Role Prediction using Electronic Medical Record System Audits

Wen Zhang[1], Carl A. Gunter[2], David Liebovitz[3], Jian Tian[1], and Bradley Malin[1,4]
[1]Dept. of Electrical Engineering & Computer Science, Vanderbilt University, Nashville, TN; [2]Dept. of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL; [3]Dept. of Medicine, Northwestern University, Chicago, IL; [4]Dept. of Biomedical Informatics, Vanderbilt University, Nashville, TN

Abstract

Electronic Medical Records (EMRs) provide convenient access to patient data for parties who should have it, but, unless managed properly, may also provide it to those who should not. Distinguishing the two is a core security challenge for EMRs. Strategies proposed to address these problems include Role Based Access Control (RBAC), which assigns collections of privileges called roles to users, and Experience Based Access Management (EBAM), which analyzes audit logs to determine access rights. In this paper, we integrate RBAC and EBAM through an algorithm, called Roll-Up, to manage roles effectively. In doing so, we introduce the concept of "role prediction" to identify roles from audit data. We apply the algorithm to three months of logs from Northwestern Memorial Hospital's Cerner system with approximately 8000 users and 140 roles. We demonstrate that existing roles can be predicted with 50% accuracy and intelligent grouping of roles through Roll-Up can facilitate 65% accuracy.

Introduction

There are two dominant strategies for limiting access to Electronic Medical Records (EMRs) within enterprises such as hospitals. One strategy, known as Role Based Access Control (RBAC)[1,2], groups access privileges into collections called *roles* and then assigns user to roles to determine their access privileges. This is commonly done by looking at the job positions in the enterprise and the tasks the employees in these positions need to perform, then assigning privileges to positions, or variants of them, to enable the employees to accomplish their assigned tasks.[3] A second strategy, which we group under the general heading of Experience Based Access Management (EBAM), emphasizes accountability and the use of audit data to punish abuse. An often referenced strategy for EBAM is to manually review the audit logs of VIPs to determine when abuses transpire.[4,5,6] Another strategy, often called "break-the-glass" security, discourages abuse by warning users that certain types of access are likely to be manually reviewed.[7]

However, at the current point in time, RBAC and EBAM are used without much common foundation. This is a pity since there seems to be significant opportunities for synergy between the techniques. Consider, audit data may provide valuable information about roles, such as whether a new role would be beneficial or whether two existing roles should be merged. More appropriate role definitions, or roles that are context-specific, driven by auditing analytics, may be applied to restrict access so that fewer checks are required in the auditing process.

The aim of this paper is to investigate a key step that could lead to such a synergy between RBAC and EBAM. We call the concept *role prediction* and it refers to the ability to use audit logs to predict whether a given user is associated with a given role. Role prediction can be a valuable tool for the role engineer, that is, the security administrator responsible for creating roles and managing assignments to them. For instance, a pair of roles that are often confused in the role prediction process might be good candidates for merging. Moreover, role prediction can provide insights into role hierarchies, such as indicating whether the right relationships have been allocated.

This paper has three specific goals:

- **Hospital Role Classification:** First, we aim to determine the extent to which expert-defined job titles in a large academic medical center help to distinguish between roles. To perform this part of the investigation, we train a machine learning-based classifier over the various features invoked by users acting in a role while accessing a patient record, and classify a test set of users. The accuracy acquired is used to measure the quality of the role specifications.
- **Intelligent Role Abstraction:** Second, we hypothesize that certain abstractions of roles can permit more accurate differentiation of roles in the system. To answer this hypothesis we developed and applied role hierarchies to determine appropriate levels of role auditing. Moreover, we develop a heuristic-based algorithm, called *Role-Up*, to execute a "rolling-up" procedure for the hierarchy.

- **Empirical Evaluation** Third, we apply our methods to three months of access logs from a large academic medical center, Northwestern Memorial Hospital. From these results we judge whether the role specification performs well and how role specification might be optimally informed.

Our findings suggest that RBAC for EMR systems can be effectively guided through information mined from audit logs. We demonstrate generalization of roles can improve the predictability of role behavior with minimal sacrifices to the specificity of the system.

The remainder of this paper is structured as follows. In the following section we review access control and auditing principles, as well as the limits of RBAC in healthcare environments. Next, we introduce the environment and EMR access logs used, as well as the *Role-Up* algorithm designed for this study. We then report on an extensive experimental analysis of role prediction and the recommendations made by the algorithm. Finally, we conclude with a review of the contributions, relationship of this work to recent federal recommendations for health information technology, and next steps.

Background

As EMR systems expand in size, scope, and distribution, it is critical, and a federal requirement8, to ensure the privacy of those whose information is stored. Uncontrolled access to health information could lead to privacy compromise, breaches of trust, and eventually harm.9[,10] Thus, it is worthwhile to consider the extent to which established approaches for access control could be applied in the healthcare domain.

The notion of access control was first formalized by the *Access matrix model* (AMM), a conceptual framework that specifies each user's permissions for each object in the system.[11] While formal, the AMM is limited in scalability and a variety of methods for more efficient and effective access management have been proposed. The most popular model is *Role-based access control* (RBAC), where permissions for access to objects are assigned to roles instead of to individual users as in the AMM.1 Users are associated with one or more roles and thus acquire corresponding access permissions. RBAC is now widely deployed in a wide range of domains because the deconstruction provided simplifies access control management in that organizational roles may be mapped to a set of functions.[12]

It is also worth noting that RBAC is context independent and a growing list of extensions has been proposed to address complex environments. For instance, *Task-based Access Control* (TBAC) extends traditional user-object relationships by including task-based contextual information.[13] With respect to healthcare, this concept forms the basis of *Situation-based Access Control*, where access rights depend on where in a guideline, or workflow, the patient is situated.[14,15] Also of merit, *Team-based Access Control* augments TBAC by providing a more natural way of grouping users in an organization. Specifically, this is accomplished by associating a collaboration context with the activity to be performed.[16]

Although RBAC, and related models, may provide a practical initial strategy, their refinement and maintenance of within large organizations, whose complex workflows undergo continuous evolution, lack an obvious practical application.[17] Several studies have been conducted to find such a solution, often called *role engineering*.3[,18] Generally, there are two types of approaches for role engineering: top-down and bottom-up. The former extracts roles by analyzing business processes and scenarios, and largely depends on knowledge in the minds of an organization's experts. By contrast, the latter approach defines potential roles from existing user-permission relationships, such as the AMM. This style of "bottom-up" role engineering is often automated by data mining algorithms and is also referred to as role mining. However, little research on role engineering leverages access logs, the approach chosen for our research.

Studies of auditing RBAC in clinical health care systems are not yet widely published. Bertino et al[19] conducted a study on intrusion detection in an RBAC-administered database. In doing so, they built a behavioral profile for each role in terms of the SQL commands issued by users. When the behavior of a user was inconsistent with the profile of the corresponding role, they claimed it was an outlier that potentially represented a violation. However, their work differs from ours in it does not assess whether roles may be distinguished from each other, nor did it quantify the extent to which they are distinguishable. Moreover, such an approach was not assessed in the context of an established clinical information system.

To mitigate the rigid nature of access control systems, EMRs often permit users to invoke "break the glass" when they lack sufficient access privileges. When a user opts to issue such an access, the system logs the event for follow-up investigation. This type of approach to auditing works well when the number of exceptions is relatively small. However, there is evidence to suggest that the initial role specification in a healthcare domain do not lead to such scenarios. A study conducted by Røstad et al[20] in the Central Norway Health Region provides a compelling illustration of this situation. After setting up an access control system, they monitored the system for one month. During this time, they observed that more than 50% of 100,000 patients' records were accessed via break the glass and that approximately 45% of users invoked this option. Overall, there over 290,000 exceptions issued, which is significantly more than can be followed up on by a human for investigation.

Methods

The Northwestern Memorial Hospital is an 854 bed primary teaching affiliate for the Feinberg School of Medicine at Northwestern University. All clinicians (including physicians and nurses) retrieve clinical content and enter inpatient notes and orders online using the Cerner Corporation's PowerChart® EMR system. The access logs generated by the system consist of user- and patient-specific information as summarized in Table 1.

Table 1. A summary of the data captured in the Northwestern EMR access logs. For the purposes of this study we represent roles (attribute 3) as a function of Reason, Service, and Location (attributes 5 through7).

Attribute		Description
1	User ID	Login credentials (de-identified)
2	Patient ID	Medical record number (de-identified for cohort)
3	User Position	Assigned role within the medical record system
5	Date and Time Stamp	Dates were randomly shifted in a 365 day period for de-identification purposes
4	Chart Access Reason	Option selected when a chart is first accessed by each user during a hospitalization. Options available are tied to the User Position
5	Orders Entered	Indicates the number of order entered by the user during the current chart access (not used in this study)
6	Location	General location of the patient within the hospital
7	Service	The hospital service caring for the patient as specified by the doctors caring for the patient. If the field is blank (OB service, e.g.), the specialty of the attending physician is used.

When approved by an authorizing entity, (e.g., the Medical Staff Office), each user of the system receives a login ID tied to a User Position. The User Positions enable or prevent access to specific EMR functions. As an example, a medical student orders require co-signing by a physician. As another example, specific administrative roles do not provide comprehensive result flow-sheet access.

As an additional safeguard, users select a "Chart Access Reason" upon first access to a chart for a particular "encounter." The available Chart Access Reasons displayed for selection are tied to the individual's User Position. Selected User Positions with minimal use case scenarios have only one potential Chart Access Reason and are therefore not prompted. An encounter in this context is defined as a hospital visit and is more narrowly specified for the research cohort below.

The cohort of accesses we reviewed covers a 3 month period of time for which patients were either in an "inpatient" status or an "observation" encounter status. Observation status refers to an admission for which discharge is expected within 24 hours. An example of such a log is presented in Figure 1.

User	Patient	Time	Service	User Position	Reason	Location
u_1	p_1	8/4/10	OBSTETRICS	NMH Physician Office - CPOE	Attending Phys/Prov	Ward A
u_2	p_2	12/14/10	OBSTETRICS	NMH Physician - CPOE	Patient Care	Ward A
u_{23}	p_3	12/14/10	PEDIATRICS	Unit Secretary 2	Unit Secretary Orders	Ward B

Figure 1. A fictional example of records in the Northwestern EMR access logs.

Each entry in the access logs corresponds to one access to the EHR, including the information on the user, patient, reason for the access, type of service, location where the access happens, and whether orders or notes activity occurred. For the purpose of privacy, the names of patient and users are replaced by pseudonyms. Moreover, for the purposes of our study, the User Position is considered to be a surrogate for the role . There are 8,095 users and 140 different roles involved by this log. Summary statistics for users and roles with respect to Reasons, Locations, Services, and accesses are provided in Table 2.

Table 2. Summary statistics for the EMR access logs used in this study.

	Users	Roles	Reasons	Locations	Services	Accesses
Total	8095	140	143	58	43	1,138,555
Average per user	-	-	2	10	9	140
Average per role	-	-	4	23	20	8132

Roles and Hierarchies

One of the specific aims of this study is to determine how generalizations of roles in the EMR system could permit more effective access control. However, at the time this study was conducted, there was no explicit relationship established between the user positions in the Cerner EMR. Thus, the investigators collaborated with several clinicians at Northwestern to design a role generalization hierarchy. This hierarchy, a section of which is depicted in Figure 2, was designed as a tree data structure and consists of four levels: 1) *Specific-Position*, 2) *General-Position*, 3) *Conceptual-Position*, and 4) *Employee*. The lowest level in the hierarchy, termed *Specific-Position*, consist of the 140 user positions (i.e., job titles) defined for the current EMR system. The next level up, termed the *General-Position* level, was established by suppressing semantic qualifiers from the user positions. This level consists of 62 nodes in the hierarchy. The qualifiers that were removed represented certain administrative pay (or responsibility) grades or specializations of particular job titles. For instance, the job titles "Dietary 1" and "Dietary 2" were generalized to the common "Dietary".

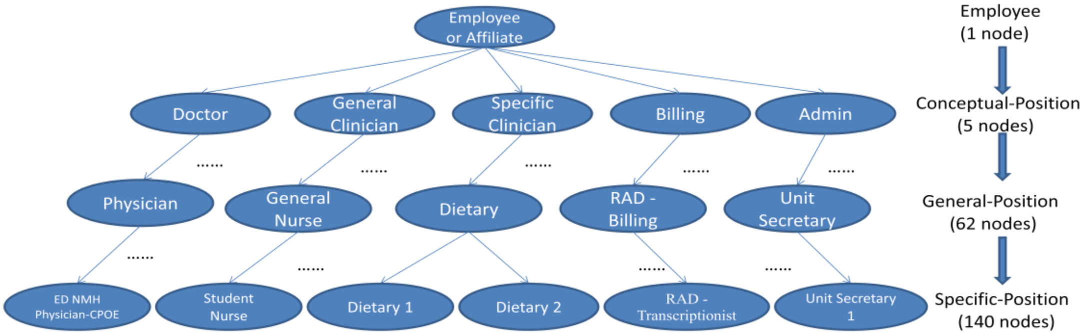

Figure 2. A selection of the role generalization hierarchy designed for this study.

The next level up is called the *Conceptual-Position* level, and was defined with the assistance of the clinicians. This level is composed of five roles defined to capture the anticipated workflow of the healthcare domain. These roles are:

i) *Doctor*: all users whose workflow is most consistent with that of a physician and includes entering orders/notes using the physician tools;

ii) *General Clinician*: all non-physician clinical staff who do not have a restricted domain of work (e.g., nurses who rotate among various care areas);

iii) *Specific Clinician*: all non-physician clinical staff who work in a specific clinical care domain (e.g., Oncology, Cardiology, and Gastroenterology). This group likely represents a more diverse set of users in comparison to the other roles at this level;

iv) *Billing*: users who interact with charts from a billing specific perspective; and

v) *Admin*: users who interact with charts from an administrative and not immediate clinical care perspective.

One of the key reasons why these roles deviate from the terms used in the lower levels is that the "User Positions" address concerns that are less characteristic of the user and instead reflect system design nuances at the time the user was enrolled. An example of this somewhat artifactual name distinction is the existence of positions reflecting whether or not a user had access to CPOE when the user was first enrolled. Now, all physician users have CPOE capability whether or not their user role at inception indicated this was available. Thus, it is anticipated that this higher level view should help mitigate outliers of particular users or job titles. And, at the same time, we believe this level should provide a structure for other healthcare organizations, and EMR systems, to adopt for similar role assignment endeavors.

Finally, and for the purposes of completeness, the highest level in the hierarchy corresponds to the root of the tree and consists of a single role, namely *Employee or Affiliate*.

A Formal Representation of the Users

Before delving into the details of the hierarchy-based role assignment process, we take a moment to formalize the EMR access log system and the resulting transformations. Let $U = \{u_1, \ldots, u_m\}$ be the set of EMR users and let *Role* = $\{role_1, \ldots, role_n\}$ be the set of roles. For reference, we use $|\cdot|$ to represent the number of elements in a set.

Given a database of EMR access transactions, we construct a vector space model for each user. Specifically, let $V = \{v_1, \ldots, v_m\}$ be a set of vectors, where v_i is the corresponding vector for u_i. Each vector is composed of three subvectors, r_i, s_i, and l_i, which represent the access features (i.e., *reason*, *service*, and *location*). Each of these subvectors is defined over the domain of categorical values the feature to which it is associated. For instance, r_i contains a position for each of the 143 specific reasons that could have been selected by a user during a session with a patient's record. For each reason, and for each user, we weight the j^{th} reason and refer to it r_{ij}.

For the purposes of this study, we represent r_{ij} using the term frequency – inverse document frequency (TF-IDF) weighting model[20], which is widely used in text mining:

$$r_{ij} = TF_{ij} * IDF_i = \frac{n_{ij}}{N_i} * log\frac{|U|}{d_j} \tag{1}$$

where n_{ij} is the number of times r_i was invoked by u_i during their EMR sessions , N_i is the total number of accesses issued by u_i, and d_j is the number of users in the system who invoked reason r_j. We apply the TF-IDF schema based on the premise that the more times a user invokes a reason, the more likely the reason is indicative of the user (i.e., TF) and that the smaller the number of users that invoke a reason, the more closely related they are (i.e., IDF). We define s_i and l_i similarly.

A Machine Learning Approach to Role Prediction

The aforementioned vectors provide a summarized view of EMR users' behavior in the healthcare system. We use the vectors as the basis of our role prediction procedure. Specifically, we train a Naïve Bayes classifier[21] with roles as the class labels and the user vectors as input. The task of predication is to determine the class label (i.e., role) for a new user vector. For the Naïve Bayes classifier, the new instance will be assigned the class label according to equation (2).

$$role_{\mathrm{MAP}} = argmax_{role_j \in Role} P(role_j | r_i, s_i, l_i) \tag{2}$$

Using Bayes theorem and assuming conditional independence over the features, we can rewrite the expression as:

$$role_{\mathrm{MAP}} = argmax_{role_j \in Role} P(role_j) \prod\nolimits_{\mathrm{x}} P(r_{ix}|role_j) \prod\nolimits_{\mathrm{y}} P(s_{iy}|role_j) \prod\nolimits_{\mathrm{z}} P(l_{iz}|role_j) \tag{3}$$

In this work, however, the features are continuous variables, which makes it difficult to estimate $P(r_{ix}|role_j)$, $P(s_{iy}|role_j)$, and $P(l_{iz}|role_j)$ directly. As a result, we replace the conditional distribution function with the conditional probability density function, for which a Gauss distribution is used.

Hence, *P(role*$_j$*)* is estimated as the proportion of users in $role_j$, while $P(r_{ix}|role_j)$ is estimated by the Gaussian density function.[21]For the latter, the parameters of μ and σ are estimated by calculating the mean and standard deviation of feature r_{ix} of the users in $role_j$, respectively. $P(s_{iy}|role_j)$ and $P(l_{iz}|role_j)$ are estimated similarly.

The Role-Up Algorithm

The primary goal of this work is to apply EBAM in the context of EMRs to discover, and assess, the appropriateness of users' roles. To achieve this goal, we developed an algorithm called *Role-Up*. The algorithm is based on two foundational premises. First, the more roles in the system, the greater the ability to ultimately manage user groups and achieve a key security goal of *separation of duty*. Second, the more homogenous the user behavior is in a role, the easier it will be to monitor and audit users with respect to their actions. Pseudocode for *Role-Up* is provided in Figure 3.

Here, we provide a high-level walkthrough of the algorithm. First, in step 1, we extract the roles in the middle levels of the hierarchy. Next, in step 2, we employ the Naïve Bayes classifier to predict roles in all of levels of the hierarchy. We use a leave-one-out cross-validation approach to evaluate the predictions. Specifically, role prediction is executed such that the classifier is trained with all, but one, user vectors. The remaining vector is then classified into a role. This procedure is repeated for each user until all users receive predictions. Then, to measure how well the roles are specified, we compute the accuracy of the system:

$$Accuracy = \frac{\#Correct\ Predictions}{\#Predictions} \tag{4}$$

In step 3, we initialize the set of roles to be returned to the administrator as null. In step 4, we calculate a score for each role at the *General-Position* and *Conceptual-Position* levels using the evaluation function of equation (5):

$$S = \alpha R + (1 - \alpha)A \tag{5}$$

R is computed by $(|U| - N_{role})/|U|$, where N_{role} is the number of users covered by this role, and reflects the specificity after generalizing this role to its parent in the hierarchy. A is computed as $(Accuracy_{role} - \overline{Accuracy}_{\mathrm{sub}(role)})$, where $\overline{Accuracy}_{\mathrm{sub}(role)}$ is the average accuracy of all subroles of *role* at the *Specific-Position* level.

```
INPUT: Vectors: A set of EMR user access vectors; Hierarchy: an EMR user role hierarchy; α: a real-valued weighting parameter in the range (0,1). τ: a threshold.
OUTPUT: ROLES: The roles an EMR security administrator should apply for system management
Steps:
1. Let H be the set of roles in the General-Position and Conceptual-Position levels of Hierarchy
2. Let $Accuracy_{role}$be the predictive accuracy score for each role in Hierarchy (the reader is referred to the main text for the details on how Vectors is applied in the accuracy computation)
3. ROLES ← NULL
4. for each role in H
   4.1. $R_{role} = (|U| - N_{role})/|U|$ , where $N_{role}$ is the number users in this role
   4.2. $A_{role} = (Accuracy_{role} - \overline{Accuracy}_{\mathrm{sub}(role)})$
   4.3. $S_{role} = \alpha R_{role} + (1 - \alpha)A_{role}$
5. Sort H by the corresponding scores $S_{role}$ in descending order
6. For each role in H
   6.1. If $S_{role} < \tau$
        6.1.1. break;
   6.2. Else
        6.2.1. ROLES ← ROLES ∪ role
        6.2.2. ROLES ← ROLES – the children of role in HIERARCHY
        6.2.3. H ← H – the children of role in HIERARCHY
 7. Return R
```

Figure 3. Pseudocode for the *Role-Up* algorithm.

Then in Steps 5 and 6, we use a greedy procedure to roll-up the hierarchy. We iteratively select the role with the highest score and implement the corresponding generalization for all of its subroles. This procedure iterates until the highest score is greater than a certain threshold value. At this point, the set of roles is returned to the administrator and the algorithm terminates.

Experiments and Results

Initial Role Prediction

Before applying the *Role-Up* algorithm, we first investigated the predictability of the roles when the system is trained and tested at each level of the role hierarchy. The results of this experiment are reported in Table 3. First, we observe that when the system is trained and tested at the initial *Specific-Position* level (i.e., with 140 user positions) we observe that the system is 51% accurate. In other words, a little more than half of the users can be accurately predicted as their corresponding roles.

Table 3. Predictability of users' roles when the system is trained and tested at various levels of the hierarchy.

Level of Role Hierarchy	**Accuracy**
Specific-Position (original role design)	51.34%
General-Position	52.45%
Conceptual-Position	82.38%

When we step up the hierarchy one level to *General-Position*, we find there is only a marginal gain in performance. We observed that the accuracy increased by approximately 1% to 52.5%. This was somewhat surprising because this level has less than half the number of roles than *Specific-Position.* However, when stepped up one more level to *Conceptual-Position*, we find that the system became significantly more predictable. Notably, the accuracy increased by approximately 30% to 82%.

However, it should be noted that the accuracy of *Specific-Position* and *General-Position* is not uniformly distributed across roles. Rather, there are a significant number of roles that are highly predictable. To illustrate this observation, Figure 4 depicts the distribution of accuracy scores for the roles at each level in the role hierarchy. Notice that for the *Specific-Position* and *General-Position* levels, an accuracy of 0.5 or greater is achieved for a 57 and 39 (or 79% and 81.2%) roles, respectively.

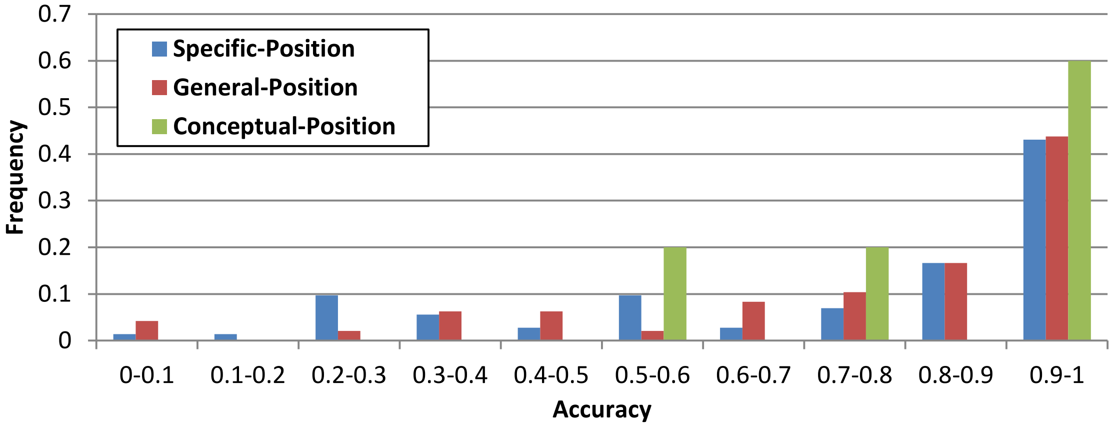

Figure 4. The distribution of role predictability (i.e., accuracy) at various level of the role hierarchy.

To make this result more concrete, Table 4 provides a summary of the five most and five least predictable roles in the *Specific-Position* level. There were ten roles that achieved 100% prediction, so for presentation purposes we randomly selected five roles.

Table 4. *Left*) The most predictable roles and *Right*) the least predictable roles in the system.

Rank	Most Predictable	Accuracy	Users
1 (tie)	AP-Technologist	100%	54
1 (tie)	ED Assistant	100%	26
1 (tie)	ED NMH Physician-CPOE	100%	43
1 (tie)	NMH Resident/Fellow ID Clinic-CPOE	100%	10
1 (tie)	Patient Care Staff Nurse – Lactation	100%	14

Rank	Least Predictable	Accuracy	Users
140	Patient Care Staff Nurse	7.6%	1554
139	Rehab OT	14.3%	28
138	Transfer	20.0%	20
137	View Only PC 3	21.4%	14
136	Patient Care Staff Nurse (Pilot)	22.1%	217

Despite the finding that a significant number of roles received accuracy greater than the system average of 0.5, many of these roles were smaller in terms of the number of users that they cover. Thus, although there are a few outliers, it appears that roles in the EMR system with a small number of users tend to obtain a high predictability while roles with a large number of users are less predictable. It is not surprising too much, because a small number of users implies the role is more specific, and have more specialized responsibilities compared to other roles. As such, roles with a small number of users can be distinguished from other roles more easily. Figure 5 provides a visual depiction of the relationship between the number of users in a role and the prediction accuracy. As also noted in Table 4, the largest role, *Patient Care Staff Nurse*, was also the least predictable. However, some of the larger roles, such as *Med Student – CPOE*, which contained about 500 users, achieved very high prediction rates (i.e., over 80%). This is a clear illustration of why there is no one-size-fits all approach to role engineering or role mining.

A Case Study in Incorrect Predictions

Accuracy provides an indication of how predictable each role (and the system) is, but it obscures the intuition behind *why* the system is failing to predict roles correctly. Thus, we take a moment to illustrate a case study in the types of mispredictions that occur in the system.

As Table 4 shows, the role of *Patient Care Staff Nurse* is the least predictable role among all 140 roles in the *Specific-Position* level. Thus, it is useful to know which roles the system has predicted these users belong to. Table 5 depicts the probabilities for the five least correct predictions for *Patient Care Staff Nurse* and *Transfer*, respectively. Semantically (and literally), they are very similar roles, and we may infer that these roles are often assigned the same tasks as *Patient Care Staff Nurse*. Hence, merging *Patient Care Staff Nurse* with *Patient Care*

Staff Nurse – Lactation, RAD – Nurse or other similar roles in this table should lead to a more predictable role. This was one of the inspirations for the expert design of reasonable role hierarchies.

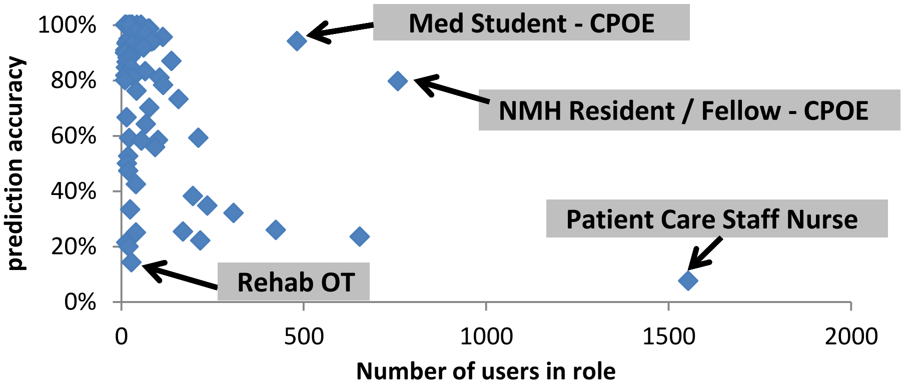

Figure 5. A plot of accuracy of role as a function of the number of users in the role.

Table 5. Most likely incorrect role predictions for *Left*) *Patient Care Staff Nurse* and Right) *Transfer*.

Predicted Role	Percent
Patient Care Staff Nurse - Lactation	19.6%
View Only PC 1	14.3%
RAD – Nurse	14.0%
Patient Care Staff Nurse (Pilot)	10.4%
SN-RN/Customer Service	5.8%

Predicted Role	Percent
RX-Pharmacist	15.0%
Patient Care Staff Nurse - Lactation	10.0%
"Unspecified"	10.0%
Unit Secretary 1	10.0%
SN-Management	5.0%

Table 6 provides an indication of which roles were being confused in the prediction process. Specifically, it reports on the conditional probability of predicting a role given the original role. For instance, there was an 85% chance of predicting *Rehab PT* if the original role was *Rehab OT*. Similarly, there was a 60% chance of predicting *Rehab OT* if the original role was *Rehab PT*. This is further justification for generalizing roles for EMR management purposes.

Table 6. Most likely incorrect predictions among all of the predictions.

Original Role	Predicted Role	Probability
Rehab OT	Rehab PT	85.7%
Patient Care Staff Nurse - Agency	Patient Care Staff Nurse - Lactation	75.0%
Rehab PT	Rehab OT	60.0%
View Only PC 3	Patient Care Staff Nurse - Lactation	50.0%
Medical Records - Scanner	Medical Records	47.4%

Rolling-up Role Prediction

The following set of experiments report on the application of the *Role-Up* algorithm. For the purposes of this work, we set the threshold in the algorithm equal to α. In contrast to the earlier experiments, *Role-Up* permits the hierarchy to allow for roles managed at different levels in the hierarchy. Table 7 shows the number of roles recommended by the approach and the accuracy of the resulting system under different values of α. From this table we wish to highlight three findings. First, there is a tradeoff in specificity in roles and accuracy of the system. Notice that when α is low, between 0.1 and 0.4, the number of roles is relatively small (i.e., 27), but the accuracy of the system is relatively high (i.e., approximately 63%). And, when α is higher, such as at 0.8, the specificity of the system is relatively high (i.e., 60 roles), but the accuracy is lower (i.e., approximately 52%).

Table 7. Results of rolling-up the hierarchy under different alpha α.

α	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Number of Roles Recommended	27	27	27	27	54	55	55	60	64
Accuracy of Role Predictions	63.3%	63.3%	63.3%	63.3%	49.9%	50.2%	50.2%	51.8%	51.3%

Second, we note that α ≥ 0.8 appears to be the most appropriate choice. When the system is set at this level, *Role-Up* achieves an accuracy that is slightly better than that of original role designation while maximizing the number of roles retained for all α settings.

Third, a security administrator might also wish to consider $\alpha \leq 0.4$. At this setting, the system proposed by *Role-Up* achieves high accuracy, but at the loss of a significant number of roles.

As an illustration of the tradeoff between high and low α's, Figure 6 provides a distribution of the accuracy per role.

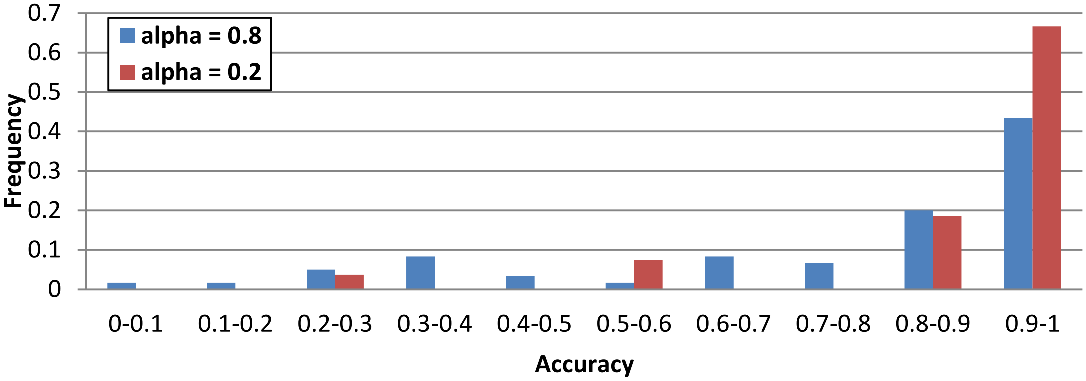

Figure 6. Distribution of the accuracy for the system when α is set to 0.2 and 0.8.

Discussion and Conclusions

Summary of Findings

Over the past several years, the healthcare community has expanded its adoption and utilization of information technologies. In the United States, this is due, in part, to incentives provided by the Health Information Technology for Economic and Clinical Health Act (HITECH), enacted as part of the American Recovery and Reinvestment Act of 2009.[22] At the same time, there is a growing movement to facilitate the dissemination of patient information across organizational boundaries, through health information exchanges, to enable more effective care and reduce costs. However, before such exchanges can be executed, appropriate security policies for access to patient-specific information need to be set in place.

This study illustrates that usage patterns of a commercial EMR system can enable accurate prediction of certain roles in a healthcare system. Additionally, we illustrated that an automated approach can be leveraged to integrate role hierarchies with information learned from EMR access logs to improve role management. These findings are notable because they suggest that RBAC, in combination with some EMR usage mining, may assist in minimizing the management of access to an EMR system. Moreover**,** the increased specificity provided by User Positions versus higher levels within the role hierarchy enables more detailed access pattern analysis.

These results are further notable because a recent report from the President's Council of Advisors on Science and Technology (PCAST) recommended that emerging health information architectures should leverage security principles that have proven successful in a range of industries beyond healthcare.[23] In particular, the PCAST report alludes to RBAC as a foundation upon which such policies can be defined. With respect to the healthcare domain, RBAC is intended to be a scalable framework for commissioning (and decommissioning) users with access rights to functions (e.g., order issuance) or elements of a clinical information system (e.g., a specific patient's record). And, notably, various commercial EMR systems have integrated such security frameworks into their design. Yet, as the PCAST report acknowledges, healthcare organizations rarely execute RBAC on the scale found in other domains.

Limitations of the Study

There are several limitations of this study that we wish to highlight to help pave the way for future research in this area. The first drawback of this study to note is that the original roles; i.e., User Positions, were defined over time and not in a single security engineering design. As a consequence, in certain cases, User Position designations represent vestigial remnants of a prior CPOE roll-out strategy. That is, for a time, selected physician user roles were not entering orders online, although now all physician User Positions include this functionality. Additionally, User Position assignments fail to take into account some workflow idiosyncrasies. For example, hospital medicine physicians, or hospitalists, often serve as a pilot physician group requiring their User Position to be distinct from other internal medicine physicians. Hospitalists may also work as a non-hospitalist (e.g., as a teaching attending), however, and, at those times, their chart access patterns would differ from their hospitalist service rotations.

The second drawback of this study is a function of the *Role-Up* algorithm. Currently, the roll-up procedure is guided by a greedy heuristic. Specifically, in each iteration, the algorithm generalizes the set of sibling roles (i.e., roles with a common parent) that provide the greatest gain in predictive accuracy without sacrificing much role specificity. However, this process does not guarantee the discovery of a system that maximizes the number of roles and system accuracy. Thus, as a next step, we intend to determine how to mitigate the greedy nature of the search.

Acknowledgements

This research was supported by grants CCF-0424422 and CNS-0964063 from the NSF, R01-LM010207 from the NIH, and HHS-90TR0003/01 from the ONC. The authors thank You Chen, Steve Nyemba, and Yuan Xue of Vanderbilt University, Sonja Jahid from University of Illinois at Urbana Champaign, and Dan Cushman, Sanjay Mehrotra, and He Zhang from Northwestern University for insightful discussions during this research. The authors also thank Nathan Sisterson for supplying the dataset used in this study.

References

1. Sandhu R, Coyne E, Feinstein H, Youman C. Role-based access control. IEEE Computer. 1996; 26: 38-47.
2. Choe J, You S. Web-based secure access from multiple patient repositories. Int J Med Inform. 2009; 77: 242-8.
3. Neumann G, Strembeck M. A scenario-driven role engineering process for functional RBAC roles. Proc ACM SACMAT. 2002: 33-42.
4. Asaro P, Ries J. Data mining in medical record access logs. Proc AMIA Symp; 2001: 855.
5. Gallagher R, Sengupta S, Hripcsak G, Barrows R, Clayton P. An audit server for monitoring usage of clinical information systems. Proc AMIA Symp. 1998: 1002.
6. Zhou Z, Liu B. HIPAA Compliant auditing system for medical images. Comput Med Imaging Graph. 2005; 29: 235-41.
7. Ferreira A, Cruz-Correia R, Antunes L, et al. How to break access control in a controlled manner. Proc. 19th IEEE Symp on Computer-Based Medical Systems. 2006: 847-854.
8. US Department of Health and Human Services, Office for Civil Rights. Standards for protection of electronic health information; final rule. Federal Register 45 CFR PT. 164. 2003.
9. Amatayakul M. Think a privacy breach couldn't happen at your facility? Healthc Q. 2009; 12: 61-5.
10. Dimick C. A guide to California's breaches: first year of state reporting requirement reveals common privacy violations. J AHIMA. 2010; 81: 34-6.
11. Lampson B. Protection. Proc ACM SIGOPS Operating Systems Review. 1974; 8: 18-24.
12. O'Connor A, Loomis R. Economic analysis of role-based access control. RTI Project Number 0211876, Research Triangle Institute. December 2010.
13. Motta G, Furuie S. A contextual role-based access control authorization model for electronic patient record. IEEE Trans Inf Technol Biomed. 2003; 7: 202-7.
14. Peleg M, Beimel D, Dori D, Denekamp Y. Situation-Based Access Control: privacy management via modeling of patient data access scenarios. J Biomed Inform. 2008; 41: 1028-40.
15. Beimel D, Peleg M. Comparing the context and the SitBAC models for privacy preservation in terms of model understanding and synthesis. AMIA Annu Symp Proc. 2008: 874.
16. Georgiadis C, Mavridis I, Pangalos G, Thomas R. Flexible team-based access control using contexts. In Proc. of ACM Symposium on Access Control Model and Technology. 2001: 21-7.
17. Malin B, Nyemba S, Paulett J. Learning relational policies from electronic health record access logs. J Biomed Inform. 2011: 44: 333-42.
18. Kuhlmann M, Shohat D. Role mining-revealing business roles for security administration using data mining technology. Proc. ACM Symp Access Control Models and Technologies. 2003: 179-186.
19. Bertino E, Terzi E, Kamra A, Vakali A. Intrusion detection in RBAC-administered databases. Proc Annual Computer Security Applications Conference. 2005: 170-82.
20. Røstad L, Edsberg O. A study of access control requirements for healthcare systems based on audit trails from access logs. Proc Annual Computer Security Applications Conference 2006: 175-86 Berry M. Survey of Text Mining: Clustering, Classification, and Retrieval. Springer-Verlag. New York. 2003.
21. Han J, Kamber M. Data Mining: Concepts and Techniques, 2nd edition, Morgan Kaufmann, 2006
22. American Recovery and Reinvestment Act of 2009. Public Law 111-5. Feb 17, 2009.
23. President's Council of Advisors on Science and Technology. Report to the President: realizing the full potential of health information technology to improve healthcare for Americans: the path forward. Office of Science and Technology Policy. 2010.