
ROLE PREDICTION USING ELECTRONIC MEDICAL RECORD SYSTEM AUDITS

EXTENDED ABSTRACT BY

WEN ZHANG¹, CARL A. GUNTER², DAVID LIEBOVITZ³, JIAN TIAN¹, AND BRADLEY MALIN^{1,4}

¹DEPT. OF ELECTRICAL ENGINEERING & COMPUTER SCIENCE, VANDERBILT UNIVERSITY

²DEPT. OF COMPUTER SCIENCE, UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAGNE

³DEPT. OF MEDICINE, NORTHWESTERN UNIVERSITY

⁴DEPT. OF BIOMEDICAL INFORMATICS, VANDERBILT UNIVERSITY

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) [SandhuCFY96], groups access privileges into collections called *roles* and then assigns users to roles to determine their access privileges. This is commonly achieved by reviewing 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 do their assigned tasks. A second strategy, which we group under the general heading of Experience Based Access Management (EBAM) [GunterLM], emphasizes accountability and the use of audit data to reprimand abuse. An often referenced strategy for EBAM is to manually review audit logs of VIPs to determine infractions. Another strategy, called “break-the-glass” security, discourages abuse by warning users that certain types of access are manually reviewed.

However, at the current point in time, RBAC and EBAM are used without much common foundation. This is unfortunate because there seems to be significant opportunities for synergy between the techniques. For example, 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. On the other hand, auditing analytics can show how more appropriate definitions for roles, or roles that are context-specific, may be applied to restrict access so that fewer checks are required on audits.

We consider how to use audit logs to predict whether a given user is associated with a given role, a concept we call *role prediction*. This extended abstract highlights and contextualizes several findings from a longer manuscript [ZhangGTM]. 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, indicating whether the right relationships have been used. These capabilities provide a useful link between RBAC and EBAM. In our current work, we address two specific questions: (1) To

what extent do expert-defined job titles in a hospital predict statistical behavior of personnel using these titles as roles? (2) To what extent does a hierarchical organization of roles permit more accurate predictions? Question (1) relates to the ability of audit logs to predict, for instance, the chance that a role prediction of a user as a student nurse might be inferred incorrectly for an emergency department biller. Question (2) relates to the extent to which this prediction capability is changed if one moves up the hierarchy and considers, for instance, whether a nurse is likely to be confused with a biller.

Our study of role prediction comprises a set of learning techniques to address the first of these two questions and an algorithm we call “*Role-Up*” to address the second. We tested these techniques on access log data from a commercial inpatient electronic medical record system at Northwestern Memorial Hospital, an 854 bed primary teaching affiliate of the Feinberg School of Medicine at Northwestern University. The cohort of accesses reviewed covers a three-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. This is studied with respect to a collection of 140 positions used as a parameter in accesses. We describe the audit data and roles in turn.

An audit record consists of a tuple with the following fields: user, patient, time, service, user position (role), reason (for access), and location (portion of the hospital where the patient is located). Example: a user u accessed the record of patient p at time t in *OBSTETRICS* service as an *NMH Physician Office CPOE* for reason *Attending Phys/Prov* while the patient resided in *Ward A*. Our data set covered 8095 users in 43 services, with 140 positions using 143 reasons at 58 locations. There were 1,138,555 accesses with a user making, on average, 140 accesses and a position making, on average, 8132 accesses.

We collaborated with several clinicians at Northwestern to design a role generalization hierarchy for this study. This hierarchy, a section of which is depicted in Figure 1 consists of four levels. The lowest level is termed *Specific-Position* and consists of the 140 positions. The next level up, termed the *General-Position* level, removes

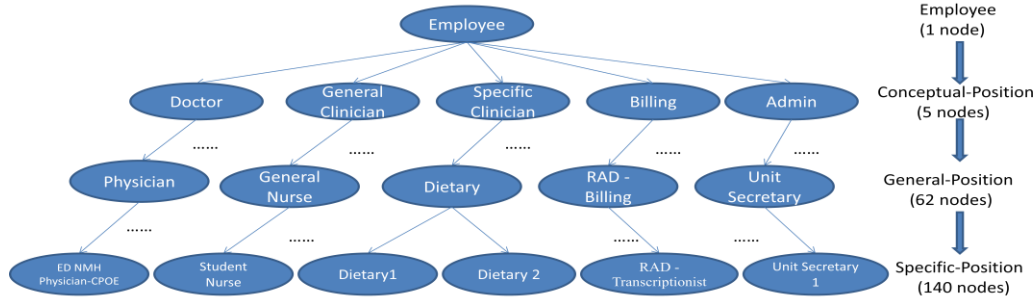


Figure 1: A section of the role hierarchy developed for this study. The *Specific-Position* level consists of the original job titles. The *General-Position* level suppresses administrative grades of job titles. The *Conceptual-Position* level consists of roles in anticipated healthcare workflows.

semantic qualifiers from the user positions. This level consists of 62 nodes in the hierarchy. The qualifiers that we removed were representative of 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”. 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. 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*.

Our analysis technique involved creating a vector space model for each user u_i consisting of vectors r_i , s_i , and l_i for reason, service and location respectively. These vectors, on which we trained a Naïve Bayes classifier, a classical machine learning algorithm, provide a summarized view of user behavior. The task of role prediction is to determine the class label (*i.e.*, role) for a new user vector. For the classifier, the new instance will be assigned the class label according to the equation $role_{MAP} = \text{argmax}_{role_j \in Role} P(role_j | r_i, s_i, l_i)$. Let us sketch the role-up algorithm as a series of steps. First, in step 1, we extract the roles from the hierarchy. Next, in step 2, we employ the classifier to predict roles in all of levels of the hierarchy. We use a leave-one-out cross validation approach to evaluate the predictions. (*i.e.*, the classifier is trained with all, but one, user). The remaining vector is then classified into a role. This procedure is repeated for each user. Then, to determine how well the roles are specified, we measure the accuracy of the system as the ratio of correct predictions to predictions. 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. Then in Steps 5 and 6, we use a greedy procedure to “roll-up” the hierarchy. Among the roles extracted in step 1, we iteratively select the role with the highest score and implement the corresponding generalization for all of its sub roles. This procedure iterates until the highest score is less than a threshold. At this point, the set of roles is returned to the administrator. The algorithm provides a parameter that permits the results to bias toward accuracy in role predictions vs. specificity in roles (*i.e.*, solutions lower in the hierarchy)

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 were 51%, 52%, and 82% accuracy at the *Specific-Position*, *General-Position*, and *Conceptual-Position* levels respectively. So, a little more than half of the users can be accurately predicted as having their corresponding job titles. Conversely, nearly half of the users may not be assigned to a role that reflects with their daily behaviors. When we step up the hierarchy to *General-Position*, there is only a marginal gain in performance, which was surprising because this level has less than half the number of roles than *Specific-Position*. However, when stepped up to *Conceptual-Position*, the system is significantly more predictable.

Role-Up solutions permit disparate roles to be managed at different levels in the hierarchy. We highlight several findings for demonstration. First, when biased toward accuracy, the number of resulting roles is relatively small (*i.e.*, 27), but the accuracy of the system is relatively high (*i.e.*, approximately 63%). When biased toward specificity, the number of roles is relatively high (*i.e.*, 60 roles), but the accuracy is lower (*i.e.*, approximately 52%).

This study illustrates that usage patterns of an EMR system can enable accurate prediction of certain roles. Additionally, integrating role hierarchies with information learned from EMR access logs, an automatic method can discover appropriate role management. Some drawbacks may exist. First, user positions (roles) are not defined in a single security engineering design but over time. Second, role-up algorithm does not lead to an optimal role sets. Finally, the extent to which *Role-Up* permits sufficient management will need to be assessed through empirical assessments with role engineers in HCOs.

References

- [GunterLM] Gunter, C. A., Liebovitz, D., Malin, B. Experience-Based Access Management: A Lifecycle Framework to Evolve Identity and Access Management systems. *IEEE Security and Privacy Magazine*. To appear.
- [SandhuCFY96] Sandhu, R., Coyne, E., Feinstein, H., Youman, C. Role-based access control. *IEEE Computer*. 1996; 26: 38-47.
- [ZhangGTM] Zhang, W., Gunter, C. A., Liebovitz, D., Malin, B. Role Prediction using Electronic Medical Record System Audits. *In: AMIA. 2011: to appear.*