Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 08-023

Unsupervised Learning Based Distributed Detection of Global Anomalies

Junlin Zhou, Aleksandar Lazarevic, Kuo-wei Hsu, and Jaideep Srivastava

July 18, 2008

Unsupervised Learning Based Distributed Detection of Global Anomalies

Junlin Zhou, Aleksandar Lazarevic, Kuo-Wei Hsu and Jaideep Srivastava

Abstract

Anomaly detection has recently become an important problem in many industrial and financial applications. Very often, the databases from which anomalies have to be found are located at multiple local sites and cannot be merged due to privacy reasons or communication overhead. In this paper, a novel general framework for distributed anomaly detection is proposed. The proposed method consists of three steps: (i) building local models for distributed data sources with unsupervised anomaly detection methods, (ii) transforming local models into uniform models , and (iii) reusing learned models for new data and combining their results by considering both quality and diversity of them to detect anomalies in a global view. In experiments performed on several synthetic and real life large data sets, the proposed distributed anomaly detection method achieved prediction performance comparable or even slightly better than the global anomaly detection algorithm applied on the data set obtained when all distributed data sets were merged.

1. Introduction

The explosion of very large databases and the World Wide Web has created extraordinary opportunities for monitoring, analyzing and predicting global economical, geographical, demographic, medical, political and other processes in the world. However, despite the enormous amount of data being available, particular events of interests are still quite rare. These rare events, often called anomalies, are defined as events that occur very infrequently (their frequency ranges from 5% to less than 0.01% depending on the application). Detection of anomalies (rare events) has recently gained a lot of attention in many domains, ranging from detecting fraudulent transactions and intrusion detection to engineering health management (prognostics and diagnostics) and direct marketing. For example, in the network intrusion detection domain, the number of cyber attacks on the network is typically a very small fraction of the total network traffic. In prognostics and diagnostics applications, data records that correspond to failures that may occur in particular engines or its components correspond only to small portion of all the data records recorded in monitoring.

Data mining techniques that have been developed for detecting anomalies have been based on two major approaches, namely supervised and unsupervised techniques. Supervised learning methods typically build a prediction model for different types of rare events based on labeled data (both normal data and rare events), and use it to classify data record [11], [24], [25]. On the other hand, unsupervised learning methods typically do not require labeled data and detect anomalies as data points that are very different from the normal (majority) data based on some measure [23]. These methods are typically called anomaly detection techniques, and their success depends on the choice of similarity measures, feature selection and weighting, etc. Anomaly detection algorithms have the advantage that they can detect new types of rare events as deviations from normal behavior, but on the other hand suffer from a possible high rate of false positives, primarily because previously unseen (yet normal) data are also recognized as anomalies, and hence flagged as interesting. There are generally two types of anomaly detection algorithms, namely semi-supervised and completely unsupervised techniques. Semi-supervised anomaly detection techniques require knowledge of the normal behavior to build a model for its characterization, while unsupervised techniques do not require any knowledge about the labels and usually assume that the anomalies are data records that are significantly different than others.

The current research in anomaly detection using advanced data mining techniques so far has been focused on detecting various types of anomalies from individual data sources. However, sometimes anomalies that occur at multiple locations simultaneously may be undetected by anomaly detection algorithms built only from a single location. For example, detecting anomalous events and trends in near-real time from several multiple data sources (e.g., sets of independent components) during a flight can be helpful in making crucial decisions such as whether to abort the launch of a spacecraft prior to reaching the intended altitude. Such anomalous events and trends can best be detected, especially in their earlier stages, by correlating information collected across dispersed locations. However, such exchange of relevant and

useful information typically requires significant communication among the sites. Thus, there is need for an efficient exchange of limited amount of relevant information to allow anomaly detection in near-real time and at early stages. Furthermore, very often due to privacy reasons, data from multiple locations cannot be aggregated or exchanged, which represents additional challenge for a distributed anomaly detection algorithm. However, the existing techniques for anomaly detection from distributed data sources typically do not consider these restrictions.

In this paper, we propose a novel general framework for anomaly detection from distributed data sources that cannot be directly merged. In the proposed method, anomaly detection methods are first applied on data collected from individual data sites and the models for each site are created. These models are then applied to new data and their results are combined for the final list of detected anomalies. We have investigated three clustering based of unsupervised anomaly detection methods. We have also examined a new weighted voting for combining anomaly detection models. We have tested our anomaly detection algorithms on simulated data, several publicly available data sets as well as on flight record data obtained from NASA ASIAS systems. Our experimental results have shown that our proposed combining methods can achieve comparable detection performance to the single global model in which data from all dispersed locations are merged together.

The remainder of this paper is organized as follows. In the next section, we review the related work and give notation that will be used throughout the paper. Section 3 provides description of investigated anomaly detection algorithms as well as methods for their combining. In Section 4, we present our experiments and report the results of applying the proposed framework on a synthetic, several publicly available as well as on NASA ASIAS data sets. Finally, Section 5 summarizes the main contribution of the work and gives the conclusion.

2. Background

2.1. Related work

To solve the problem of detecting anomalies from very large and distributed databases, some researchers have been proposing modifications of standard ensemble classification schemes. These ensemble techniques typically manipulate the training data patterns individual classifiers use (e.g. bagging [2], boosting [12], arcing [3] and random forests [4]) or the class labels (e.g. ECOC [20]). In general, an ensemble of classifiers must be both diverse and accurate in order to improve prediction of the whole. In addition to classifiers' accuracy, diversity is also required to ensure that all the classifiers do not make the same errors. The application of ensemble learning is widely used in many domains such as image analysis in [8], handwriting recognition in [13] and medical diagnosis in [23].

Using ensemble methods for distributed learning has also gained a lot of attention among researchers recently. The simplest method for distributed learning is to combine different multiple predictors in a "black-box" manner. Different meta-learning techniques explored at the Jam project [26] were proposed in order to coalesce the predictions of classifiers trained from different partitions of the training set. Similarly, a knowledge probing approach for distributed learning from homogeneous data sites in the first phase learns a set of base classifiers in parallel, and in the second, the meta-learning is applied to combine the base classifiers. The boosting technique has also been adapted for the problem of distributed learning [27], [28]. However, there have been only a limited number of proposed techniques for distributed unsupervised learning. Several researchers have proposed approaches for combining anomaly detection approaches in distributed environment (e.g., Clustering-based DIDS [22], Distributed Network Monitoring and Anomaly Detection based on Principal Component Analysis [6] and Distributed Intrusion Detection based on genetic programming [11]). However, all these approaches require exchanging more or less initial data information or data descriptions (e.g., minimal bounding rectangles, convex hulls) from multiple locations.

In this paper we propose a novel approach for distributed anomaly detection using combination of local models from local sites. The most significant feature of the proposed approach is that it does not need global information to combine local models. Moreover, we employ unsupervised learning techniques, which make our approach unique comparing to other methods.

2.2. Notation

Let $X = \{x_1, x_2, ..., x_m\} \subset R^d$ denote a set of m data records with unknown labels (normal or anomaly) in a d-dimensional feature space. The i-th data record x_i is a d-dimensional feature vector $[x_{i,1}, x_{i,2}, x_{i,3}, \ldots, x_{i,d}]$. Clusters learned by unsupervised anomaly detection model in distributed datasets are represented as $\boldsymbol{Cj} = \{ x_{j1}, x_{j2}, \ldots, x_{Nj} \}, j = 1, \ldots, N_c.$, where N_j is the number of data records in cluster j, and N_c is the total number of clusters. An unsupervised anomaly detection model predicts a data record x_i as normal or anomalous and assigns it a label $\lambda_i \in \{normal, anomaly\}$ and an anomaly score $s_i \in [0,1]$. A higher anomaly score corresponds to a higher possibility that the data record is an anomaly. Therefore, an anomaly detection model

can also be described as a function that maps the set of data records $X = \{x_1, x_2, \ldots, x_m\}$ into a label vector $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \ldots, \lambda_m\}$.

3. Methodology

3.1. General framework for distributed anomaly detection

In this paper, we present a novel approach for distributed anomaly detection that is based on building a global model by exchanging local models from distributed sites (Figure 1).

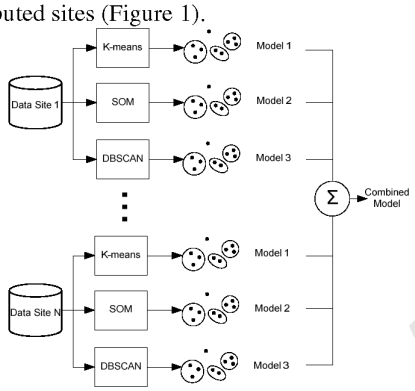

Figure 1. Framework for combining heterogeneous models derived from distributed and independent data sites to generate a global model.

In the scenario described in Figure 1, data are stored at several independent sites which do not exchange information among themselves. Each site obtains anomaly detection models from local data using unsupervised learning (clustering) techniques, such as K-means [14], SOM [16], and DBSCAN [5]. A data point that does not belong to any of the clusters will be detected as an abnormal data point or an anomaly. Multiple anomaly detection models derived from a local site could be combined into a general model. Nevertheless, such a general model is specific to local data and its usefulness and generality are limited. In order to integrate local models from each site into a global model, in this paper, we propose a new approach for model combination. The proposed approach has the following characteristics: (1) it does not require the global data and (2) it reuses the knowledge embedded in local models built from local sites.

Figure 2 presents proposed three-step model combination approach. Initially, in the first step, we build local anomaly detection models from local sites using unsupervised learning algorithms. Based on the assumption that anomalies in the data are quite rare, we can build the local anomaly detection model using clustering algorithms. After applying particular clustering algorithm, we label the largest cluster (one with the largest number of data points) as a normal. Lets assume its centroid is μ_0. We sort the remaining clusters in the ascending order of the distance from their cluster centroid to μ_0. Within a cluster, we sort the data points in the same way. We select the first $N_1 = \mu \cdot N$, data points (N is the total number of data points) and label them as normal; where N is the total number of data points and μ is the percentage of normal data points. The parameter μ is given or estimated fraction of all data points as normal ones.

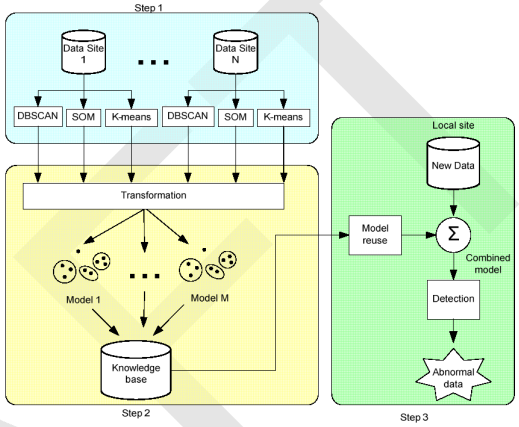

Figure 2. The proposed 3-step approach for reusing learned local models to detect abnormal data for new data.

Next, in step 2, we transform all local anomaly detection models into a uniform format and use four measures to compare qualities of the transformed models. We select the best qualified model from each site and save them into a knowledge base for the next step. Here, we use several algorithms to build different models from the same local data site, and we transform them into a uniform format. This step would allow us to compare the algorithms' anomaly detection qualities with each other, and to choose the best one.

Finally, in the third step, we reuse the knowledge that is learned from all local sites by effectively combining them. When a new data point arrives, we compute the anomaly score and label the new data point by using uniform anomaly detection models in the knowledge base. The anomaly score is computed by comparing the distance from the new data point to all cluster centroids with the corresponding distance thresholds. The value, for all centroids, is the minimum rate of {distance from new data to centroid / centroid distance threshold}, while the label {normal, abnormal} is decided by the anomaly score. If the anomaly score is higher than 1, then the new data point is an anomaly. From a global viewpoint, detecting new data by combining all these results into final anomaly scores is the main contribution of our work. The quality and diversity of the model are considered in the combination process. Since we have computed the quality of each model in the second step, we can reuse it in the final step. For diversity, we apply three

measures. Furthermore, we also consider the model diversity spread in assigning the diversity weights to each model. We determine the final weights for each model using both the equality and the diversity metrics. Since they are both important factors, we give them the same emphasis. Note that all weights are normalized into (0,1).

3.2. Uniform model format

Combining unsupervised anomaly detection models is a variant of the distributed anomaly detection problem. We combine the models learned from each local site in order to build the global model. Since a uniform model is necessary, we develop it based on the *k*-means algorithm. We evaluate the anomaly detection models by comparing the measure of the clustering quality. To transform a model into the uniform *k*-means model, we need to do two things. First, we need to remove the anomalies that the model has detected in the dataset. Second, we apply *k*-means algorithm on each cluster that some other anomaly detection algorithm has learned. When the initial anomaly detection model, such as DBSCAN, does not provide information about centriods and diameters of clusters, we consider it the case that all the normal data are in the same cluster. After that, all the models are transformed into the uniform *k*-means model. The uniform model examines distances between new data with centroids of the model to distinguish anomalies.

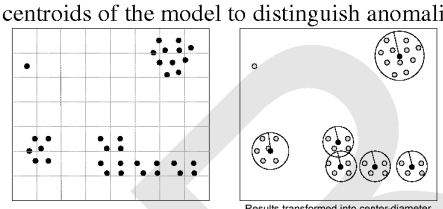

Figure 3. Example: results obtained by using DBSCAN (left) and from transformation (right)

DBSCAN is a density-based unsupervised learning approach and it is different from cluster-based approaches, in which a cluster is determined by its center and diameter and each data point is assigned to (at least) one cluster. In order to regularize the storage format in the knowledge base, we apply a transformation to results learned by DBSCAN. Figure 3 gives an example for this transformation. The left-hand side of Figure 3 illustrates results obtained by using DBSCAN. Grids are used to characterize the density-based approach. On the other hand, the right-hand side of Figure 3 demonstrates the transformed results that are compliant with those in cluster-based approaches. The concept behind the transformation is to keep sufficient information (centroids and diameters) used to describe clusters, e.g., position and size of a cluster.

3.3. Anomaly detection algorithms

We employ three unsupervised learning (clustering) algorithms to perform anomaly detection. We also study how to effectively combine results given by these algorithms.

k-means. [14] This method aims to find *k* clusters such that the average distance between a data point and a cluster center is minimized. We determine a data point as an anomaly if it does not belong to any cluster in the given threshold.

Self-Organizing Maps (SOMs). [16] In this method, all data objects in the feature space retain as much as possible their distance and neighborhood relations in the mapped space. The mapping is performed by a specific type of neural network, equipped with a special learning rule. In this process, each neuron in this neural network stores *d*-dimensional vector that serves as a cluster center *C*. To evaluate if a new test data point is anomaly, we use a procedure similar to one in *K*-means clustering approach.

DBSCAN Approach [5]. is a density based approach for clustering data. It can find arbitrary shaped clusters along with noisy outliers. DBSCAN clustering is based on two input parameters the size of epsilon neighborhood *e* and the minimum points in a cluster. Points are declared to be outliers if there are few other points in the *e*-Eucledean neighborhood. Thus, DBSCAN method can be viewed as a modified nearest-neighbor algorithm [18].

3.4. Combining methods

We propose a novel approach for combining unsupervised anomaly detection models by considering both the quality and diversity of models.

There are approaches to combine the different clustering results (for example [21]). In our scenario, we combine the multiple diverse local models learned from distributed sites by exchanging the uniform models.

For each new data record *y*, this method takes predicted anomaly scores $\boldsymbol{S}^{(j)}$ from all the anomaly detection models $\boldsymbol{M}_j$ that are built at local sites and computes a global anomaly score S by a weighted voting scheme to determine the final label for *y*.

3.4.1. Quality. The performance of anomaly detection is related to the clustering quality of the uniform model. A higher clustering quality typically corresponds to a better description of normal data behavior and hence leads to a better performance of anomaly detection algorithm. In our work, four internal measures are used to evaluate the clustering quality of uniform model. All measures are usually used when the true data labels are unknown. They include Silhouette index, Davies-Bouldin, Dunn index [1] and Calinski-Harabasz [10].

Silhouette index [1], is a composite index reflecting the compactness and separation of clusters.

The Silhouette index of the *i-th* data record in the cluster $Cj = \{ x_{j1}, x_{j2}, \ldots, x_{jn}\}, j = 1, \ldots, N_c$, is defined as

$$SI(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \quad (1)$$

where *a(i)* is the average distance between the i^{th} data record and all the records included in *Cj,* and *b(i)* is the minimum average distance between the i^{th} data record and all of the records that are located in other clsuters C_k, $k = 1, \ldots, N_c$, $k \neq j$. A larger Silhouette value indicates a better quality of a clustering result.

Davies-Bouldin Index [1] was used as the primary measure of merit. It is the average similarity between each cluster and aims at identifying if the clusters are compact and well separated. It is defined as

$$DB = \frac{1}{N_c}\sum_{j=1}^{N_c} \max_{j \neq k}((S_c(k) + S_c(j)) / d_{ce}(k, j)) \quad (2)$$

where S_c and d_{ce} denote the centroid intra-cluster and inter-cluster distances respectively. A low Davies-Bouldin index value indicates good cluster structures.

Calinski-Harabasz index [10] is the pseudo F statistic, which evaluates the clustering results by looking at how similar the objects are within each cluster and how well the objects of different clusters are separated. For the *i-th* cluster C_i, Calinski Harabasz index is defined as

$$CH_i = \frac{trB_i / (N_i - 1)}{trW_i / (n - N_i)} \quad (3)$$

where $\boldsymbol{B}_i$ and $\boldsymbol{W}_i$ are $p*p$ matrices of between and within N_i-clusters, *tr* denotes the trace of a matrix, which means the sum of the diagonal entries.

Dunn index [1] was used to identify sets of clusters that are compact and well separated. It is defined as

$$DU = \min_{1 \le i \le N_c}\left(\min_{1 \le j \le N_c, j \neq i} \left(\frac{d_{ce}}{\max_{1 \le k \le N_c} S_c}\right)\right) \quad (4)$$

where S_c and d_{ce} again denote the centroid intra-cluster and inter-cluster distances respectively. Large Dunn ndex values indicate the presence of compact and well-separated clusters.

The final quality weight $\boldsymbol{Quality}_i$ of the model *i* is computed by all the clusters' average values. Due to the difference of four internal measures, we compute four quality weights by each of them.

$$Quality_i = \frac{1}{N_c}\sum_{j=1}^{N_c} Q_j$$

where N_c is the total cluster number learned by uniform *k*-means model *i*, Q_j denote *j*-th cluster's quality value with any quality index described in section 3.4.1.

3.4.2. Diversity. Since diversity plays a significant role in combining prediction models, this method also uses three metrics to measure the diversity of local anomaly detection models. The metrics include Adjusted Rand index, Jaccard index [17], Fowlkes-Mallows (FM) index [10].

Let *A* and *B* correspond to two anomaly detection models that provide prediction label vectors $\lambda^{(a)}$ and $\lambda^{(b)}$. Assume there are *N* data records in the test data set *T*, the model *A* predicts N_1^a data records as normal and N_2^a data records as anomalies, and the model *B* predicts N_1^b data records as normal and N_2^b data records as anomalies. Assume also that N_{22}^{ab} denotes the number of data records that are predicted as anomalous by both models *A* and *B*. These three diversity measures are defined using the following equations:

Adjusted Rand index

$$t_1 = \sum_{i=1}^{2}\binom{N_i^a}{2}, t_2 = \sum_{i=1}^{2}\binom{N_i^b}{2}, t_3 = \frac{2t_1 t_2}{N(N-1)}$$

$$AR(A, B) = \frac{\sum_{i=1}^{2}\sum_{j=1}^{2}\binom{N_{ij}^{ab}}{2} - t_3}{\frac{1}{2}(t_1 + t_2) - t_3} \quad (5)$$

Jaccard index

$$JA(A, B) = \frac{N_{11}^{ab} + N_{22}^{ab}}{N_{11}^{ab} + N_{22}^{ab} + N_{12}^{ab} + N_{21}^{ab}} \quad (6)$$

Fowlkes-Mallows (FM) index.

$$FM(A, B) = \frac{(N_{11}^{ab} + N_{22}^{ab})}{2\sqrt{\sum_{i=1}^{2}\binom{N_i^a}{2}\sum_{j=1}^{2}\binom{N_j^b}{2}}} \quad (7)$$

For these three indices, the higher the index, the higher the diversity between the models.

As we all known, if the local models have more diversity then we can get more accuracy in anomaly detection by combining their results. Furthermore, for each anomaly detection model M_j, the average diversity $\boldsymbol{\sigma_k}$ is computed as

$$\sigma_i = \frac{1}{n-1} \cdot \sum_{l=1, l \neq i}^{n} Index(\lambda^{(k)}, \lambda^{(l)}), \quad i = 1, 2, \ldots, n$$

where ***Index(A, B)*** could be any of the diversity metrics defined in section 3.3.2, and *n* is the total number of local anomaly detection models. The value of σ_i indicates the diversity of this model; the larger σ_i the more diverse the model is.

In our studies, we found that for a better model combination not only there is a need for the higher diversity but also a need for a large spread of individual diversities. If there is some very bad model, it also can have very high diversity, so the condition should be higher diversity and lager spread of diversity. We use the standard deviation to present the spread of diversity, so we can get the over measure of diversity by taking the average diversity of m local models.

$$\sigma = \frac{1}{n}\sum_{i=1}^{n} \sigma_i$$

Then the measure can be adjusted as

$$Div_i = \frac{1-\sigma_i + \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(1-\sigma-\sigma_i)}}{2} \quad (8)$$

Therefore, when we consider the diversity of anomaly detection models, we should assign the weights based on $\boldsymbol{Div_i}$.

3.4.3. Combination. Finally, the weights (considering both quality and diversity) of local model *i* are defined as:

$$\delta_i = \alpha * Quality_i + (1-\alpha) * Div_i$$

where the first term of the equation in the right weigh the qualities of the distributed anomaly detection models while the second term measures the diversities. α indicates how much emphasis we put on both factors. In our work, α is set to be 0.5 since there is no clear evidence to give more emphasis on one of the factors. The future work is to study the sensitivity of α, to get the better choice for it. The value δ_i indicates the qualified information that the local model M_i shares with other anomaly detection models. The larger δ_i, the more qualified information is shared from individual models M_i. Therefore, the best local model with high quality and diversity should have the largest value δ_i.

Finally, the combination model gives the final result as label vector $\boldsymbol{\lambda} = \{\lambda_1, \lambda_2, \ldots, \lambda_m\}$ and anomaly score vector $\boldsymbol{S} = \{s_1, s_2, \ldots, s_m\}$ by using the weight voting considered both quality and diversity of local models. The predicted label vector and anomaly score vector of local model *i* are denoted by $\boldsymbol{\lambda}^{(i)} = \{\lambda_1^{(i)}, \lambda_2^{(i)}, \ldots, \lambda_m^{(i)}\}$ and $\mathbf{S}^{(i)} = \{s_1^{(i)}, s_2^{(i)}, \ldots, s_m^{(i)}\}$.For all prediction anomaly scores vector $\boldsymbol{S} = \{\boldsymbol{S}^{(1)}, \boldsymbol{S}^{(2)}, \ldots, \boldsymbol{S}^{(n)}\}$ provided by individual models with weight δ_i. respectively.

$$S = \frac{1}{n}\sum_{i=1}^{n}\delta_i S_i$$

3.5. Evaluation methods

Anomaly detection algorithms are typically evaluated using the detection rate, the false alarm rate, and the ROC curves [9]. In order to define these metrics, let's look at a confusion matrix, shown in Table I. In the anomaly detection problem, assuming class "C" as the anomaly, and "NC" as a normal (majority) class, there are four possible outcomes when detecting anomalies (class "C"), namely true positives (TP), false negatives (FN), false positives (FP) and true negatives (TN).

TABLE I
CONFUSION MATRIX DEFINES FOUR POSSIBLE SCENARIOS WHEN CLASSIFYING ANOMALOUS CLASS "C"

	Predicted C	**Predicted NC**
Actual C	True Positives (TP)	False Negatives (FN)
Actual NC	False Positives (FP)	True Negatives (TN)

From Table I, *recall*, *precision* and *F-value* may be defined as follows:

$$Precision = TP / (TP + FP)$$
$$Recall = TP / (TP + FN)$$
$$F\text{-}value = \frac{(1+\beta^2)\cdot Recall \cdot Precision}{\beta^2 \cdot Recall + Precision},$$

where β corresponds to relative importance of *precision* vs. *recall* and it is usually set to 1. The main focus of all learning algorithms is to improve the *recall*, without sacrificing the *precision*. However, the *recall* and *precision* goals are often conflicting and attacking them simultaneously may not work well, especially when one class is rare. The *F-value* incorporates both *precision* and *recall*, and the "goodness" of a learning algorithm for the anomalous class can be measured by the *F-value*. While ROC curves represent the trade-off between values of TP and FP, the F-value basically incorporates the relative effects/costs of *recall* and *precision* into a single number.

To confirm the effectiveness of weighted voting for combining unsupervised anomaly detection models, we train the global model by collecting all the data record together to compare with the combination models result and use the true label of the data records to compute the *F-value* for evaluation.

4. Experiments

Our experiments were performed on synthetic data and several real life datasets, all of which are summarized in Table II. In all the experiments, we used only the unlabeled data for building local anomaly detection models and assumed that we have no knowledge about failure and other anomalous behaviors. The anomalies' labels were only used in evaluating the final anomaly detection performance. In the first step, all 3 unsupervised learning algorithms (K-means, SOM and DBSCAN) were used to build the anomaly detection models in a distributed dataset. Then, we select the best qualified model according the model quality which is computed by the Quality Measures and put it into the knowledge base. In particular, when the best model was built by density-based algorithm such as DBSCAN, we should use the transformation approach proposed in the knowledge reusing section to transfer the model into a new applicable model. In the second step, we employed all anomaly models in the knowledge base to the test dataset and obtained a label vector as well as an anomaly score vector. The next step is to compute the diversity of each model with diversity measures. The final step is to compute the combination weight by considering both quality and diversity of each model and combining the anomaly score vector. Then we found a final label for each test data point.

Performances of all the anomaly detection models are evaluated by the comparison with global anomaly detection model, which is built by collecting all the

data together. However, a global model is not available in most real-life cases. The true anomalies' labels of test data are used only in computing F-measure of both models.

4.1. Experiments on Synthetic Data Sets

Our first experiment was performed on synthetic data consisting of five datasets that were used for building our unsupervised anomaly detection models. The test dataset is generated by the same emulator. Both have about 1.96% percentage of anomalies.

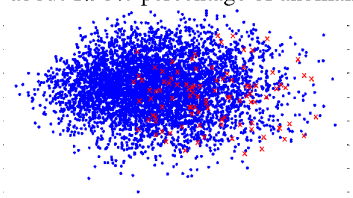

Figure 4. Synthetic data set represented using two features obtained through PCA (Principal Component Analysis) (blue points represent normal behavior, while red crosses represent anomalies).

Each distributed dataset has four continuous features and four discrete features. All together, there are 5000 data records that correspond to normal behavior and 100 data records corresponding to anomalies. Normal behavior (represented by blue points in Figure 4) is modeled as a skewed distribution with exponential drop-off, while the anomalies (red crosses in Figure 4) that correspond to a background noise are modeled by a uniform distribution. The whole dataset was randomly divided into six parts, where five parts are considered to be five distributed datasets, while the sixth one is used for testing.

We build unsupervised anomaly detection models on all five disjoint training datasets and store the best model into the knowledge base. Afterwards, we apply all models in the knowledge base to the test dataset and then employ proposed combination techniques to combine their prediction anomaly scores.

Table III presents the performance by F-measure used to compare with the global anomaly detection model. The global model can be built only when all the distributed datasets are collected together which is not always applicable due to the privacy or property protection policy. Here, we study different combinations of model quality measures as well as diversity measures. The high F-measure means the model has good performance both in precision and in recall. From the experiment results on synthetic datasets, we can find that a better quality often comes with a larger diversity. Therefore, a higher model weight in the combination will lead to a better performance of the combination anomaly detection model.

Table IV reports the quality value in the knowledge base after we select the best unsupervised anomaly detection model in each site and transform them into uniform format models. The quality of a model is computed by equation (1), (2), (3) and (4) individually, and the value is the average of all clusters in the uniform model. Large values of Silhouette, Calinski Harabasz and Dunn indicate better qualities of a local unsupervised anomaly detection model. For Davies-Bouldin the low value indicates a good cluster structure which can be considered a metric of better qualified model.

Table V gives the diversity value of all local models with different measures .Values in Table V are computed by equation (5), (6), (7) and (8). For comparison, we normalize every value into (0,1) where all values sum to one. For the three indexes, a large value indicates the presence of high diversity.

4.2. Experiments on Real Life Data Sets

All real life datasets used in our experiments have been widely used by other researchers for anomaly detection. Table II gives a summery of them. KDD CUP 1999 dataset includes a set of 41 features derived from each connection and a label that specifies the status of a connection record which is either normal or presents a specific attack type. Attacks fall into four main categories: DoS (Denial of Service), R2L (Remote to Local), U2R (User to Root) and Probe. We selected U2R, which covers only 246 instances, to detect the smallest intrusion class. Since the anomalies are detected as deviations from the normal behavior, we modified the original dataset (311029 records), and collected only normal class (60593) and U2R attack records for the experiment. Considering the large-scale volume of the KDD CUP dataset, we divided the whole experiment dataset into 10 parts, 9 of them are considered to be 9 distributed local sites, and the other 1 is used for testing. For satimage dataset we chose the smallest class to represent anomalies and collapsed the remaining classes into one class as was done in [7]. This procedure gave us a skewed 2-class dataset, with 5809 majority class examples and 626 minority class examples (anomalies). When performing experiments on mammography [7] and rooftop [19] datasets, we did not change any class distribution.

Like the synthetic experiment procedure described above, local unsupervised anomaly detection model are employed and five best models are stored into the knowledge base. Following that, we get the diversity values and accordingly compute the combination weights. Finally, we distinguish the anomalies in test data by gathering all the anomaly detection model knowledge from all sites.

Our experiment was also performed on the flight record data sets obtained from the NASA Advanced Diagnostics and Prognostics Testbed (ADAPT) project. ADAPT, a facility developed at NASA, aims for supporting the development of diagnostic and prognostic models, for evaluating advanced warning

systems, and for testing diagnostic tools and algorithms against a standardized testbed. The facility's hardware consists of an electrical power system with components for power generation, storage, and distribution. Over a hundred sensors report the status of the system to the test article that monitors the health status of the system. The testbed provides a controlled environment to inject failures, either through software or hardware, in a repeatable manner.

The data that we have used simulates an electrical power subsystem (EPS) in which faults have been injected by manual or software means. The dataset is available as a set of conducted experiments. In each experiment, testers inject an anomalous condition within a system to cause the system to get into a faulty state. We have used datasets from five experiments where five different types of system faults were injected. The entire dataset had 1029 flight data records collected from 194 sensors, where 555 data records corresponded to normal operation, and 474 data records corresponded to different failures. We extracted 90% of data records with normal operation and injected failures, and then we randomly split them into five distributed datasets. Each of the datasets had 46% abnormal data records; they are used to construct local anomaly detection models by unsupervised learning algorithms. The left dataset is used for testing; it remains 46% abnormal data records corresponded to failures (50 data records).

For example, Figure 5 and Figure 6 present the process, where a fault was injected into the system at around 58.5 second, and then the sensor output for both Light Intensity and Fan Speed changed abnormally. The fault is described as the antagonist commanded relay EY141 sensor, which connects the battery to the load bank, to open and thus break the circuit. This was one of the faults detected by all anomaly detection models.

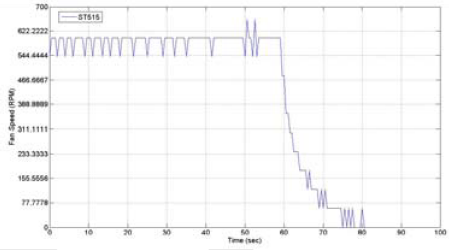

Figure 5. Fan Speed vs. Time

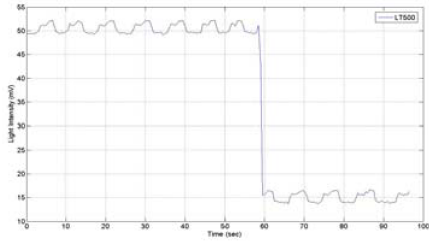

Figure 6. Light Intensity vs. Time

We build unsupervised anomaly detection models on all five disjoint training datasets, apply all the best models from all sites on the test dataset, and finally use our combination techniques to combine their prediction anomaly scores. Similar to experiments on synthetic data sets, we first report the quality values of models in the knowledge base and then give the diversity value of each model. Finally, we give the performance in F-measure to compare with a global anomaly detection model. From the result we can see the best model is not adhered to any unsupervised learning algorithm, regardless of the different distributions of data and different shapes or sizes of normal behavior clusters. Particularly, when the local model has various qualities, the combination method can improve the performance of anomaly detection. Frankly, the combination could potentially have a global point of view by combining the knowledge (i.e., models) learned from all distributed datasets. In general, the combined model provides a comparable performance even when the global model is not available. Please note that, the assumption of an available global model is not always true and here we introduce the global model only for evaluating our combination techniques.

5. Conclusion

A general framework for unsupervised distributed anomaly detection was proposed. It is intended to efficiently learn stable anomaly detection models over large and distributed datasets that cannot be merged into a single one. Experimental results on synthetic and real-life datasets indicate that the proposed techniques for distributed anomaly detection can effectively achieve the same or even better performance, compared to a global anomaly detection model built from a centralized data site. One of future works is to fully characterize the proposed method especially in a distributed environment with heterogeneous databases. New algorithms for selectively combining anomaly detection models from multiple heterogeneous sites with different distributions are worth considering. It would also be interesting to examine the performance and the scalability against the influence of the larger number of local sites and their sizes.

6 Acknowledgements

This work was supported by NASA under award NNX08AC36A.

References

[1] N. Bolshakova & F. Azuaje. Cluster validation techniques for genome expression data, *Signal Processing*. 83(4):825-833, 2003.
[2] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123-140, 1996.
[3] L. Breiman. Arcing classifiers. *Annals of Statistics*, 26(3):801-849, 1998.
[4] L. Breiman. Random forests. *Machine Learning*, 45(1):5-32, 2001.
[5] M. Ester, H. Kriegel, J. Sander, X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise", proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining, Portland, OR, 1996, p. 226
[6] V. Chatzigiannakis, A. Lenis, C. Siaterlis, M. Grammatikou, D. Kalogeras, S. Papavassiliou, V. Maglaris. Distributed network monitoring and anomaly detection as a GRID application. In *Proc. HPOVUA* 2005.
[7] N. V. Chawla, A. Lazarevic, Lawrence O. Hall, Kevin Bowyer. SMOTEBoost: Improving the Prediction of Minority Class in Boosting. In *Proc. PKDD 2003*.
[8] K. J. Cherkauer. Human expert-Level performance on a scientific image analysis task by a system using combined artificial Neural Networks. *Working Notes of the AAAI Workshop on Integrating Multiple Learned Models*, pp.15-21, 1996.
[9] F. Provost & T. Fawcett. Robust Classification for Imprecise Environments, Machine Learning, vol. 42, pp. 203-231, 2001.
[10] S. Dudoit & J. Fridlyand. A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3(7):0036.1-21, 2002.
[11] G. Folino, C. Pizzuti, G. Spezzano. GP Ensemble for Distributed Intrusion Detection Systems. 3683:54-62, 2005.
[12] Y. Freund & E. R. Schapire. A decision theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119-139, 1997.
[13] S. Gunter & H. Bunke. Ensembles of classifiers for handwritten word recognition. *Int'l Journal on Document Analysis and Recognition*, 5(4):1433-2833, 2003.
[14] J. A. Hartigan & M. A. Wong. A K-Means Clustering Algorithm. *Applied Statistics*, 28(1): 100-108, 1979.
[15] I.T. Jolliffe. Principal Component Analysis. *Springer Series in Statistics*, 2nd ed., XXIX, 487, pp. 28, 2002.
[16] T. Kohonen. Self-Organizing Maps. *Springer Series in Information Sciences*, vol. 30. 3rd Extended Ed., 2001.
[17] L.i. Kuncheva & S T. Hadjitodorov. Using diversity in cluster ensembles. *IEEE Int'l Conf. on Systems, Man and Cybernetics*, 2:1214-1219, 2004.
[18] A. Lazarevic & V. Kumar. Feature bagging for outlier detection. In *Proc. SIGKDD 2005.*
[19] N. M. Maloof, P. Langley, T. Binford, R. Nevatia, S. Sage. Improved Rooftop Detection in Aerial Images with Machine Learning. *Machine Learning*, 53:157-191, 2003.
[20] E. Kong & T. Dietterich. Error-Correcting Output Coding Corrects Bias and Variance. In Proc. ICML 1995, pp.313-321.
[21] A. Strehl, J. Ghosh, C. Cardie. Cluster Ensembles-A Knowledge Reuse Framework for Combining Multiple Partitions. JMLR, 3:583-617, 2003.
[22] Y.-F. Zhang, Z.-Y. Xiong, X.-Q. Wang. Distributed intrusion detection based on clustering. In *Proc Int'l Conf. on Machine Learning & Cybernetics*, 4:2379-2383, 2005.
[23] Z.-H. Zhou & Y. Jiang. Medical diagnosis with C4. 5 rule preceded by artificial neural network ensemble. *IEEE Trans. Info. Tech. in Biomedicine*, 7(4):37-42, 2003.
[24] D.M.J. Tax, One-class classification. Ph.D. thesis, Delft University of Technology, June, 2001.
[25] D. Karger, P. Klein, R. Tarjan. A randomized linear-time algorithm to find minimum spanning trees. *Journal of the ACM*, 42(2):321-328, 1995.
[26] P. Chan & S. Stolfo. On the Accuracy of Meta-learning for Scalable Data Mining. *Journal of Intelligent Integration of Information*, 1998.
[27] W. Fan, S. Stolfo, J. Zhang. The Application of AdaBoost for Distributed, Scalable and On-line Learning. In *Proc. SIGKDD 1999*, pp. 362-366.
[28] A. Lazarevic & Z. Obradovic. The Distributed Boosting Algorithm. In *Proc. SIGKDD 2001.*

7. Appendix

TABLE II
SUMMARY OF DATA SETS USED IN EXPERIMENTS

Dataset	Modification made in the data set	Size of Dataset	Numbers of features		Number of anomalies (rare class records)	Percentage of anomalies
			Continuous	Discrete		
Synthetic	-	5100	4	4	100	1.96%
KDDCUP 1999	U2R vs. normal	60839	34	7	246	0.4%
Mammography	-	11183	6	0	260	2.32%
Rooftop	-	17829	9	0	781	4.38%
Satimage	Smallest class vs. rest	16435	36	0	626	9.73%
NASA data	-	1029	194	0	474	46%

TABLE III

F-MEAUSURE COMPARISON FOR COMBINATION MODEL AND GLOBAL MODEL ON ALL DATA SETS

Dataset	Model	Silhouette index AR	Silhouette index JA	Silhouette index FM	Davies-Bouldin AR	Davies-Bouldin JA	Davies-Bouldin FM	Calinski-Harabasz AR	Calinski-Harabasz JA	Calinski-Harabasz FM	Dunn index AR	Dunn index JA	Dunn index FM
Synthetic	CoM	0.9843	0.9873	0.9867	0.9885	0.9836	0.9836	0.9861	0.9836	0.9861	0.9824	0.983	0.985
	GlM	0.987(DBSCAN)				0.973(SOM)				0.976(K-means)			
KDD	CoM	0.9963	0.9965	0.9963	0.9968	0.9968	0.9970	0.9963	0.9968	0.9968	0.9963	0.9968	0.9965
	GlM	0.99667 (DBSCAN)				0.99632 (SOM)				0.99489 (K-means)			
Mg	CoM	0.9795	0.9723	0.9783	0.9717	0.9759	0.9686	0.9767	0.9677	0.9669	0.9791	0.9739	0.9783
	GlM	0.97949(DBSCAN)				0.98033(SOM)				0.97932(K-means)			
Rooftop	CoM	0.9656	0.9653	0.9653	0.9648	0.9650	0.9650	0.9651	0.9650	0.9705	0.9624	0.9625	0.962
	GlM	0.97663(DBSCAN)				0.96836(SOM)				0.96283(K-means)			
Satimage	CoM	0.9196	0.9289	0.933	0.9333	0.9368	0.9272	0.9325	0.9338	0.9285	0.9196	0.9289	0.933
	GlM	0.93294(DBSCAN)				0.9271(SOM)				0.9306(K-means)			
NASA	CoM	0.65	0.7373	0.66	0.6326	0.65	0.632	0.7655	0.6294	0.6764	0.6326	0.6532	0.6567
	GlM	0.70518(DBSCAN)				0.70368(SOM)				0.69214(K-means)			

Legend: KDD = KDDCUP 1999, Mg = Mammo-graphy , CoM = Combined Model(The model combined by distributed models), GlM = Global Model(The model built by collecting all the distributed data sets, the global model is not available in most cases, here we build it just for performance evaluation), AR = Adjusted Rand index, JA = Jaccard index, FM = Fowlkes-Mallows index

TABLE IV

QUALITY -MEAUSURE OF 9 DISTRIBUTED LOCAL MODELS ON 5 DATASETS

Dataset	Quality	Model_1	Model_2	Model_3	Model_4	Model_5
Synthetic	Silhouette Index	0.2219	0.2125	0.2209	0.2203	0.2466
	Davies-Bouldin	0.7583	0.7429	0.7481	0.7613	0.6955
	Calinski-Harabasz	0.2021	0.1856	0.2009	0.1978	0.2134
	Dunn index	2.441	2.349	2.312	2.248	2.632
Mammography	Silhouette Index	0.6353	0.7489	0.8058	0.7543	0.7893
	Davies-Bouldin	1.421	0.3505	0.2352	0.2336	0.2258
	Calinski-Harabasz	0.1311	0.2244	0.2906	0.1811	0.1726
	Dunn index	0.2357	1.8841	3.5097	1.4745	2.0781
Rooftop	Silhouette Index	0.4078	0.3932	0.4093	0.4101	0.5044
	Davies-Bouldin	1.1679	1.3058	1.4902	1.0549	0.2631
	Calinski-Harabasz	0.2146	0.2203	0.2323	0.1701	0.1625
	Dunn index	0.1072	0.0943	0.1269	0.1090	4.1635
Satimage	Silhouette Index	0.7471	0.7704	0.5645	0.7543	0.7990
	Davies-Bouldin	0.3015	0.1763	0.5921	0.2036	0.2483
	Calinski-Harabasz	0.2284	0.1907	0.0353	0.2174	0.3280
	Dunn index	1.6205	1.9307	1.4328	1.4745	2.985
NASA data	Silhouette Index	0.4567	0.4023	0.3314	0.3647	0.4902
	Davies-Bouldin	0.5011	0.5018	0.5502	0.4313	0.4279
	Calinski-Harabasz	0.2378	0.2384	0.2135	0.0798	0.2302
	Dunn index	2.8555	3.6236	3.4063	2.369	2.9903

TABLE V

DIVERSITY-MEAUSURE OF 5 DISTRIBUTED LOCAL MODELS ON 5 DATA SET

Dataset	Diversity	Model_1	Model_2	Model_3	Model_4	Model_5
Synthetic	Adjusted Rand	0.1955	0.1817	0.1997	0.1736	0.2494
	Jaccard Index	0.2084	0.1973	0.1618	0.2702	0.1621
	Fowlkes-Mallows	0.1900	0.1879	0.1010	0.2666	0.2543
Mammography	Adjusted Rand	0.2753	0.1276	0.2550	0.2496	0.0922
	Jaccard Index	0.1595	0.1821	0.2081	0.2475	0.2025
	Fowlkes-Mallows	0.1528	0.2461	0.2351	0.2300	0.1359
Rooftop	Adjusted Rand	0.2186	0.1988	0.2002	0.1831	0.1991
	Jaccard Index	0.2189	0.2004	0.2013	0.1852	0.1940
	Fowlkes-Mallows	0.2191	0.2008	0.1994	0.1854	0.1950
Satimage	Adjusted Rand	0.1577	0.2188	0.2487	0.2351	0.1395
	Jaccard Index	0.0916	0.2521	0.2507	0.1766	0.2287
	Fowlkes-Mallows	0.2705	0.1006	0.2261	0.1496	0.2530
NASA data	Adjusted Rand	0.2997	0.1642	0.1028	0.1325	0.3006
	Jaccard Index	0.3555	0.1514	0.2043	0.0076	0.281
	Fowlkes-Mallows	0.2431	0.2053	0.2050	0.1160	0.2304

TABLE VI

QUALITY -MEAUSURE OF 9 DISTRIBUTED LOCAL MODELS ON KDD CUP 1999 DATA

Quality / Model	Model_1	Model_2	Model_3	Model_4	Model_5	Model_6	Model_7	Model_8	Model_9
Silhouette Index	0.5988	0.6372	0.6590	0.6617	0.6309	0.647	0.6188	0.6364	0.6067
Davies-Bouldin	0.7484	0.5703	0.5351	0.6101	0.5032	0.6065	0.4559	0.5885	0.4416
Calinski-Harabasz	0.0778	0.1038	0.1081	0.1410	0.132	0.0982	0.1010	0.139	0.0988
Dunn index	0.5492	0.5310	0.5834	0.6240	0.5445	0.5768	1.5571	0.58112	1.7257

TABLE VII

DIVERSITY-MEAUSURE OF 9 DISTRIBUTED LOCAL MODELS ON KDD CUP 1999 DATA

Diversity / Model	Model_1	Model_2	Model_3	Model_4	Model_5	Model_6	Model_7	Model_8	Model_9
Adjusted Rand	0.1411	0.0629	0.1274	0.1052	0.0893	0.1431	0.0982	0.1108	0.1217
Jaccard Index	0.0910	0.0995	0.0832	0.1078	0.1089	0.1127	0.1434	0.1048	0.1482
Fowlkes-Mallows	0.1277	0.0922	0.1105	0.1091	0.1044	0.0985	0.1194	0.1087	0.1291