning. They used a committee of perceptrons to sample from a training data set to reduce predictive error rates. Abe and Mamitsuka [1] proposed two variants of the QBC algorithm, query-by-bagging and query-by-boosting. Both of them performed better than QBC, C4.5, and boosting with C4.5. McCallum

and Nigam [27] modified the QBC method to use the unlabeled pool of documents and to select the examples to be labeled by explicitly estimating the density of the documents. They further combined active learning with the Expectation-Maximization algorithm to obtain class labels for unlabeled instances. Active learning was also shown to be useful in improving query answering [10]. The authors demonstrated how selective sampling can be approximately implemented using neural networks.

Another line of recent research [21, 18, 37, 38] concentrates on developing algorithms to process data automatically so that much less expert involvement is needed. A variant of an Active Learning algorithm has been suggested in [21] to learn from specific unlabeled instances via uncertainty sampling. Their goal is to reduce the number of queries that require attention from human experts. Hakkani-Tur et al. [18] suggested a similar approach in the domain of automatic speech recognition (ASR). The difference between their approaches is in their distinct sampling methods that select the most informative examples for active learning.

Muslea et al. [28] used selective sampling instead of uncertainty sampling to find the most informative unlabeled instances. The authors used two disjoint sets of feature-values *(views)* to learn separate classifiers and then proceeded to label the most informative unlabeled instances for which the two classifiers disagree, add them to the training data, and relearn the classifiers. They suggested that choosing the contention instances for which both classifiers are most confident provides maximal improvement. The authors continued their research [29] and experimentally showed that their algorithm Co-Testing + Co-EM (Co-EMT) outperforms the algorithms EM [31], Co-Training [5] and Co-EM [30] using artificial and real-world data sets.

Some researchers [37, 38] mentioned that most of the previous work on active learning focused on improving accuracy rather than reducing expert involvement. Instead, they concentrated on using class probability estimates to obtain the class probability rankings, which enable effective sampling from unlabeled instances. The authors proved that their sampling technique is better (in terms of size of the training data) than uncertainty sampling or bootstrapping.

The goal of applying active learning to *Egeria* detection is threefold: (1) to reduce the number of instances to be labeled by experts in digital imagery into *Egeria* and non-*Egeria* regions; (2) to learn an active learner from these labeled instances; and (3) to apply the active learner to the remaining unlabeled instances that are unseen in the training phase. Only when new instances cannot be handled confidently by the active learner will they be recommended to experts to resolve their labels. The number of recommended instances should be significantly smaller than the number of unlabeled instances. The reduction of these recommendations means reducing expert involvement. This interactive process can be repeated until almost all the unlabeled instances are confidently classified by the active learner. We next illustrate in detail how to apply active learning to process unlabeled images.

4.2 Iterative Active Learning

Clearly, an active learner built using a small training data set could have limitations. One key issue is whether the active learner can be successfully applied to instances of new images. It is possible that it might result in a large number of uncertain instances. Especially in the case of *Egeria* detection, as mentioned earlier, images were taken in varied conditions and had various noise elements, such as sun glint, turbidity, deep/shallow water, etc. Many images may share some commonalities, but may also have unique characteristics of *Egeria*. In other words, *Egeria* in different images may not have uniform spectral distributions. This indicates that there may not be straightforward correlations between the instances of the training image(s) and those of the unseen images. When the correlations are strong, the active learner may produce fewer instances with uncertain predictions; in other cases, the number of such instances may be large. This observation necessitates the adaptation of the active learner to new images.

Instead of asking experts to resolve all these uncertain instances, we propose an iterative active learning approach. Considering both the function of active learning and the efficacy of an expert at labeling, we propose to ask an expert to resolve a small number of instances, say 25, and use

this additional labeled data set to adapt the original active learner to a new image. Iterative active learning allows the learner to efficiently work on new images with limited expert involvement. It is expected that this significant reduction should mitigate the task of labeling for a domain expert. The iterative active learning continues until no improvement can be made.

The IALA algo goes here (Figure 2)

We present an iterative active learning algorithm (IALA) in Figure 2. It takes as input Tr, a new image Ts, the number of uncertain instances m to be labeled , and two classifiers (called dual ensembles, to be detailed in the next section). The value of m should be reasonably small so an expert can label m instances reliably. We set $m = 25$ in this work. The algorithm returns the adapted dual ensembles for Ts. The essence of the algorithm is to use a small amount of the expert's input to iteratively adapt the ensembles to a new image so that expert involvement can be further reduced while maintaining good performance. The oracle in the algorithm is the human expert. The iterative learning stops if the improvement of two performance measures (Fgain and UCgain defined in the algorithm) is insignificant ($< 5\%$ and $< 10\%$ respectively) or if UC_{new} is smaller than m. UC is the number of uncertain regions that the active learner cannot classify with high confidence. We now discuss performance measures for iterative active learning.

4.3 Performance measures

Precision, Recall, and Accuracy are the common criteria used for performance comparison. These measures are defined in terms of the instances that are relevant and the instances that are correctly classified (or retrieved). The true positives (TP) and true negatives (TN) are the correctly classified instances. A false positive (FP) is when the outcome is incorrectly predicted as YES when it is in fact NO. A false negative (FN) is when the outcome is incorrectly classified as NO when in fact it is YES. Precision, recall, and accuracy are defined in terms of TP, TN, FN, and FP [45, 4]

- $Precision = TP/(TP + FP)$: the fraction of the classified information which is relevant.

- $Recall = TP/(TP + FN)$: the fraction of the classified relevant information versus all relevant information.

- $Accuracy = (TP + TN)/(TP + FP + TN + FN)$: the overall success rate of the classifier.

Accuracy takes into account the true negatives (TN) in its numerator. If a particular image has a large number of class "negative" that are classified correctly, then the resultant accuracy rate may be misleadingly high, overshadowing the other components (TP, FN, FP). Particularly, in our application, we are mainly concerned with detecting *Egeria* (true positives). It has also been noted in [32] that accuracy may not provide a good measure for classification. Since both precision and recall have only TP in their numerator, they are suitable for performance measuring. In addition, we consider *reduction in uncertain regions* (UC) as a third measure.

High precision or high recall alone is not a good performance measure as each describes only one aspect of classification. Combined as in the F measure [26, 34], they provide a good measure.

- $F = 2 * P * R/(P + R)$: the harmonic mean of precision and recall.

If both precision and recall are 1 then F is 1, which means all and only positive instances are classified as positive. When either precision or recall is 0 then the F measure is 0. Hence, the F measure is a good measure for both generality and accuracy.

It is clear that the classifiers employed in the IALA algorithm should be effective in learning and able to work collaboratively. We introduce dual ensembles as the classifiers in active learning.

5 Dual Ensembles for Active Learning

Active learning is just a learning framework. In order to achieve highly accurate learning with a small set of labeled data (in order to reduce expert involvement), we need highly accurate base classifiers. The traditional approach of using a single classifier for detection in a complex domain becomes inadequate. Roli et al. [35] documented that finding a single "appropriate" classifier

for a particular classification task is very difficult, an appropriate classifier being the one with high predictive accuracy (i.e., generalizing well). Many recent approaches work with *ensembles* of classification algorithms and use a decision function to combine the classification outputs [12]. The ensemble methods often produce accurate and robust classifiers. We therefore adopt ensembles as base classifiers for active learning.

One quandary arises for active learning using ensembles. In order for active learning to work, an ensemble of highly accurate classifiers is needed so that the classifiers will disagree with each other but not too often. However, highly accurate classifiers usually do not disagree with each other so the prediction of an ensemble is always certain; while highly inaccurate classifiers may disagree too much, leading to an unnecessarily high number of unlabeled instances being recommended. The challenge now is how we employ highly accurate and diverse classifiers to form good ensembles for active learning. Class-specific ensembles are an example. We show below the novel features of class-specific ensembles, in particular dual ensembles, for active learning, and elaborate on how to learn dual ensembles.

5.1 Single vs. Dual Ensembles

Assuming a domain of two classes, the two examples in Figure 3 illustrate the difference between a single ensemble and dual ensembles. A single ensemble contains a fixed number of classifiers, which learn the separation between the classes, *True* and *False*. A single ensemble can produce three outputs based on consensus: *True, False, Uncertain*, as shown in the left of Figure 3. The middle part is uncertain as the ensemble cannot reach consensus; its posterior probability is close neither to 1 nor to 0. The *True* and *False* parts do not overlap because in such an ensemble learning, the focus is on one class and the other class is determined by default. In a more general setting where the class distributions for *True* and *False* are not exactly reversed, being certain about *True* does not necessarily mean being certain about *False*. Such a scenario is depicted in the right of Figure 3. In a domain with variable class distributions, it can be observed that ensembles may not

be highly certain about their predictions on some instances of the unseen images. These instances, depicted by the regions between the lines A and B are *don't know*s or uncertain. For a single ensemble, such uncertain predictions are observed when there is no obvious consensus among all the classifiers within the ensemble. A dual ensemble consists of two separate ensembles, one for each class. For a dual ensemble, one ensemble can predict the class of an unlabeled instance as either *True* or *Not True*, the other ensemble can predict the class as either *False* or *Not False*. When the two ensembles do not agree in their classifications, the prediction is deemed *Uncertain*.

The figure showing difference between single and dual ensembles (Figure 3)

Using dual ensembles allows us to take advantage of the difference between two highly accurate classifiers. Each ensemble tries to predict its class with high accuracy and is expected to provide a better classification and a better separation between the certain and uncertain classifications. We can not only use a small number of training instances to effectively learn ensembles (each ensemble being tuned specifically for detecting one class), but also ensure that high accuracy does not always produce false consensus. The subsequent problem to solve is how to identify relevant classifiers to form each of the dual ensembles.

5.2 In Search of Optimal Dual Ensembles

We may tend to use as many classifiers in an ensemble as possible for the following reasons.

- Each classification algorithm may have a different view of the training image. So different algorithms can capture varied aspects of the image because of their different biases and assumptions.

- No single classifier can completely cover a complex domain and generalize well. In other words, some algorithms may succeed in capturing some latent information about the domain, while others may capture different information.

However, problems can result from using too many classification algorithms. Some examples are as follows.

- Using more classification algorithms can result in longer overall training time, especially if some of the algorithms are time-consuming to train.
- Some classification algorithms may be prone to overfitting in the image domain. If these algorithms are included in the ensemble, there may be a high risk of allowing the ensemble to overfit the training image(s).

The above analysis suggests the necessity of searching for a relevant set of classifiers to form an ensemble. Exhaustive search for the best combination is impractical because the search space is exponential in the total number of classification algorithms for consideration. Thus we need a methodology to find the optimal combination of classifiers for the dual ensembles without resorting to exhaustive search. The optimality is defined in terms of performance measures as we discussed earlier. The search for optimal ensembles is to find a set of classifiers that forms an ensemble with best performance. An appropriate learning algorithm is needed that can optimize the performance measures in search of optimal ensembles.

Among many learning algorithms for classification, clustering, and association rules, we observe that association rule algorithms [2] can search the attribute space to find the best combination of attribute-values associated with a class. An association rule $A \Rightarrow B$ satisfies the minimum support and minimum confidence. The support for a rule is the joint probability $P(A, B)$ and the confidence is the conditional probability $P(B|A)$, where A and B are itemsets of attribute values (e.g., $a_1 = v_1, a_2 = v_2$, $b_1 = c_1, b_2 = c_2$). In our case, B is a class value ($b = c$), and A is a combination of attribute values. Thus the confidence of a rule gives us the measure of accuracy of the rule, while the support gives us the measure of generality of the rule. Association rules with high support and confidence, are those both general and accurate. There are efficient algorithms to learn association rules from data [19, 2].

Reviewing the definitions of precision and recall, we notice that precision and recall are parallel to confidence and support. Hence, we employ association rule algorithms to search for the optimal dual ensembles. This approach is different from feature selection [6, 11, 24], where the attribute space is searched to find the best combination of attributes rather than attribute-values.

Now we need a data set that links classifiers to the label of each image region in search of optimal ensembles. This new data set can be obtained by applying all the classification algorithms to the training data so that each classifier is a feature (i.e., column) and its value is the prediction of the classifier. For each image region (one instance in the new data set), there are predictions of all the classifiers and also the class label '*Egeria*' or 'non-*Egeria*' given by experts. We are concerned only with those association rules that have the class label '*Egeria*' or 'non-*Egeria*' on the righthand side (consequent). We will restrict our search to such rules and obtain rules with the maximum number of features (classifiers) on the lefthand side (precedent) without a significant loss in support or confidence. The best rule for each class label indicates the best combination of classifiers for the ensemble. Thus the ensembles obtained from this procedure are optimal in terms of both support and confidence, and correspondingly recall and precision. Next we discuss in detail the algorithm that implements the idea described above.

5.3 Algorithm Searching for Optimal Dual Ensembles

The search algorithm is presented in Figure 4 and further illustrated in Figure 5. It takes as input the entire set of classification algorithms E and training data Tr with class labels l_{Tr}, and produces as output the optimal ensembles for class label *yes* and class label *no*. The major steps are: (i) creating a new data set D (steps 1-3) by training all the classifiers E with the training data; (ii) learning association rules from D for dual classes (steps 4 and 5); and (iii) finding the best association rules for each class (steps 6-10). Rules with support-confidence product $> 90\%$ of the maximum support-confidence for Tr are considered for selection. Each rule set is ranked according to *length* - the number of classification algorithms in the precedent. This is because such rules have the

maximum number of tightly bound classifiers in predicting the class label. The longest rule from each set is selected to obtain the optimal ensemble for each class label.

The algorithm for finding optimal ensembles goes here (Figure 4)

The figure for the flow of the algorithm goes here (Figure 5)

The next task is to use the dual ensembles ($E_{l=yes}$ and $E_{l=no}$) to determine certain and uncertain instances. We need to decide the maximum number of classifiers in an ensemble that should agree on a prediction to reach a decision of "certain" or "uncertain" for each ensemble. An ensemble with all classifiers being required to agree on a prediction would lead to high precision, but low recall; an ensemble with few classifiers being required to agree would lead to high recall and low precision. Thus, we need to find the maximum number of classifiers with which the ensemble gives the best estimated precision and recall, and hence the best F measure. The training data is used again for this task. $E_{l=yes}$ is certain only if all $n_{l=yes}$ classifiers agree on *yes*. The F measure (F_0) is recorded. If $(n_{l=yes} - 1)$ classifiers agree, then F_1 is checked. This process is repeated to find F_k for $(n_{l=yes} - k)$ classifiers by incrementing k until 1 classifier remains. The agreement threshold for $E_{l=yes}$ is then the maximum number of classifiers with highest F measure. The same procedure is repeated for ensemble $E_{l=no}$.

The dual ensembles $E_{l=yes}$ and $E_{l=no}$ work together to decide if an instance's prediction is certain or not as follows. In predicting an instance, if both $E_{l=yes}$ and $E_{l=no}$ are certain and agree with their predictions, the instance is considered certain and labeled with the prediction; if they are certain and disagree, the instance is considered uncertain; if one is certain and the other is uncertain, follow the certain one; and if both are uncertain, the instance is uncertain. We now turn to the experimental evaluation of the algorithms proposed above.

6 Empirical Study

We performed experiments with a set of digital images of size 300×300 pixels in TIF format (RGB). The extracted features are of color, texture, and edge. There are 13 features in total. The details of feature extraction were described in our earlier work [14]. The template for feature extraction is 8×8 pixels. With 50% overlap between neighboring regions, there are a total of 74×74 or 5476 regions (instances) per image. We designed four experiments to evaluate the following:

1. How dual ensembles fare against single ensembles;
2. Whether we need to *learn* the dual ensembles;
3. How the dual ensembles fare against classification rules determined by experts; and
4. Whether the dual ensembles learned from the training image are applicable to unseen images.

With the principal goal of reducing the burden on experts, we used only one image for training and applied the learned results to another 16 testing images of different areas for *Egeria* detection. Among the classification algorithms available in the machine-learning package WEKA [45], we selected those that can be applied to the image domain to ensure the variety of classification algorithms. We applied the algorithm in Figure 4 with the complete set of classification algorithms as input. The optimal dual ensembles found by the algorithm are given below. The two ensembles are composed of different combinations of classifiers.

$E_{l=yes}$: C4.5, Alternating Decision Trees, Decision Trees (PART), PRISM, Hyper Pipes, Kernel Density, Logistic, Decision Tables $\Rightarrow$ 'Class = **yes**'.

$E_{l=no}$: Id3, Alternating Decision Trees, Decision Trees (PART), PRISM, Kernel Density, Instance Based1, Decision Tables $\Rightarrow$ 'Class = **no**'.

Let F and the number of uncertain instances for the k^{th} testing image from ensemble i be F_k^i and UC_k^i, and let the corresponding values from ensemble j be F_k^j and UC_k^j. We calculate the F measure gain and the uncertain instance increase averaged over n testing images as follows:

$$AverageUCIncr = \frac{\sum_{k=1}^{n} UC_k^j - \sum_{k=1}^{n} UC_k^i}{\sum_{k=1}^{n} UC_k^i} \tag{1}$$

$$AverageFGain = \frac{\sum_{k=1}^{n} \frac{F_k^j - F_k^i}{F_k^i}}{n} \tag{2}$$

The first set of results goes here (Table 1)

Table 1 summarizes experimental results in four columns (A, B, C, D). Image #1 is the training image. The last two rows show the average Fgain and average UCincrease with respect to the results in Column A.

Experiment 1. We compared single optimal ensembles (either $E_{l=yes}$ or $E_{l=no}$) with dual optimal ensembles ($E_{l=yes}$ and $E_{l=no}$). The results are shown in Column B. The average UCincrease is almost 53% and the average Fgain is -0.55%. It is evident that in general, dual ensembles are not only more accurate, but also separate certain and uncertain instances better than single ensembles, except for 2 cases (images #9 and #15).

Experiment 2. We compared 10 pairs of randomly selected dual ensembles with the optimal dual ensembles to check if the optimal dual ensembles could be found by chance. For each pair of random dual ensembles, each classifier was randomly chosen from one of the categories mentioned earlier and learned from the training image. Although the average Fgain is only increased by 1.26%, the UC increases significantly by 846.6% as shown in Column C of Table 1. We conclude that it is necessary to search for optimal dual ensembles, as random dual ensembles work poorly in reducing UC.

Experiment 3. We compared the classification rules given by the domain experts with the optimal dual ensembles. The experts' rules outperform the optimal dual ensembles in terms of Fgain by 8.6%, but UC increases by 408.4% (in Column D of Table 1). The high Fgain and high UC for the expert classification rules is due to the fact that an expert can only directly work on the

former (designing highly general and accurate rules), but not on the latter (finding low UC rules). Our active learning system is particularly designed to overcome this shortcoming.

Experiment 4. We explored if the optimal dual ensembles can be further improved via iterative active learning. This function would be very useful in dealing with new images for *Egeria* detection. We can observe in Table 1 that some of the unseen testing images (e.g., # 8) have a high number of uncertain instances. It is impractical to overwhelm the expert to resolve such a high number of uncertain instances. The algorithm in Figure 2 iteratively selects a small number of certain and uncertain instances (from such images) and adds them into the original training data after experts resolve the uncertain instances.

The second set of results goes here (Table 2)

The results of iterative active learning are shown in Table 2. After a few more iterations of learning, three out of the four images with $UC > 25$ achieve Fgain (average 15.41%) and negative UCincrease (average -64.77%). These results suggest that it is practical to adapt the learned dual ensembles to new unseen images to achieve high performance in terms of Fgain and reduced uncertain instances.

7 Summary and Conclusion

We have introduced active learning to reduce expert involvement in data labeling, presented a novel approach to active learning with class-specific ensembles of classifiers, and proposed iterative active learning to adapt the active learner to new images. In particular, dual ensembles were implemented and tested, and one ensemble was trained for *each class*. The search of optimal ensembles was transformed into discovering association rules between classifiers and a class label. The learned ensembles were then adapted to new images via iterative active learning. Extensive experiments were conducted in the real-world domain of detecting '*Egeria*' in scan-digitized aerial photography. The experiments compared the optimal dual ensembles with optimal single ensem-

bles, with randomly selected dual ensembles, and with classification rules determined by domain experts. The class-specific ensembles outperformed other methods in terms of uncertain region reduction by 52.7%, 846.6%, and 408.4% respectively. Thus, active learning with dual ensembles can decrease expert involvement in instance labeling. The experimental results show that both components of the solution (class specific ensembles and iterative active learning) can significantly reduce expert involvement without compromising performance. The base classifiers used in ensembles are currently of different types. Future work will be extended to using ensembles with one type of classifier (e.g., decision trees as in Random Forests). This may alleviate the the problem of classifier selection, and pave the way to efficiently build class specific ensembles for more than two classes.

8 Acknowledgments

The authors wish to thank Deepak Kolippakkam and Jigar Mody for their contributions to the *Egeria* Mining project.

References

[1] N. Abe and H. Mamitsuka. Query learning using boosting and bagging. In *Proceedings of the 15th International Conference on Machine Learning*, pages 1–10, 1998.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of International Conference on Very Large Data Bases (VLDB)*, pages 487–499, Santiago, Chile, September 1994.

[3] M. Antonie, O. Zaiane, and A. Coman. Application of data mining techniques for medical image classification. In *Proceedings of Second International Workshop for Multimedia Data*

Mining (MDM/KDD'2001) in conjunction with ACM SIGKDD conference, pages 94–101, 2001.

[4] R. Baeza-Yates and B. Ribeiro-Neto. *Morden Information Retrieval*. Addison Wesley and ACM Press, 1999.

[5] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the Workshop on Computational Learning Theory*. Morgan Kaufmann, 1998.

[6] A.L. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97:245–271, 1997.

[7] L. Breiman. Random forests. Technical report, Statistics Department, University of California Berkeley, 2001.

[8] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*. Wadsworth & Brooks/Cole Advanced Books & Software, 1984.

[9] C.J.C. Burges. A tutorial on support vector machines. *Journal of Data Mining and Knowledge Discovery*, 2, 1998.

[10] D. Cohn, L. Atlas, and R. Ladner. Improving generalization with active learning. *Machine Learning*, 15:201 – 221, 1994.

[11] M. Dash and H. Liu. Feature selection methods for classifications. *Intelligent Data Analysis: An International Journal*, 1(3), 1997.

[12] T.G. Dietterich. Ensemble methods in machine learning. In *First International Workshop on Multiple Classifier Systems*, pages 1–15. Springer-Verlag, 2000.

[13] Romberg Tiburon Center for Environmental Studies. Egeria densa project, 2002. http://romberg.sfsu.edu/ egeria.

[14] P. Foschi, N. Kolippakkam, H. Liu, and A. Mandvikar. Feature extraction for image mining. In *International Workshop on Multimedia Information Systems (MIS 2002)*, pages 103 – 109, October 2002.

[15] P. Foschi and H. Liu. Active learning for classifying a spectrally variable subject. In *2nd International Workshop on Pattern Recognition for Remote Sensing (PRRS 2002), Niagara Falls, Canada*, pages 115–124, 2002.

[16] Y. Freund, H. Seung, E. Shamir, and N. Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28:133–168, 1997.

[17] G. Giacinto and F. Roli. Ensembles of neural networks for soft classification of remote sensing images. In *Proceedings of the European Symposium on Intelligent Techniques, Italy, 1997*, pages 166–170, 1997.

[18] D. Hakkani-Tur, G. Riccardi, and A. Gorin. Active learning for automatic speech recognition. In *International Conference on Acoustics Speech and Signal Processing 2002*, 2002.

[19] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In *Proceedings of Special Interest Group on Management of Data, SIGMOD-2000*, pages 1–12, 2000.

[20] L. Hermes, D. Frieauff, J. Puzicha, and J. Buhmann. Support vector machines for land usage classification in landsat TM imagery. In *Proceedings of International Geoscience and Remote Sensing Symposium (IGARSS'99)*, pages 348–350, 1999.

[21] V. Iyengar, C. Apte, and T. Zhang. Active learning using adaptive resampling. In *Proceedings of 6th ACM International Conference on Knowledge Discovery and Data Mining*, pages 92–98, 2000.

[22] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In C. Nedellec and C. Rouveirol, editors, *Proceedings of 10th European Conference on Machine Learning*, pages 137 – 142, Chemnitz, Germany, 1998. Springer.

[23] A. Kitamoto. Data mining for typhoon image collection. In *Proceedings of Second International Workshop for Multimedia Data Mining (MDM/KDD'2001) in conjunction with ACM SIGKDD conference*, pages 68–77, 2001.

[24] R. Kohavi and G.H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.

[25] R. Liere and P. Tadepalli. Active learning with committees for text categorization. In *Proceedings of the 14th National Conference on Artificial Intelligence and 9th Innovative Applications of Artificial Intelligence Conference (AAAI-97/IAAI-97)*, pages 591–597, Menlo Park, 1997. AAAI Press.

[26] J. Makhoul, F. Kubala, R. Schwartz, and R. Weischedel. Performance measures for information extraction. In *Proceedings of DARPA Broadcast News Workshop*, February 1999.

[27] A. McCallum and K. Nigam. Employing EM in pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 350–358, 1998.

[28] I. Muslea, S. Minton, and C. Knoblock. Selective sampling with redundant views. In *Proceedings of the National Conference on Artificial Intelligence*, pages 621–626, 2000.

[29] I. Muslea, S. Minton, and C. Knoblock. Active + semi-supervised learning = robust multi-view learning. In *Proceedings of the 19th International Conference on Machine Learning*, pages 435–442, 2002.

[30] K. Nigam and R. Ghani. Analyzing the effectiveness and applicability of co-training. In *Proceedings of Conference on Information and Knowledge Management*, pages 86–93, 2000.

[31] K. Nigam, A. K. Mccallum, S. Thrun, and T. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39:103–134, 2000.

[32] F. Provost, T. Fawcett, and R. Kohavi. The case against accuracy estimation for comparing induction algorithms. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 445 – 453. San Francisco: Morgan Kaufmann, 1998.

[33] J.R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[34] C. Van Rijsbergen. *Information Retrieval, 2nd edition*. Butterworths, 1979.

[35] F Roli, G. Giacinto, and G. Vernazza. Methods for designing multiple classifier systems. In *Multiple Classifier Systems*, pages 78–87. Berlin: Springer-Verlag, 2001.

[36] N. Roy and A. McCallum. Toward optimal active learning through sampling estimation of error reduction. In *Proceedings of the Eighteenth International Conference On Machine Learning*, 2001.

[37] M. Saar-Tsechensky and F. Provost. Active learning for class probability estimation. In *Proceedings of International Joint Conference on AI*, pages 911–920, 2001.

[38] M. Saar-Tsechensky and F. Provost. Active sampling for class probability estimation. In *Proceedings of Machine Learning*, 2002.

[39] S. Salzberg, R. Chandar, H. Ford, S. Murthy, and R. White. Decision trees for automated identification of cosmic rays in hubble space telescope images. In *Proceedings of the Astronomical Society of the Pacific,*, volume 107, pages 279–288, 1995.

[40] G. Schohn and D. Cohn. Less is more: Active learning with support vector machines. In *Proceedings of the Seventeenth International Conference On Machine Learning*, pages 839–846, 2000.

[41] H.S. Seung, M. Opper, and H. Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pages 287–294, Pittsburgh, PA, 1992. ACM Press, New York.

[42] C.A. Thompson, M.E. Califf, and R.J. Mooney. Active learning for natural language parsing and information extraction. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pages 406–414. Morgan Kaufmann, 1999.

[43] A.N. Tikhonov and V.Y. Arsenin. *Solutions of Ill-posed Problems*. W.H. Winston ed., 1977.

[44] V.N. Vapnik. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., 1995.

[45] I.H. Witten and E. Frank. *Data Mining - Practical Machine Learning Tools and Techniques with JAVA Implementations*. Morgan Kaufmann Publishers, 2000.

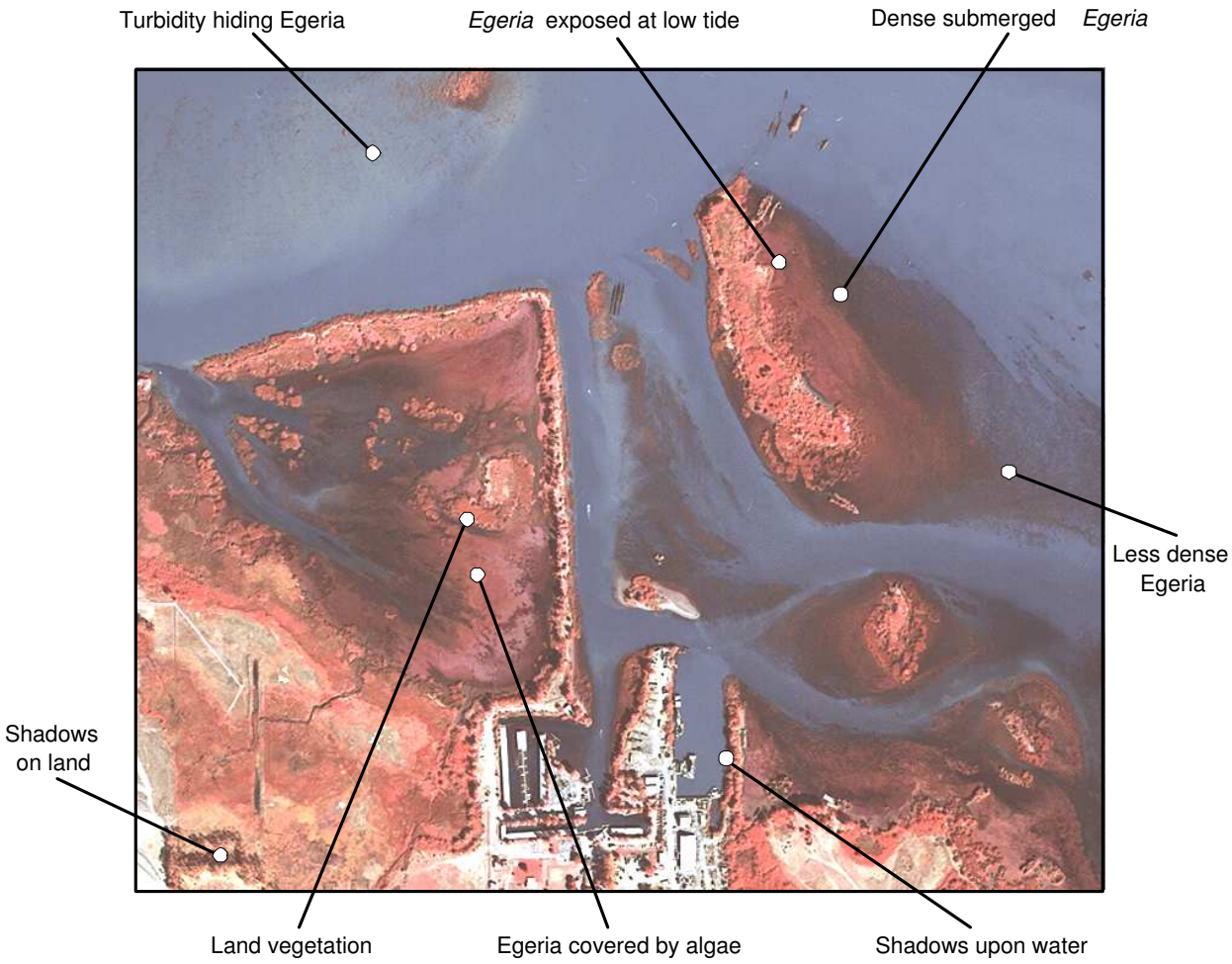

Figure 1: Scan-digitized CIR aerial photography showing spectral variations in *Egeria* and lack of spectral separation between *Egeria* and other extraneous classes.

input: Tr, Ts, $m = 25$, $E_{l=yes}$, $E_{l=no}$, $F' = 5\%$, $UC' = 10\%$;
output: $E'_{l=yes}$, $E'_{l=no}$: adapted ensemble pair ;

$P_{old} \leftarrow 0$, $R_{old} \leftarrow 0$, $F_{old} \leftarrow 0$, $UC_{old} \leftarrow 0$;
Classify Ts with $E_{l=yes}$ and $E_{l=no}$;
Obtain Ts_{cer} and Ts_{uncer}, UC_{new} =#Ts_{uncer};
if $UC_{new} \geq m$
  Calculate P_{new}, R_{new}, F_{new};
  do
    $Ts_{uncer} \leftarrow$ RandomSamples(Ts_{uncer},m);
    $Ts_{cer} \leftarrow$ RandomSamples(Ts_{cer},m);
    $Tr \leftarrow Tr + Ts_{cer}$;
    foreach $x_i \in Ts_{uncer}$ **do**
      $l \leftarrow$ class label(x_i) from an oracle;
      $Tr \leftarrow Tr + \{x_i, l\}$;
    Retrain $E_{l=yes}$and $E_{l=no}$ with Tr; apply to Ts;
    Obtain Ts_{cer} and Ts_{uncer};
    $UC_{old} = UC_{new}$; $F_{old} = F_{new}$;
    Recalculate P_{new}, R_{new} and F_{new};
    Fgain = $\frac{F_{new} - F_{old}}{F_{old}}$;
    UCgain = $\frac{UC_{old} - UC_{new}}{UC_{old}}$;
  while (Fgain$> F' \wedge$ UCgain$> UC'$) $\vee UC_{new} > m$;
  Return $E'_{l=yes}$ and $E'_{l=no}$;
end;

Figure 2: Iterative Active Learning Algorithm

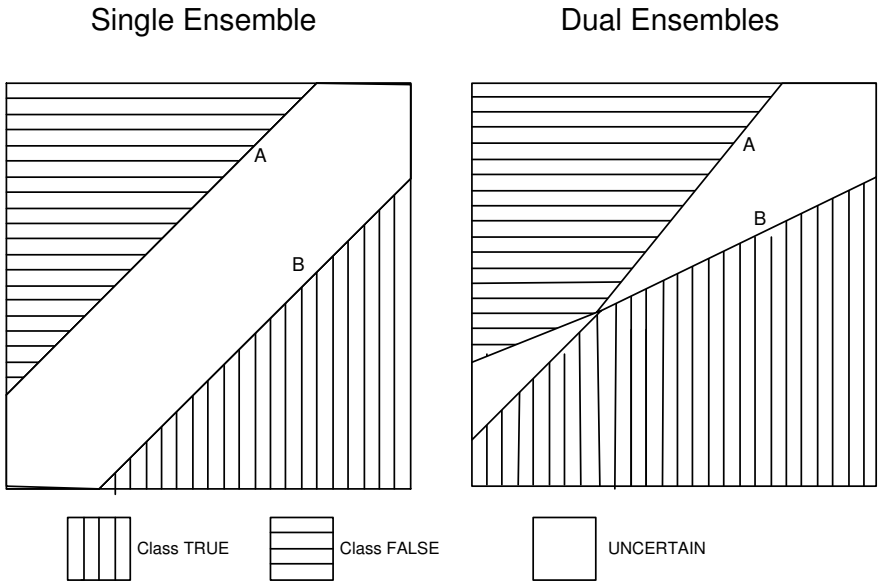

Figure 3: An illustrative example for two types of ensembles

input: Tr, E : Set of n classification algorithms
output: $E_{l=yes}$, $E_{l=no}$

01 Train E with Tr to obtain n classifiers, cl_1 to cl_n;
02 Obtain class labels, l^1_{Tr} to l^n_{Tr} for Tr using cl_1 to cl_n;
03 Form a data set, $D \leftarrow \{l^1_{Tr}, l^2_{Tr}, ..., l^n_{Tr}, l_{Tr}\}$;
04 Learn association rules, $Assoc$ from D;
05 $Assoc_1 \leftarrow$ Filter ($Assoc$ / consequent is $l_{Tr} = yes$);
06 $m_{l=yes} \leftarrow$ Max ($Assoc_1, supp * conf$);
07 $Assoc_1 \leftarrow$ Filter ($Assoc_1 / supp * conf \geq 0.9 * m_{l=yes}$);
08 $Assoc_1 \leftarrow$ Sort ($Assoc_1, length(precedent)$);
09 $E_{l=yes} \leftarrow$ Precedent(First($Assoc_1$));
10 Repeat steps 5 to 9 for $l_{Tr} = no$ to obtain $E_{l=no}$;

Figure 4: Algorithm for Optimal Ensemble Selection

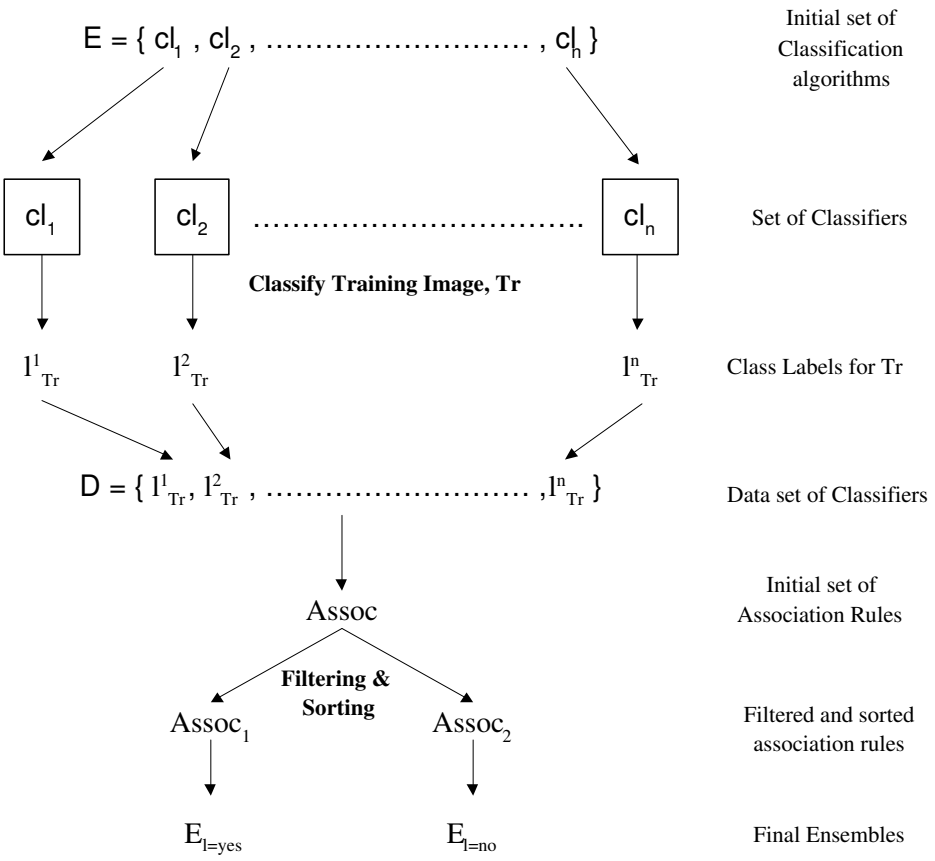

Figure 5: Illustration of the Algorithm in Figure 4

#	Optimal Dual Ensembles (A)		Optimal Single Ensemble (B)		Random Dual Ensembles (C)		Domain Expert's Rules (D)	
	F*	UC*	F*	UC*	F*	UC*	F*	UC*
1	0.7557	0	0.7557	0.5	0.7561	23.6	0.8509	35
2	0.7851	0	0.7851	0	0.77011	523.2	0.8040	582
3	0.6609	7	0.6611	23	0.67101	305.3	0.7401	50
4	0.7101	8	0.7103	22.5	0.7053	161.5	0.7785	18
5	0.5920	9	0.5921	14	0.5989	290.6	0.7467	86
6	0.7711	20	0.7543	72	0.7428	230.2	0.7755	95
7	0.8169	5	0.8162	18.5	0.8091	224.3	0.7980	121
8	0.4540	159	0.4415	209.5	0.5139	349.9	0.7327	253
9	0.5069	29	0.5120	13.5	0.5121	252.3	0.4586	33
10	0.4950	44	0.4923	53.5	0.5425	152	0.4627	134
11	0.4403	66	0.4197	107	0.4122	241.1	0.5644	129
12	0.6806	8	0.6811	9.5	0.6677	85.5	0.6780	63
13	0.6002	16	0.6008	22	0.5962	121.9	0.5835	58
14	0.6736	24	0.6722	41.5	0.6954	396	0.7091	99
15	0.5850	14	0.5867	3	0.6044	268.4	0.6291	245
16	0.8024	12	0.8011	22	0.8039	85.4	0.8132	41
17	0.6957	7	0.6936	21.5	0.7254	340.3	0.7011	134
	Avg UC* Insts	25.18	Avg UC*	38.44	Avg UC*	238.32	Avg UC*	128
Comparative Results			Avg UC* Incr	52.7%	Avg UC* Incr	846.6%	Avg UC* Incr	408.4%
			Avg F Gain	-0.55%	Avg F Gain	1.26%	Avg F Gain	8.60%

Table 1: Experimental Results

#	Before Iterative AL		After Iterative AL		Fgain	UCincr	# runs	# queries
	F Measure	UC	F Measure	UC				
8	0.4540	159	0.5762	47	26.90%	-70.44%	3	75
9	0.5069	29	0.5069	29	0.0%	0.0%	1	25
10	0.4950	44	0.5385	11	8.77%	-75.0%	2	50
11	0.4403	66	0.5547	18	25.96%	-72.73%	3	75
	Average UC Insts	74.5	Average UC Insts	26.25	15.41%	-64.77%	2.25	56.25

Table 2: Experiment 4 results