From Data to Actionable Knowledge: Applying Data Mining to the Problem of Intrusion Detection

Terrance Goan
Stottler Henke Associates Inc.
1107 NE 45th St.
Seattle, WA 98105
Phone: 206-545-1478
Fax: 206-545-7227

Presented at the 2000 International Conference on Artificial Intelligence (IC-AI'2000)

1. Abstract

Much of the research in the area of Knowledge Discovery in Databases (KDD) has focused on the development of more efficient and effective data mining algorithms, but recently issues related to the usability of these techniques in extracting exploitable knowledge from databases has drawn significant attention. In this paper we describe the progress we are making in building an interactive knowledge discovery system to help assist in developing knowledge for activity monitoring tasks such as intrusion detection. We describe why data mining in support of problems such as intrusion detection is hard and why it is so important to make optimal use of the human in the KDD process for guiding the exploration and evaluating its results. In particular, we describe how inductively generated simulations can be used to help understand the relationships amongst disparate events.

2. Keywords

Knowledge discovery; data mining; activity monitoring

3. Introduction

Massive datasets arise naturally as a result of automated monitoring and transaction archival. Military intelligence data, stock trades, bank account deposits and withdrawals, retail purchases, medical and scientific observations, and spacecraft sensor data are all examples of data streams continuously logged and stored in extremely large volumes. Unfortunately, the sheer magnitude and complexity of data being stored acts to conceal valuable information that may lie below the surface, making manual analysis infeasible. Much of the research in the area of Knowledge Discovery in Databases (KDD) has focused on the development of more efficient and effective data mining algorithms. These new techniques can yield important information about patterns hidden in databases, but in the end a user's ability to uncover interesting and useful patterns remains limited by human cognitive capacity, and that capacity remains easy to overwhelm with today's KDD tools. This fact has led to an increasing interest in improving the accessibility of KDD tools for extracting knowledge that can form the basis for improved decision making in real life situations.

The term "actionable pattern" [Adomavicius and Tuzhilin, 1997] refers to knowledge that can be uncovered in large complex databases and can act as the impetus for some action. It is important to distinguish these actionable patterns from the lower value patterns that can be found in great quantities and with relative ease through so called data dredging. This highlights the need to make optimal use of the human in the KDD process for directing the exploration and evaluating its results — one cannot simply apply some predefined procedure to distill large volumes of data into high value knowledge. Rather, KDD tools must provide users with the insight required to focus the tool's search processes and with the means to efficiently evaluate discovered patterns.

In the remainder of this paper we describe the progress we have made in enabling users to extract actionable patterns/knowledge through iterative exploration. As an example, we describe how our IKODA (Intelligent KnOwledge Discovery Assistant) system can be utilized to facilitate the intuitive exploration of complex temporal activity data. In particular we describe how we can employ data mining algorithms to generate simulations of computer user behavior and use them to construct effective knowledge bases for intrusion detection.

3.1 Activity Monitoring

"Activity monitoring" [Fawcett and Provost, 1999] is a particularly challenging class of knowledge discovery tasks, and it represents a good vehicle for exploring the requirements for the practical extraction of actionable knowledge. Activity monitoring refers to the general task of analyzing the behavior of entities over time for interesting events that require action. For instance, one type of activity monitoring called "machine condition monitoring" involves watching equipment sensors for signs of impending failure. A significantly different activity monitoring domain is intrusion detection, where software systems attempt to correlate evidence of a user's actions in order to uncover malicious activity. For each of these problems there are a wide range software capabilities that can improve real-time monitoring capabilities including: extracting informative features from sensor data; discovering early indicators of future alarms (e.g., high concentration of copper in oil may be a precursor to engine failure); and accurately modeling the underlying behaviors.

Another aspect of activity monitoring problems is the need for timely analysis. In the case of equipment condition monitoring, it is important that the software be able to predict problems early enough to allow the users to intervene and prevent a catastrophic failure. Similarly, in intrusion detection, it is important to detect an intrusion as quickly as possible in order to limit damage or restrict access to sensitive data. Finally, in all activity monitoring tasks, false alarms can cause potentially substantial personnel inefficiency. Having to balance the need for quick detection of trouble against the cost of false alarms makes the development of activity monitoring systems a substantial challenge.

3.2 Intrusion Detection

In defending network resources any number of technologies might be applied. Firewalls, encryption technology, authentication devices, vulnerability checking tools, and other products can all offer improved security. But, even when a computer system is equipped with stringent authentication procedures and firewalls, it is still susceptible to hackers that take advantage of system flaws and social engineering tricks. Computer systems with no connection to public networks remain vulnerable to disgruntled employees or other insiders who misuse their privileges. Given these enduring threats, it is only sensible to establish a second line of defense in the form of an intrusion detection system (IDS).

The field of intrusion detection began close to 20 years ago. The majority of early systems were designed to detect attacks upon a single host. More recent systems consider the role of networks and look for evidence of intrusions by passively monitoring the traffic on the local area network. Yet another set of IDSs are designed to collect and aggregate evidence from multiple sources in order to detect coordinated or multi-stage attacks on a network (e.g., [Snapp, Goan, et al., 1997]). Such evidence fusion is an important capability since skilled intruders often employ a number of techniques to hide their identity or disguise their attacks.

Effective intrusion detection capability remains elusive as computing environments become more complex and crackers continually adapt their techniques to overcome innovations in computer security. Additionally, the excessively high false alarm rates associated with existing tools must be addressed before these tools can fulfill their potential.

3.3 Developing Actionable Knowledge through Data Mining

Typically the knowledge bases underlying IDSs are developed through knowledge engineering and are based on limited case studies. The resulting ad hoc decision models can result in poor rates of intrusion identification and high false alarm rates. Recognizing this, Wenke Lee [Lee, Stolfo, and Mok, 1999] developed a data mining approach to building IDSs that promises to improve the accuracy and computational efficiency of IDSs. Lee's work employs dependency analysis to systematically identify evidence that can be efficiently assessed and can be used to accurately distinguish intrusions from normal activity. The resulting intrusion detection models can be used in real-time to detect certain forms of suspicious behavior in network traffic with high accuracy.

While we share Lee's goal of developing data mining based automation that can be used to develop sound intrusion detection models, we take a substantially different approach. Lee's work focuses on developing a minimum set of quickly computable *perfect* indicators of an intrusion (i.e., patterns of activity that occur only in intrusions). As they describe in their paper this approach excels in detecting certain forms of intrusion such as denial of service attacks, where the attacker floods a machine with particular data packets. In order to tackle other classes of attack where the intrusion takes a less well defined course and where skilled attackers can obscure their actions by destroying evidence, we must identify a wider selection of individually imperfect but corroborating evidence sources (see [Goan, 1999]) because any one piece of evidence may be destroyed or corrupted.

This expanded goal of extracting *and fusing* the knowledge required for accurate and robust intrusion detection in the face of skilled intruders is a significant challenge. In our work we employ a dependency analysis based approach similar to that described by Lee but we cast a wider net, and rely on a rich interaction with the human user to separate the wheat from the chaff.

3.4 Intrusion Simulation

In order to craft an intrusion detection knowledge base that incorporates diverse evidence it is necessary to understand the dynamics of an intrusion and in particular how it is expressed through the available evidence sources. In particular, developers of IDSs need to uncover the relationships amongst partially redundant evidence sources as well as counter evidence (e.g., evidence that suggests that the particular attack is not occurring). These relationships necessarily include the affects of evidence on the certainty/probability that an attack is occurring, temporal constraints (e.g., the order in which evidence appears), as well as other constraints that allow the IDS to determine what evidence to associate with what intrusion hypotheses. Understanding these relationships sufficiently to codify intrusion detection knowledge bases that effectively manage uncertainty is incredibly challenging.

Within IKODA we have incorporated a technique similar to Goodman's Projective Simulation technique which applies data mining to developing activity/process simulators directly from historical sensor data [Goodman, 1994]. The first step in developing a Projective Simulator is to identify the *k* "sensors" that generate the data that characterize the behavior of interest. In applying this technique to intrusion detection, one can use domain knowledge, intuition, and feature construction techniques like those proposed by Lee [Lee, Stolfo, and Mok, 1999]. Second, we build *k* classifiers (e.g., decision trees), one for each of the sensors of interest. In building the classifier for a particular sensor *m* the induction algorithm utilizes a sliding window (time_x to time_y) of historic data for all *k* sensors to predict the value of *m* at time time_{y+1}.

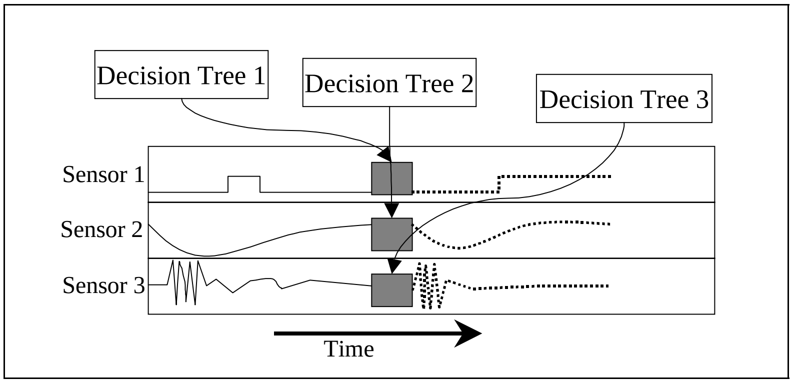

Figure 1. Using a Projective Simulation model to predict future values of three sensors.

Once the classifiers are created, we can then predict the value of sensor *m* at time *t*+1 (where *t* is the current time) by "classifying" the current situation using the values of the *k* sensors over the time window time$_{t-(y-x)}$ to time$_t$. Using the new projected values we can slide the window forward one step and again utilize the classifiers to predict the next step, time *t*+2. This process can continue (with compounding error) infinitum (see Figure 1).

While it is certainly possible that one might use Projective Simulation directly for intrusion detection, we feel its true value is in allowing the user to gain insight into the time dependent unfolding of evidence of intrusive behavior. Because the classifiers that underlie the Projective Simulator effectively generalize across historic episodes, it can allow users to see how attacks can unfold through a variety of what-if analyses. Through this exploratory process, users will then be able to construct well-founded intrusion detection knowledge bases.

While we are just beginning to explore the full breadth of potential uses for Projective Simulation, it is now clear that this approach can be very useful in a variety of ways to intrusion detection and network manage. In particular, it has proved itself to be useful in modeling the dynamics of a computer network and diagnosing network problems (e.g., subverted routers, denial of service attacks, network link failure, host failure).

Early results also suggest that Projective Simulation can be useful in:

- Understanding variances in attack signatures and timing constraints.
- Determining the usefulness of partially redundant evidence sources and the time profile of evidence expression.
- Examining the affect of attacks on system performance metrics (e.g., CPU load or network connection failures).
- Uncovering indicators that can distinguish merely anomalous behavior from truly intrusive behavior.

These benefits are gained not simply from the fact that we have a simulation, but rather by the fact that the knowledge mined to form the simulation is itself explorable. This is in contrast to opaque approaches such as neural networks that can be applied to form such simulations. In the next section we describe how Projective Simulation fits into IKODA's knowledge discovery environment.

3.5 User Centered Exploration

We have developed our IKODA knowledge discovery system with the recognition that while computers can be exceptionally valuable in finding patterns in very large databases, it is the challenge of a KDD system to enable human users to efficiently apply their own very significant powers of intuition, pattern recognition, and judgement. To do this IKODA utilizes "data aware" visualization techniques that provide a mapping between graphical objects and the underlying data resources.

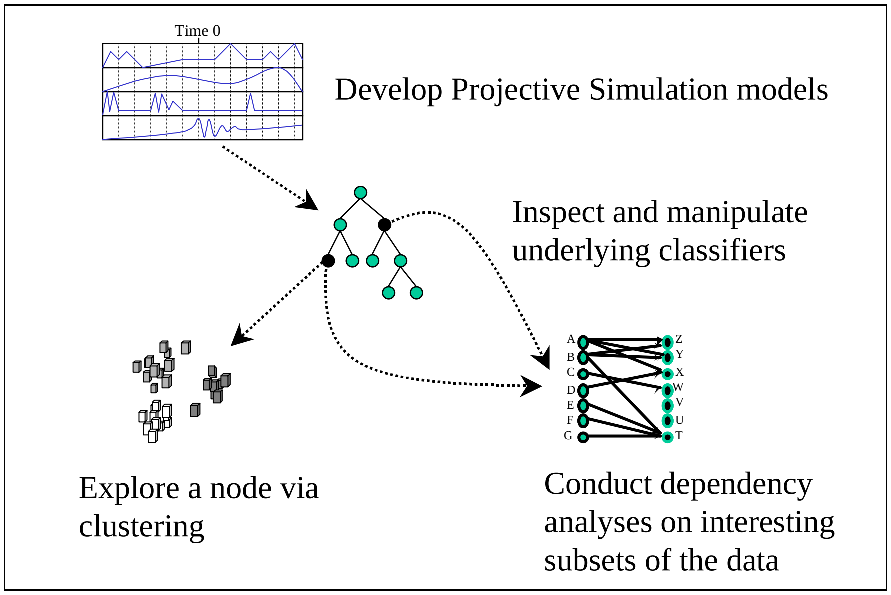
Time 0
Develop Projective Simulation models
Inspect and manipulate underlying classifiers
A
B
C
D
E
F
G
Z
Y
X
W
V
U
T
Explore a node via clustering
Conduct dependency analyses on interesting subsets of the data

Figure 2. Recursive data mining with IKODA.

This approach results in the ability to perform direct manipulation operations such as drag-and-drop transfer of data between tools and a unique capability to drill-down into data mining results. Unlike previous integrated KDD systems, our IKODA's visualizations act as interactive tools rather than simple information displays. More specifically, IKODA's visualizations of data mining results (e.g., decision trees and automatically created data clusters), can be manipulated and used directly to form new datasets that feed future data mining operations. The resulting "recursive" knowledge discovery capability represents a substantial step forward in reducing KDD tool complexity while simultaneously increasing flexibility and efficacy.

Figure 2 shows how Projective Simulation can be used in exploratory data mining. After generating a Projective Simulation model, a user might engage in various what-if analyses to gain insight into the dynamics of the behavior being modeled. The user might then seek additional information by examining the classifiers, in this case decision trees, that are used as a foundation for the simulation. The user can then drill down further into the discovered patterns by utilizing the data underlying particular components of these classifiers as input to other data characterization actions. IKODA’s pervasive direct manipulation interface greatly facilitates each of these steps.

4. Future Work

While we have made good progress in building tools to help in extracting useful intrusion detection knowledge from large databases of data, there remains much to do. In particular we need to extend our notion of Projective Simulation to allow for “time warping” (i.e., varying temporal constraints amongst events). This will be critical for building simulators for the vast majority of computer intrusions. We are also working to extend the ability of users to apply their own domain knowledge to target their data mining efforts in ways that are analogous to those proposed by Lee [5].

5. Conclusions

In this paper we have described how a Projective Simulation capability together with an integrated KDD environment can greatly facilitate a human user’s ability to extract actionable knowledge from large data resources. By provided a new level of insight into complex temporal data, IKODA can act as the basis for systematic problem analysis and facilitate the creation of high quality knowledge bases. We focused on the application of this approach to intrusion detection, but the underlying techniques can be very beneficial to any number of activity monitoring tasks.

6. References

[1] Adomavicius and Tuzhilin. Discovery of Actionable Patterns in Databases: The Action Hierarchy Approach. KDD, 1997.

[2] Goan. A Cop on the Beat: The Collection and Appraisal of Intrusion Evidence. Communications of the ACM, July 1999.

[3] Fawcett and Provost. Activity Monitoring: Noticing interesting changes in behavior. KDD, 1999.

[4] Goodman. Results on Controlling Action with Projective Visualization. In Proceedings of AAAI, 1994.

[5] Lee, Stolfo, and Mok. Mining in a Data-flow Environment: Experience in Network Intrusion Detection. KDD, 1999.

[6] Snapp, Goan, et al. DIDS (Distributed Intrusion Detection System)- Motivation, Architecture, and an Early Prototype. In *Internet Besieged: Countering Cyberspace Scofflaws*. Eds. Peter J. Denning and Dorothy Elizabeth Denning. Addison-Wesley, 1997.