CASE STUDY OF AN ADAPTIVE AUTOMATED HEALTH INSURANCE FRAUD AUDITOR

Fletcher Lu, University of Ontario Institute of Technology

Abstract

One of the significant potentials of encoding health records into an electronic format is as a vast resource that may be mined to find hidden relationships such as the task performed by auditors in their search for fraud. However, the simplicity of the idea belies the difficulty of this task. The key software engineering challenge involves extracting information from different sources despite being presented in different formats. An additional challenge is to merge this information with multiple fraud detection methods to take advantage of all the current detection techniques available. In this paper, a case study of a prototype software implementation of an automated fraud auditor is presented which was designed to replicate the investigative operation of human fraud auditors. The focus was on the adaptive design of the system. The implementation of this design on a set of real health insurance and hospital records, as well as a performance test on real audited data, demonstrated its improved efficiency over human auditor fraud case building.

Introduction

In 1999, Stead and Lorenzi [1] illustrated the need to link investment in Health Informatics to derived value from that investment in terms of improved public health, improved quality as perceived by consumers, and lowered costs. The automated fraud auditor presented here addressed the issue of lowering the costs of providing health care by reducing instances of health insurance fraud. A cost that Simborg [2] estimated to be between 3% and 10% of the total health care costs, which is a huge range illustrating the great amount of uncertainty in the amount of fraud that may go undetected.

Building a software application that will replicate the operation of a human fraud auditor, however, presented many challenges. The key challenge involved the core definition of fraud. The National Health Care Anti-Fraud Association (NHCAA) defines it as an intentional deception or misrepresentation made by a person or an entity with the knowledge that the deception could result in unauthorized benefits to that person or entity [3]. Thus, fraud requires two components:

1. deception, and
2. an unjustified gain or loss.

What these two components imply is that whatever fraud is perpetrated, it is generally hidden to some degree and some party must obtain an unjustified benefit or loss. Automated fraud-detection systems have been developed that use pre-defined sets of rules built by domain experts to uncover specific types of fraud. These pre-defined requirements limit their broad-based usability. More flexible methods capable of uncovering new patterns of fraud involve statistical machine-learning techniques that may be classified as either supervised or unsupervised. Supervised methods train on data that has been labeled by domain experts for fraud and then searches on general data for information that satisfies the fraud patterns learned from the training data. Unsupervised methods, in contrast, search for outliers/anomalies in the data. They potentially have a greater chance of uncovering new types of fraud as they are not restricted in any way to preset rules or trained data.

The software application developed in this study differs markedly from these previous fraud-detection tools in the fact that it may use these fraud detection tools to uncover potential cases of fraud. But it goes one step further by building a case of fraud around those potential cases, just as an auditor might do, by searching for related records to ascribe the unjustified gain or loss to some party. This new system, then, may best be described as a fraud-detection case builder, unlike any commercial system currently available in its case-building objective.

Just as human auditors may use any and all possible fraud/anomaly detectors, the automated system developed here may do so as well. The utility of this mechanism is in

- automating the task of human auditors who have to run multiple detectors and then manually explore the results to build the fraud case;
- speeding up the searching of the possibly vast amounts of data related to an outlier that must be sifted through in order to link records together that build a case for fraud; and
- relating data across different database tables with different data representation formats for the same information.

These challenges may be compounded by the large number of potential fraud cases that may need to be investigated depending on the magnitude of the original search space that the outliers were drawn from. In addition, healthcare data-

bases are continuously growing with new data continuously presenting new potential cases of fraud.

The system developed in this study uses a form of semi-supervised learning known as reinforcement learning [4] to allow it to have the flexibility to take information from multiple detectors and search data from multiple databases that could be in numerous data formats. The authors provide a background look at reinforcement learning, the various current fraud-detection techniques, a detailed explanation of the system's design, and a case study of the proposed system's operation on data from a real insurance provider.

It should be emphasized that this system is a fraud case builder paralleling the tasks of human fraud auditors. This means that the system can take information from other fraud detection programs to build its case for fraud. Thus, it should operate at least on par with any of the commercial fraud detectors as long as that fraud detector's results are included in the system's information when it creates its cases. The advantage of this fraud case builder is that it should be able to work faster than human auditors building these cases. Attempting to replicate the human auditor's task is what is unique to this system in contrast to fraud detection systems that currently exist which function primarily as tools for the human auditor to identify records and situations that the auditor must still investigate.

As a comparison, the authors needed to compare their performance not to existing fraud detectors, since the proposed system would actually use those fraud detector's results, but rather to the performance of human auditors in identifying actual cases of fraud. The proposed system should either be able to find all of the cases that the human auditors find, but faster, or cases that were not known to the human auditors.

Three experiments were conducted:

1. A preliminary experiment demonstrating the utility of combining information from more than one fraud detector to produce more correctly identified cases of fraud over the fraud cases of two separate detectors each working alone.
2. An experiment to measure the effects of combining information from two different fraud detectors in varying ratios.
3. A performance comparison on processing times of the proposed system against actual human fraud-auditor-created rule sets on real health insurance data.

Background

As noted by Li et al.[3], health-care-fraud behaviors may be classified under the potential involved parties:

i. Service provider fraud
ii. Insurance subscriber fraud
iii. Insurance carrier fraud

Each of these parties may commit fraudulent actions such as

i. Service-provider fraud: Billing for services not actually performed
ii. Insurance-subscriber fraud: Falsifying claims to medical services never received
iii. Insurance-carrier fraud: Falsifying benefits statements

In the case of a false billing of a service never performed, this could be perpetrated by either the provider or subscriber and illustrates the importance of relating a billing anomaly with is the person ultimately benefiting from the false bill in order to correctly identify the guilty and avoid false accusations.

Uncovering such relations may require information from multiple resources such as the provider's billing records compared against the insurance subscriber's health records. Such records can be, and often are, in different formats, requiring a system flexible in searching and matching across these multiple formats. It is this required flexibility in representation combined with a capability to search and build patterns of relations that the authors chose a semi-supervised learning technique known as reinforcement learning.

Reinforcement Learning

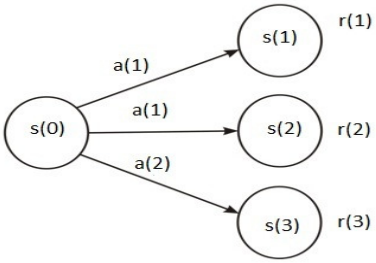

Figure 1. Representation of a Reinforcement Learning Network

Here, the authors provide a very brief overview of the reinforcement learning technique to machine learning. For a more detailed explanation, see Sutton and Barto [4]. In reinforcement learning, an environment is modeled as a network of states, S. Each state, s∈S, is associated with a set of possible actions, a(s) ∈A and a reward for entering that state r(s) ∈R, where A and R are sets of rewards and actions, respectively. It is possible to transition from one state, s(i), to another, s(j), by choosing an action, a(s(i)), with a certain probability, Prob(s(j) |s(i),a(s(i))). The advantage of this representation is that it places very little restriction on formatting. The objective of this representation is to find an optimal policy. A policy is a function that maps states to

actions. In other words, it makes choices on actions to take for any given state visited. An optimal policy maximizes the long-term rewards one may obtain as one navigates through the network [4-6]. For the proposed fraud-auditing system, the optimal policy is one which will maximize the likelihood that a set of related data records form a case of fraud. This relation is made by allowing rewards, R, in the reinforcement environment to represent a numerical value derived from fraud detectors, where the larger the reward value, the more likely attributes in the state are an indicator of fraud. The appropriateness to use reinforcement learning, a method traditionally used in robotic search, to search across database tables was presented by Lu [7].

Fraud Detectors

As stated in the introduction, a variety of fraud detection methods exist, with the adaptive ones being divided into supervised and unsupervised methods [8], [9]. For supervised methods, both fraudulent and non-fraudulent records are used to train a system, which then searches and classifies new records according to the trained patterns. The limitation of supervised methods is that one must have both classes of records identified for the system to train on. Thus, this approach is limited to only previously known methods of committing fraud. A few of the more popular supervised approaches involve Bayesian networks [10] and classifiers to detect suspicious claims [11], Neural networks [12], [13] and Decision trees [14], [15].

Unsupervised methods, in contrast, often identify records that do not fit expected norms or essentially looking for outliers [16]. The advantage of this approach is that one may identify new instances of fraud. A common approach to this method is to use forms of outlier detection. The main limit to this approach is that we are essentially identifying anomalies that may or may not be fraudulent behavior. An audit investigator in this case may then be employed to analyze these anomalies for their likelihood to be indicative of fraud. One expert system that has been developed using unsupervised techniques is known as SmartSifter [17], which uses probabilistic models to generate its outliers.

Application Method

Reinforcement learning is well-suited to linking together states through its state-action policy mapping. For reinforcement learning to be used as a fraud case builder, it needs to be able to relate rewards with outliers that are indicative of possible fraud. It does so by preprocessing all records using sets of fraud detectors such as outlier/anomaly methods, so that all records have a numerical reward value that represents their likelihood of fraud. The states of the proposed RL environment relate to individual records of the application environment, and the actions are the attributes of a record. In this way, two records with the same attributes are linked together by a common attribute just as an action can relate two states of a classic reinforcement learning environment network. A second phase of the preprocessing is to search record columns for matching attributes across different tables so that the columns may be linked during the reinforcement learning search. For example, two databases may both include a column for patient names. These should be linked in order to match records across databases.

After the preprocessing, an exploration search on the RL environment produces an optimal policy indicative of fraud. This policy will then be followed to uncover a list of records that are related with a high chance of fraud. Following is a summary of the proposed approach.

1. First, preprocess all database records with a selection of fraud detectors to assign reward values.
2. A second preprocess phase links database columns through their attributes matching different database tables using a pattern match search on the values within the columns.
3. Run a reinforcement learning approach using attributes of a record as action choices in a reinforcement learning context to search the databases until an optimal policy is found.
4. Navigate through the environment using the derived optimal policy with a start state drawn from one of the significant outliers from the statistical distributions produced from step 1.
5. Return all records encountered.

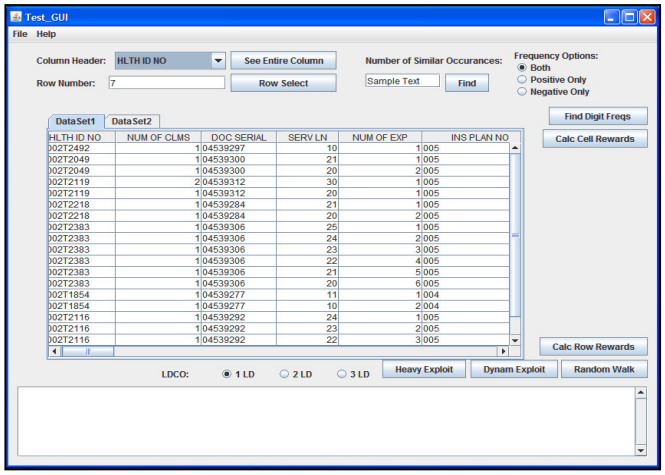

Figure 2. Screen capture of the application with sample health insurance data set

The returned records are all in ranked order from most likely fraudulent to least likely, based on their reward value. For further details on these algorithmic techniques and principles behind this application see Lu [7].

Figure 2 illustrates the interface window of the authors' software application. There are a number of features that have been incorporated for usability and functionality. To allow for searching multiple database tables, the authors used multiple tabs; each tab holds a table of a database. Dataset 1 in Figure 2 illustrates a sample health insurance record, while data set 2 shown in Figure 3 illustrates a table of hospital in-take records. The blank window below the datasets window is where the list of related records that are potentially fraudulent are presented, with each potential case of related records listed per row. The system has capabilities to search for specific terms within columns of tables in order to support a user's analysis of the results. This initial prototype includes functionality for:

1. Two forms of fraud outlier detectors;
2. Search capability to set thresholds for different random walks;
3. Options on using an online update method with the 'Dynamic Exploit' button, which continuously updates the policy with new updated records or an offline solution with the 'Heavy Exploit' button.

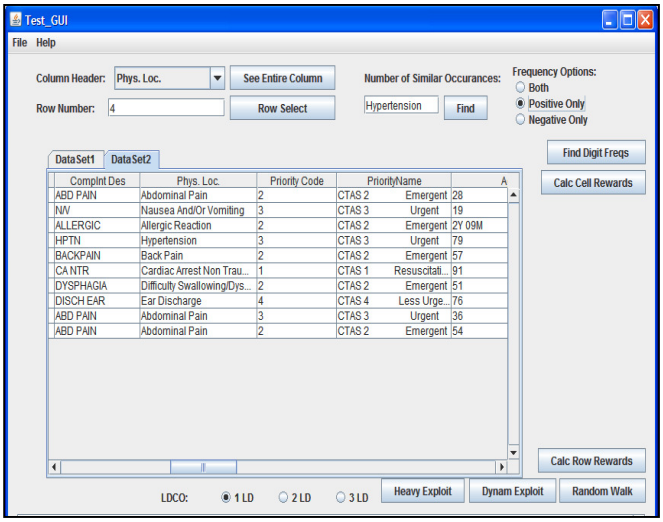

Figure 3. Screen capture of application with sample hospital intake records displayed

Experiments

Experiment 1

As a first test on the proposed automated auditing approach, the authors ran a system test using real insurance data consisting of 31,804 records that had been audited and labeled for fraud cases. The data was divided into a set of ten test sets with a random selection method to choose the records that would be included in each test case. An outlier detection approach, where records that deviated from a set threshold were included in the fraud category, were compared against the proposed system which used the same threshold but with aggregated information over several related records. A single fraud detector is used in any given record, but different records could use different fraud detectors. In the current experiment, the authors used two fraud outlier detectors: a Normal Distribution outlier and a method known as Benford's law distribution outlier.

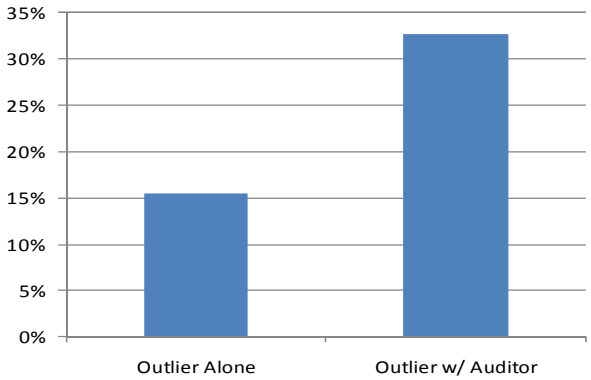

Figure 4. Improvement with automated auditor

Figure 4 illustrates the averaged results of this experiment comparing the percentage of correctly identified fraudulent cases relative to the number of cases each method recommended as potentially fraudulent. Generally, the automated auditor had approximately double the precision of using the fraud detector alone.

Experiment 2

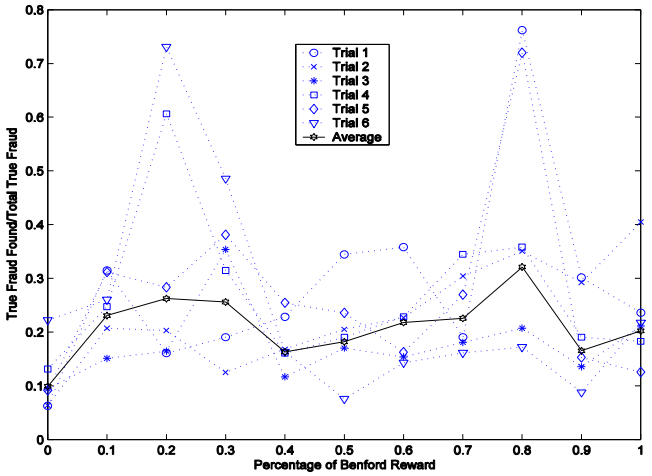

Figure 5. Combining fraud detection data with Benford vs. Normal Distribution Rewards

As a second test, the authors wanted to consider the scenario where more than one fraud detector may contribute useful fraud information to a record. Under such a situation, how should one combine the multiple pieces of evidence into a single reward value? In this experiment, the authors

combined normal distribution outlier rewards with the Benford's Law outlier reward. The reward formula used was

$$\text{Reward}(i) = RB(i) \times \%B + RN(i) \times (1 - \%B), \quad (1)$$

where RB(i) is the Benford reward calculated for state i, RN(i) is the Normal distribution reward calculated for state i and %B is the percentage Benford's reward contributes to the overall reward value ($0 \leq \%B \leq 1$).

Using real industry data that had been audited for fraud, a reinforcement learning algorithm was run over six trial runs, where each trial was comprised of fifteen sampling trajectories of 1000 sampling steps and where rewards were varied in increments of 10%. Normal distribution rewards appear on the far left side of Figure 5 and decrease from there with increasing amounts of Benford rewards until 100% Benford's Law distribution rewards on the far right side of the figure are reached. The data used includes 227,156 records with 1,526 previously identified fraudulent records, which were used to verify the accuracy of the fraud results.

Analyzing the average results from the results of Figure 5, it would also appear that the two reward methods interact in a complicated manner. The fraud-detection accuracy increases to a peak at 20% Benford/80%Normal, and then peaks again at approximately 80% Benford/20% Normal, giving a possible bimodal interpretation. This indicates that combining the two reward mechanisms has some benefit yielding generally greater true-fraud cases compared to the detection accuracies of each detector alone shown on each far end of the graph. This result supports the idea that combining information from multiple fraud sources will generally yield improved results.

Experiment 3

In the third experiment, an insurance business dealing in coverage for individuals for both dental and pharmaceutical drug claims wished to assess their auditors' performances against the proposed systems to determine how much efficiency improvement, if any, there may be using this automated fraud auditor. The experiment looked at comparing the performance time of a human-auditor-created-fraud rule using customized commercial software tools against the proposed automated fraud auditor. The rule sets were designed to extract records satisfying conditions that they had identified as indicative of potential fraud. The rules also included a rank ordering of the records from most-likely-fraudulent to least. The company used an oracle database with customized query capability with commercial statistical software for the fraud auditors to extract their records. A base requirement was that the ranking should be the same on the same set of data. If that condition was satisfied then the performance times of the human-auditor-created rule set query and ranking time could be compared with the authors' system's time.

The data consisted of three months of drug claims records comprised of 1,890,548 actual records from January 1, 2010 to March 31, 2010. The authors were asked to conduct two searches: the first was a 'Field test' of the proposed system. The second was an 'Unknown results test' that was actually a search that the business' own fraud system failed to complete. During a test, their system simply crashed during the query processing; thus, they were unaware of what actual results they should get. The authors' system was able to produce the required rank ordering for the first test and completed the second search with a list of rankings. The runtimes are presented in Table 1. The table compares the actual search and ranking times of the proposed system against the time the insurance business' rule set queries took to conduct the same search.

Table 1. Runtime Comparisons of Fraud Claims with ranking order

	Automated Fraud Auditor	**Business' Fraud Auditor's system using rule sets**
Field test	18 min 22 sec.	Approximately 7 hours
Unknown Results test	4min. 18sec.	Failed to Complete

The results of this experiment demonstrated that not only did the system produce results satisfying the required criteria, but that the runtimes were impressively better. This supports the premise for the creation of this system to support human auditors by replicating their task. But by automating portions of the task, the human auditors can be allowed to conduct larger and more diverse searches with the saved time.

Conclusions

In this paper, the authors presented a case study of software implementation for an automated fraud auditor designed to support human auditors in their search for fraudulent acts. The main advantages of this approach are its flexibility in combining information from multiple database records and using the information from multiple fraud detectors to build its cases. A detailed approach was used to allow for this flexibility by using a semi-supervised learning approach with few restrictions on format or structure. Preliminary tests on the prototype were conducted incorporating two outlier fraud-detection methods and demonstrated its improved precision for finding fraudulent records over the outlier fraud detectors alone. Different approaches were tested for combining fraud detectors. Finally, runtime improvements were demonstrated over human-auditor-created fraud

searches on real health insurance data producing comparable results but in a significantly shorter time frame.

In terms of future work, to test the approach's efficacy, the authors plan to incorporate more sophisticated fraud detectors to build fraud rewards such as a Bayesian [11] and a neural network detector [18]. The ultimate objective is to measurably recover losses due to uncovered cases of actual fraud. However, issues of privacy [2], [19], maintaining corporate and institutional confidence [1], and the speed of actual prosecution of potential fraud cases [20] presents challenges for both the measuring of actual cost savings and publicly presenting those savings.

Acknowledgements

This work was supported by The Ontario Partnership for Innovation and Commercialization and the Natural Sciences and Engineering Research Council of Canada.

References

[1] Stead WW, Lorenzi NM. Pryor. Health informatics: linking investment to value. JAMIA. 2010;6:341-348.
[2] Simborg DW. Healthcare fraud:whose problem is it anyway? JAMIA. 2008;15:278-280.
[3] Li J, Kuei-Ying Huang K-Y, Jin J, Shi J. A survey on statistical methods of health care fraud detection. Health Care Management Sci. 2008;11:275-287.
[4] Sutton RS, Barto AG. Reinforcement learning: an introduction. MIT Press. 1998.
[5] Sutton RS. Learning to predict by the method of temporal differences. Machine Learning. 1988;3:9-44.
[6] Singh SP, Sutton RS. Reinforcement learning with replacing traces. Machine Learning. 1996;22:123-158.
[7] Lu F. Uncovering fraud in direct marketing data with a fraud auditing case builder. PKDD. 2007;11:540-547.
[8] Fawcett T. AI Approaches to fraud detection and risk management. AAAI Workshop: Technical Report. 1997;WS-97-07.
[9] Bolton RJ, Hand DJ. Statistical fraud detection: a review. 1999;17(3):235-255.
[10] Ormerod T, Morley N, Ball L, Langley C, Spenser C. Using ethnography to design a mass detection tool (MDT) for the early discovery of insurance fraud. Proceedings of ACM CHI. 2003.
[11] Chan CL, Lan CH. A data mining technique combining fuzzy sets theory and Bayesian classifier-an application of auditing the health insurance fee. Proceedings of the Inter. Conf. on A.I. 2001;402-408.
[12] Hall C. Intelligent data mining at IBM: new products and applications. Intelligent Software Strategies. 1996;7(5):1-11.
[13] Ortega PA, Figueroa CJ, Ruz GA. A medical claim fraud/abuse detection system based on data mining: a case study in Chile. Proceedings of Int. Conf. on Data Mining. 2006.
[14] Bonchi F, Giannotti F, Mainetto G, Pedreshi D. A classification-based methodology for planning auditing strategies in fraud detection. Proceedings of SIGKDD99.1999:175-184.
[15] Williams G, Huang Z. Mining the knowledge mine: The Hot Spots methodology for mining large real world databases. Lecture Notes in Computer Science. 1997;1342:340-348.
[16] Langley P. Elements of machine learning. Morgan Kaufmann, San Francisco. 1995.
[17] Yamanishi K, Takeuchi J, Williams G, Milne P. Online unsupervised outlier detection using finite mixtures with discounting learning algorithms. Data Mining & Knowledge Discovery. 2004;8:275-300.
[18] He H, Wang J, Graco W, Hawkins S. Application of neural networks to detection of medical fraud. Expert Syst Appl. 1997;13:329-336.
[19] Buckovich SA, Helga ER, Rozen MJ. Driving toward guiding principles: a goal for privacy, confidentiality, and security of health information. JAMIA. 1999;6:122-133.
[20] National Health Care Anti-Fraud Association website. Available at: http://www.nhcaa.org. Accessed March 2010.

Biography

FLETCHER LU is an Assistant Professor in the Faculty of Health Sciences at the University of Ontario Institute of Technology. He is peer-review published in areas of health informatics, machine learning and artificial intelligence. He received a Bachelors of Mathematics with distinction as well as a PhD. in Computer Science from the University of Waterloo. Professor Lu may be reached at fletcher.lu@uoit.ca