Expert-Based Fusion Algorithm of an Ensemble of Anomaly Detection Algorithms

Esther David[1], Guy Leshem[1], Michal Chalamish[1], Alvin Chiang[2], and Dana Shapira[3]
astrdod@acad.ash-college.ac.il, leshemg@cs.bgu.ac.il, {michal.chalamish,alvin.chiang.180,shapird}@gmail.com

[1] Department of Computer Science, Ashkelon Academic College, Ashkelon, Israel
[2] Department of Computer Science and Information Engineering, National Taiwan University of Science and Technology, Taiwan
[3] Department of Computer Science, Ariel University, Israel

Abstract. Data fusion systems are widely used in various areas such as sensor networks, robotics, video and image processing, and intelligent system design. Data fusion is a technology that enables the process of combining information from several sources in order to form a unified picture or a decision. Today, anomaly detection algorithms (ADAs) are in use in a wide variety of applications (e.g. cyber security systems, etc.). In particular, in this research we focus on the process of integrating the output of multiple ADAs that perform within a particular domain. More specifically, we propose a two stage fusion process, which is based on the expertise of the individual ADA that is derived in the first step. The main idea of the proposed method is to identify multiple types of outliers and to find a set of expert outlier detection algorithms for each type. We propose to use semi-supervised methods. Preliminary experiments for the single-type outlier case are provided where we show that our method outperforms other benchmark methods that exist in the literature.

Keywords: Anomaly Detection Algorithms, Cluster, Ensemble, Outlier, Scores.

1 Introduction

According to the current state of the art, a wide range of anomaly detection algorithms (ADA) are proposed in various disciplines such as statistics, data mining, machine learning, information theory and spectral decomposition [1], which are also known as outlier detection algorithms [2]. Given the decisions of multiple ADAs, which all operate in the same environment, in this research we aim to confront the challenge of integrating the individual decisions into a final unified representative decision. Specifically, we are interested in non-stationary (i.e., unstable and unexpected) environments where the algorithms improve the decision making process using partial feedback given to them sporadically (that is, at unknown times) and the correctness of the feedback is also unknown [18].

S.-M. Cheng and M.-Y. Day (Eds.): TAAI 2014, LNAI 8916, pp. 114–123, 2014.

The need for such a fusing system stems from the fact that there are many ADAs that suffer from a certain percentage of error. By fusing and aggregating the outputs of multiple ADAs we aim to minimize the error percentage as much as possible. In other words we intend to maximize the recall rate of the process. An illustrative example is the case where a computer system administrator aims to identify and block any offensive attack on his computer system or any malicious program [19–22]. Another noncriminal example of an anomaly detection scenario is the case where countries with high typhoon vulnerability aim to identify approaching storms and to act in such a way that will minimize potential damage [23].

An ideal ADA satisfies the conditions of (i) having a True Positive Rate (TPR) equal to 1 (the TPR indicates the portion of accurate positive instances of all positive instances that were classified as positive; this measurement is also known as the recall rate or alternatively the sensitivity rate) and (ii) having a False Positive Rate (FPR) equal to 0 (the FPR is the portion of positive instances that were misclassified as positive of all positive classified instances (the FPR is also known as the false alarm rate).

Assuming that a set of ADAs do not overlap and are independent we may use a simple *OR* operation among them in order to fuse them. Namely, it is sufficient that a single ADA decides a certain input instance is an outlier in order to have the final decision of an outlier. Unfortunately, this assumption is far from applicable. Therefore, in this paper we propose a fusing method that will achieve a false alarm rate smaller than each of individual ADA.

Against this background, in this paper we propose a two phase mechanism. In the first step an offline process will take place to classify all the given ADAs into clusters based on their expertise. In the second phase an online and continuous process will take place in which we aim to fuse the decision of all the ADAs in a way that promotes the expert ADAs that were identified in the previous phase for each given type of outlier.

Next we provide a preliminary experiment that deals with the case of a single type of outlier. Here we focus on a more basic debate that exists in the literature about the process of unifying the scores given by different ADAs taken from different scales and ranges termed the normalization phase. We show a way to overcome the problem of normalization by using ranking in its place, which was found to perform better than the normalization.

The paper is structured as follows. In section 2 we describe the current state of the art. In section 3 we provide details about the general fusing structure. Then, in section 4 we describe our proposed expertise based fusing mechanism. In section 5 we describe our simulation and provide initial results. Finally we conclude in section 6 and discuss future work.

2 Related Work

Information or data fusion has been widely researched in the last decade [3, 4]. According to Ahmed and Pottie [4], data fusion is the process by which a data

from a multitude of sensors is used to yield an optimal estimate of a specified state vector pertaining to the observed system, whereas sensor administration is the design of communication and control mechanisms for the efficient use of distributed sensors with regard to power, performance, reliability, etc. In this paper we deal with a special case of information fusion which is decision fusion [5]. In information fusion the goal is to fuse complex noisy information provided by multi-sources or multi sensors, of distributed networks, to produce a single unified information model (e.g., vision systems, sonar, robotic platform, weather prediction [4]). However, in our case of decision fusion, we aim to integrate the multiple decisions we receive from the ADAs into a single decision that will be more accurate than the decision of each ADA itself. Each of the ADAs provides a decision in a binary form or in a score form. The special characteristics of our decision fusion problem make most of the available information fusion methods irrelevant.

Next we provide some background on the way an ensemble of methods of the same type can be used to improve the efficiency and correctness of the decision making as we suggest by the term fusion. The first use of such an ensemble was in the classification domain. Building an ensemble of single classifiers to gain an improved effectiveness has a rich tradition and sound theoretical background [6–8]. The idea of using an ensemble can also be found in the clustering domain. [6, 9]. Next, in the domain of anomaly detection or also known as outlier detection algorithms, most of the efforts have been invested in the development of new methods for outlier detection. Only very few preliminary studies have attempted to use the notion of ensemble in order to compose a group of outlier detection algorithms in order to create a meta outlier that will perform better [10–13]. Going back to ensemble in classification, the main insight from using ensemble in the classification domain is that for an ensemble to outperform each of the individual classifier requires that they are (i) accurate (i.e., at least better than random); (ii) diverse (i.e., making different errors with new instances). These conditions are necessary and sufficient. When these conditions are satisfied the majority voting rule of the ensemble also may be correct [6]. In conclusion, the rule of thumb in constructing a meaningful ensemble is to choose members that make uncorrelated errors. We have followed this principle in composing our ensemble of outlier detection algorithms/ADAs.

Some information fusion methods are based on weighting techniques of varying degrees of complexity [4]. For example, Berger [14] discusses a majority voting technique based on a probabilistic representation of information. In our work we also consider a weighting method that is basically based on expertise associated with the multiple ADAs.

3 Fusion Structure

In this paper we assume that there is a set of N ADAs whereby each monitors the same system, aiming to detect an outlier event or data. Each individual ADA performs based on different methods. According to Chandola [15] three types of anomaly detection algorithms exist, which were defined by him as follows:

1. **Point Anomalies:** If an individual data instance can be considered as anomalous with respect to the rest of data, then the instance is termed a point anomaly. This is the simplest type of anomaly and is the focus of the majority of research on anomaly detection (e.g. a fraudulent credit card transaction).
2. **Contextual Anomalies:** The anomalous behavior is determined using the values for the behavioral attributes within a specific context. For example, suppose an individual usually has a weekly shopping bill of 100 except during the Christmas week, when it reaches 1000. A new purchase of 1000 in a week in July will be considered a contextual anomaly, since it does not conform to the normal behavior of the individual in the context of time even though the same amount spent during the Christmas week would be considered normal.
3. **Collective Anomalies:** If a collection of related data instances is anomalous with respect to the entire data set, it is termed a collective anomaly .e.g. a low value for an abnormally long period of time where the low value is not, in itself, anomalous or a typical Web-based attack by a remote machine followed by copying of data from the host computer to a remote destination via ftp.

In our research we will focus on ADAs of the first type i.e., point anomaly detection.

This group of ADAs is also termed an ensemble of ADAs. From this ensemble we aim to fuse scores/decisions to reach the final score/decision. We assume the input data to be behavioral which is characterized by being temporal and sequential. We define an outlier to be an input instance that substantially differs from previous time series data for which no a priori knowledge exists. An example of an outlier in the Web-based environment is a data package in the flow between two remote computers that contains a malicious attack. Another example in the weather monitoring domain system may be out of range attribute values that may indicate an approaching storm.

The output of each ADA may use a different scoring/ranking range. Therefore, in order to enable a meaningful integration or fusion of these values we first must use a normalization phase that will keep a proportion across the ADAs' scores. The simplest way of bringing outliers scores onto a normalized scale is to apply a linear transformation such that the minimum (maximum) occurring score is mapped to 0 (1). [11]. In the experiment section we show that ranking may replace the normalization and even outperform it.

4 Expertise Based Fusion Algorithm

In this section we present our method for fusing the score/decision of an ensemble of ADAs. Our leading/base assumption is that there might be multiple types of outliers and that there may be some ADAs that will be experts in the detection of a certain type or multiple types of outliers, but with a very low prediction ability regarding other types of outliers.

The motivation for the assumption of having multiple types of outliers in a certain monitored data, stems from the domain of web-based attacks from remote computers that can be for example of multiple types of Trojans (we have a huge dataset that includes for example temporal data flow consisting of around 13 types of Trojans). To this end, we propose a two stage expertise based fusion protocol:

1. **Offline Stage:** Identify groups/clusters of outliers within an initial data set and associate expert ADAs with each of the outlier clusters, based on their classification/decision on the instances of the initial data set. Namely, if an ADA exists, such that in most cases it has correctly classified a certain outlier type than it will be considered an expert for that type of outlier. Moreover a certain ADA may be found to be expert for multiple outlier types. At the end of this phase we should have a set of outlier types and for each such outlier type we should have a list (which might be empty) of ADAs that are associated with it and are assumed to be expert in detecting it.
2. **Online Phase:** For any new given instance, identify its nearest outlier cluster/type, then using an **expertise based weighting function** combine the decisions, in order to reach the final decision/score. The expertise based weighting function aims to promote the decision of the ADAs that were found to be experts for the given instance's type. Thus, we aim to achieve a more accurate decision.

The offline stage may be performed using either supervised or unsupervised methods. The motivation for using an unsupervised offline stage is the common assumption that in some environments (e.g., big data) in which anomaly detection algorithms perform, the anomalies are not expected and are unknown; therefore it is impossible to assume we have tagged or classified data that can be used.

For the supervised case we propose to reveal the list of anomaly types (if available). Next, for each anomaly type and for each instance of it in the available data, compare the score/decision of each member of the ensemble (i.e., an ADA member) to the accurate decision. Each ADA algorithm that was found to have a relatively high performance with regard to a certain outlier type will be referred to as an expert for that particular anomaly type.

For the unsupervised case, on the other hand, the process of identifying experts is much more complicated. In particular, in order to overcome the fact that the initial data set is not classified/tagged we will follow a procedure that was proposed by Schubert et al. [6]. According to the initialization procedure of Schubert in order to identify the anomaly instances we will take the k top scored instances according to each ADA. Next, we collect the instances identified by each ADA (using the union set) to create the group of outliers. Once this initialization classification is derived, we continue with identifying the expertise of each ADA similar to the supervised case.

In the online phase, for each new given instance, E we propose to calculate the nearest anomaly cluster and this similarity measurement s and the specific

most similar cluster SC will define the expertise based weighting function's parameters. Given these notations, the fused score for a certain instance E will be calculated as follows:

FusedScore(E)=s*[average score of expertsADAs(SC)]+(1-s)*[average score of unexpertsADAs(SC)]

5 Preliminary Experiment

In this section we limit ourselves to the case of a single outlier type. For this case we describe the Union Voting Fusion method that basically aims to overcome the normalization issue.

5.1 The Union Voting Fusion Algorithm

The scores from each ADA are converted to rankings. Then these rankings are combined into a (inverse) suspicious score by taking the k-th highest rank from the group of the ADAs. This is interpreted as at least k ADAs having a consensus that the final rankings of the data are not too suspicious. For example, if an instance's final suspicious score is 10, then k of the classifiers agree that this instance deserves to be on a top-10 outlier list.

It is preferable to combine rankings rather than the raw scores due to the fact that the scores cannot be interpreted in the same way. The rankings obtained in this manner are more robust than the individual rankings of the outlier detectors because they are smoothed out.

Table 1. Scores table

Instance	Score 1	Score 2	Score 3
A	0.5 (1^{st})	0.75 (3^{rd})	0.9 (1^{st})
B	0.4 (2^{nd})	0.9 (2^{nd})	0.8 (3^{rd})
C	0.3 (3^{rd})	0.95 (1^{st})	0.85 (2^{nd})
D	0.2 (4^{th})	0.3 (4^{th})	0.2 (5^{th})
E	0.1 (5^{th})	0.1 (5^{th})	0.25 (4^{th})

Example. If we do a "one-vote union", we take the smallest ranking as the ranking of the 3-detector ensemble. i.e.
A : min(1,3,1) = 1
B : min(2,2,3) = 2
C : min(3,1,2) = 1
... and so on

Instances with the score n appeared in the top-n outlier list of at least 1 detector. In other words the "one-union" means that it is sufficient to be considered by at least one ADA classifier.

Similarly, for an "x-vote union" we take the x-th smallest ranking as the score of the ensemble. e.g. Union-2
A : min2(1,3,1) = 1
B : min2(2,2,3) = 2
C : min2(3,1,2) = 2 Points with the score n appeared in the top-n outlier list of at least x detectors.

The ALOI Outlier Data-Set was used for the experiment. Some details are given below:

1. The ALOI [17] dataset is a set of 110250 color images taken from 1000 small objects under varying conditions (i.e., approximately 100 pictures per object).
2. In order to be appropriate for use as an outlier dataset, the ALOI dataset was converted into an RGB histogram form, with 3 bins for each color channel, and the number of images was reduced to 50000, with 1508 outliers.
3. To create these outliers, 1-5 images taken from the photo galleries of 562 objects such that there were a total of 1508 images to be used as the outliers. While the other image galleries were left intact to serve as non-outliers. The result was a dataset of 50000 with a dimensionality of 27.

For our candidate algorithms we used KNN, Aggregated KNN, LOF [24], LDOF[25] and LoOP[11], which all have a single parameter k. k was adjusted from 3 to 30 for a total of 5*28 = 140 candidates.

In the comparison made we compared our proposed "one union vote"(at least one ADA has marked it as an outlier), the "140-union vote" (all the ADAs agree that a certain instance is an outlier, where we use 140 versions of various ADAs), the greedy fusion proposed by [6], and the simple average of all ADAs termed the "Mean Ensemble". We also include the result of initializing the greedy ensemble method using the labels themselves. This is obviously not possible in practice and is done to get the upper bound on performance for benchmarking. The performance of the various methods are measured by the ROC curve. The Receiver-Operator Curve (ROC) graphically displays a classifier's TPR vs. it's FPR as the discrimination threshold is varied. It is often used to compare the goodness of the rankings, scores, or probabilities produced by different classifiers. The curve always starts at the bottom left (0,0) and ends at (1,1) representing the extremes of a threshold so high that no instances will be considered positive, and a threshold so low such that all instances become positive.

The ROC of an ideal classifier reaches TPR=1 when FPR=0 (is tight with the y-axis and the top left corner), implying that there exists a decision threshold where the classes are split perfectly. In terms of rankings, this means all instances of the positive class are ordered before the negative class. The area-under-curve of this ideal ROC curve is the area of the entire plot and is taken to be 1, normalized. A classifier that randomly labels instances as positive or negative will have a ROC curve approaching the diagonal and an AUC of 0.5.

Since there may not be a pre-specified acceptable rate of false positives or a decision threshold, oftentimes the area-under-curve(AUC) is used as a crude way

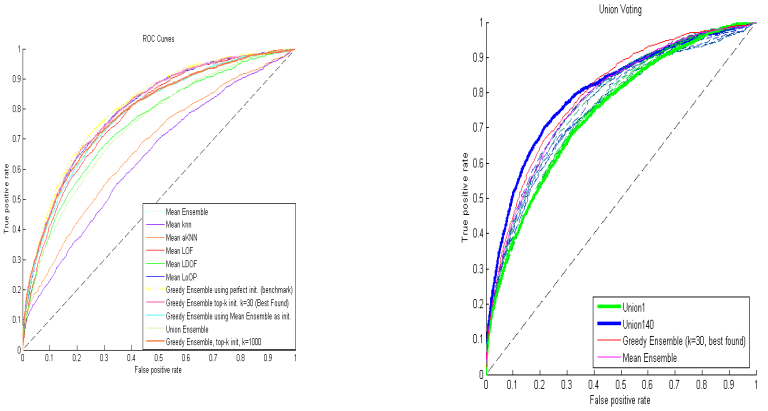

Fig. 1. ROC Curves and Union Voting

Table 2. AUC results for different ensemble

Ensemble System	AUC
KNN Ensemble	0.6388
Agg KNN Ensemble	0.6703
LOF Ensemble	0.7802
LDOF Ensemble	0.7479
LoOP Ensemble	0.7966
Greedy Ensemble, Original Method (using best k found)	0.7971
Greedy Ensemble w/ Perfect Initialization (Labels as Target Vector)	0.8048
Greedy Ensemble using Mean Ensemble as Target Vector	0.7861
Union Voting1	0.7438
Union Voting140	0.8009

of comparing the performance of two classifiers on the whole. Given a positive and negative example, the AUC can be interpreted as the probability of the positive example receiving a higher score than the negative example. From the AUCs values listed in Table. 2 we can see that by requiring a high degree of consensus between the anomaly detector models (Union Voting 140), we prevent any one detector from mislabeling the data as outlier and so reduce the false positive rate. This effect is more convincingly demonstrated in the right side of Fig. 1 where we show the "rise" in the curve when we switch from a simple union (Union Voting 1, green curve) to Union Voting 140 (blue curve). (The dotted blue green curves show Union Voting with thresholds in-between 1 and 140.)

6 Conclusion

In this paper we consider the Fusion of multiple anomaly detection algorithm. The motivation for this fusion process has evolved due to the widespread belief that even though none of the existing ADAs achieves perfect classification, the combination of multiple ADAs may create a superior outlier detection algorithm as has been achieved in the classification and clustering domains. In this paper we describe the expertise based fusion algorithm we developed. This algorithm may be classified as a semi-supervised method. To evaluate the performance of the proposed method with the benchmark method that exists in the literature, we limited the study to the case of a single type of outlier. For this case we showed that by using the /union voting method, we can overcome the normalization problem which is one of the critical parts in any fusion process. We do so by using ranking instead of actual scores. Thus we demonstrated that our method outperforms the benchmark method from the literature.

References

1. Chandola, V., Banerjee, A., Kumar, V.: Outlier detection: A survey. ACM Computing Surveys (2007) (to appear)
2. Petrovskiy, M.I.: Outlier detection algorithms in data mining systems. Programming and Computer Software 29(4), 228–237 (2003)
3. Zhang, L., Leung, H., Chan, K.C.C.: Information fusion based smart home control system and its application. IEEE Transactions on Consumer Electronics 54(3), 1157–1165 (2008)
4. Ahmed, M., Pottie, G.: Fusion in the context of information theory. Distributed Sensor Networks, 419–436 (2005)
5. Jeon, B., Landgrebe, D.A.: Decision fusion approach for multitemporal classification. IEEE Transactions on Geoscience and Remote Sensing 37(3), 1227–1233 (1999)
6. Schubert, E., et al.: On Evaluation of Outlier Rankings and Outlier Scores. In: SDM (2012)
7. Dietterich, T.G.: Ensemble methods in machine learning. Multiple classifier systems, pp. 1–15. Springer, Heidelberg (2000)
8. Tan, A.C., Gilbert, D.: Ensemble machine learning on gene expression data for cancer classification (2003)
9. Balke, W.-T., Kießing, W.: Optimizing multi-feature queries for image databases. In: Proc. of the Intern. Conf. on Very Large Databases (2000)
10. Asuncion, A., Newman, D.: UCI machine learning repository (2007)
11. Kriegel, H.-P., Kröger, P., Schubert, E., Zimek, A.: Interpreting and unifying outlier scores. In: Proc. SDM, pp. 13–24 (2011)
12. Lazarevic, A., Kumar, V.: Feature bagging for outlier detection. In: Proc. KDD, pp. 157–166 (2005)
13. Nguyen, H.V., Ang, H.H., Gopalkrishnan, V.: Mining outliers with ensemble of heterogeneous detectors on random subspaces. In: Kitagawa, H., Ishikawa, Y., Li, Q., Watanabe, C. (eds.) DASFAA 2010. LNCS, vol. 5981, pp. 368–383. Springer, Heidelberg (2010)

14. Berger, T.M., Durrant-Whyte, H.F.: Model distribution in decentralized multi-sensor data fusion. In: American Control Conference. IEEE (1991)
15. Chandola, V.: Anomaly detection for symbolic sequences and time series data. Diss. University of Minnesota (2009)
16. Kriegel, H.-P., et al.: Interpreting and Unifying Outlier Scores. In: SDM (2011)
17. Geusebroek, J.M., Burghouts, G.J., Smeulders, A.: The Amsterdam Library of Object Images. Int. J. Computer Vision 61(1), 103–112 (2005)
18. Grnitz, N., Kloft, M.M., Rieck, K., Brefeld, U.: Toward supervised anomaly detection. arXiv preprint arXiv:1401.6424 (2014)
19. Rajab, M.A., et al.: CAMP: Content-Agnostic Malware Protection. In: NDSS (2013)
20. Rieck, K., et al.: Automatic analysis of malware behavior using machine learning. Journal of Computer Security 19(4), 639–668 (2011)
21. Jang, J., Brumley, D., Venkataraman, S.: Bitshred: feature hashing malware for scalable triage and semantic analysis. In: Proceedings of the 18th ACM Conference on Computer and Communications Security. ACM (2011)
22. Egele, M., et al.: A survey on automated dynamic malware-analysis techniques and tools. ACM Computing Surveys (CSUR) 44(2), 6 (2012)
23. Thom, D., et al.: Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages. In: 2012 IEEE Pacific Visualization Symposium (PacificVis). IEEE (2012)
24. Breunig, M.M., Kriegel, H.-P., Ng, R.T., Sander, J.: LOF: identifying density-based local outliers. SIGMOD Rec. 29(2) (May 2000)
25. Zhang, K., Hutter, M., Jin, H.: A New Local Distance-Based Outlier Detection Approach for Scattered Real-World Data. In: Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, PAKDD 2009 (2009)