Preventing Unwanted Social Inferences with Classification Tree Analysis

Sara Motahari, Sotirios Ziavras, Quentin Jones
New Jersey Institute of Technology
{sg262, ziavras, quentin.jones}@njit.edu

Abstract

A serious threat to user privacy in new mobile and web2.0 applications stems from 'social inferences'. These unwanted inferences are related to the users' identity, current location and other personal information. We have previously introduced 'inference functions' to estimate the social inference risk based on information entropy. In this paper, after analyzing the problem and reviewing our risk estimation method, we create a decision tree to distinguish between high risk and normal situations. To evaluate our methodology, test and training datasets were collected during a large mobile-phone field study for a location-aware application. The classification tree employs our two inference functions, for the current and past situations, as internal nodes. Our results show that the achieved true classification rates are significantly better than approaches that employ other available features for the internal nodes of the trees. The results also suggest that common classification tools cannot accurately capture the information entropy for social applications. This is mostly due to the lack of enough training data for high-risk, low-entropy situations and outliers. Thus, we conclude that estimating the information entropy and the relevant inference risk using a pre-processor can yield a simpler and more accurate classification tree.

Key Words: Reasoning Under Fuzziness or Uncertainty, Knowledge Representation and Reasoning

1. Introduction

The continuous input of information in technological environments involving multiple users can create numerous unaddressed risks related to user privacy. Current social computing applications, such as Facebook, enable users to exchange messages, reveal aspects of their profile, and even find profile-based matches. Location-based applications leverage location, mobility, or proximity information to support navigation, provide recommendations, recommend matches, etc. The resulting use and sharing of personal information raises serious privacy concerns. Previous efforts to protect the users' privacy have made considerable advances in terms of computer and network security [1], user control mechanisms [2, 3], ethical considerations, and privacy policies [4]. However, the collaborative and pervasive nature of new ubiquitous social computing (USC) applications can give users the ability to leverage background knowledge about the social environment/context to make *social inferences* [5]. Social inferences can result from an inferrer taking advantage of information revealed by the application and, sometimes, of information available outside of the specific application. Unwanted inferences can impact users associated with these applications, in relation to identity, location, activities, social relations, and profiles. Our work has identified two categories of social inferences [5]:

- *Instantaneous Social Inferences* (e.g. my cell phone shows that I have a romantic match, Bob, who is nearby and I can only see two people around me. One of them must be Bob, thus increasing my chances of identifying him).
- *Historical Social Inferences* through persistent user observation (e.g. two nicknames are repeatedly shown on the first floor of the gym where the gym assistant normally sits. One of them must be the gym assistant).

Numerous social computing applications deal with privacy concerns through access control [2, 6] (e.g., Facebook enables users to set privacy preferences) and a few of them have even employed machine learning (e.g., Cornwell et al. [6] used case-based reasoning to capture the users' privacy preferences). However, such control models are not designed to prevent unwanted social inferences.

Existing inference prevention methods [7, 8] are inadequate in addressing social inferences particularly because of two reasons: (a) inferrers typically also utilize available information outside of the application (i.e. background knowledge) as a premise for inferences and (b) the sensitivity of user information may be of dynamic nature based on context, such as time and location. Most of the researchers view inferences as threats to information security or database confidentiality [7, 8].

Although many researchers have applied artificial intelligence (AI) techniques in the domain of security protection [9, 10], the use of AI to address the inference problem, especially in the context of USC applications, remains under-researched.

In previous work we have defined the social inference problem in the USC context and have also shown the relation between the risk of social inferences and information entropy [11, 12]. We have presented a methodology to estimate the information entropy based on novel modeling of a potential inferrer's background knowledge. We have also shown that information entropy is a strong predictor of social inference risks after we devised *inference functions* to encapsulate information entropy. However, it is not obvious how each individual inference function (quantifying instantaneous and historical inferences, respectively) or a combination of them can be used to detect a high risk situation that should alert an affected user or should require the USC system to take a preventive action. Furthermore, it has not been investigated whether using other observable features could improve the risk classification rate.

In this paper, we briefly review our risk estimation theoretical framework. We then show that AI techniques can be employed to identify inference risks in order to distinguish between high-risk and normal (i.e. low-risk) situations. Towards this objective, we first calculate values for the entropy-based instantaneous and historical inference functions under various scenarios comprising real data obtained from a proximity-based field study. We then generate classification and regression trees (CARTs) that distinguish between high risk and normal situations.

Our results show that the generated decision trees are usually simple while also being comprehensive as they utilize both of the individual (i.e. instantaneous and historical) inference functions as well as their combination. Whereas most of the AI-based security protection and intrusion detection systems examine all available features to detect intrusion or misuse activities/patterns [13], our work here shows that adding more features (in the form of independent variables) does not improve the classification success rate. We also show that feeding the system with all of the available features as raw data (i.e. without entropy calculations) increases the complexity of the tree without improving the classification accuracy.

2. Background

We first present three categories of relevant research efforts that attempt to enhance user privacy: ethics of information management, access control systems, and information and network security. AI has been vastly employed in security protection and occasionally by access control systems.

1. *Ethics, principles, and rules:* Privacy concerns can be partially addressed through the application of ethical principles and rules. Langheinrich [4] defines the principles of fair information practices as openness and transparency, individual participation, collection limitation, data quality, use limitation, reasonable security, accountability and explicit consent.

2. *Access control systems:* They provide a user interface to set privacy preferences and directly control people's access to other user's information based on privacy settings. Access control systems providing an interface to protect user privacy started with internetworking [14, 15], and progressed to context-aware and ubiquitous computing systems [2, 16, 17]. AI has rarely been used to improve access control systems for ubiquitous computing. For example, Cornwell et al. [6] employed case-based reasoning to learn and predict the users' privacy preferences in a scheme achieving a maximum accuracy of 80%.

3. *Security protection:* It handles the following aspects [18]:

- Availability (services are available to authorized users).
- Integrity (information is free from unauthorized manipulation).
- Confidentiality (only an intended user can access the respective information).
- Accountability (actions of any entity should be uniquely traceable).
- Assurance (guarantee that all security measures have been properly implemented).

The inference problem is mostly known as a security risk targeting system-based confidentiality. Two types of techniques have been proposed to identify and remove inference channels. One makes use of semantic data modeling methods to locate inference channels in the database design, in order to redesign the database for the removal of these channels. The other one evaluates database queries to understand whether they lead to unauthorized inferences. These techniques have been studied for statistical databases [19], multilevel secure databases [20, 21] and general purpose databases [7, 22]. A few researchers have also addressed the inference problem for data mining [8, 23-25]. Denning and Morgenstern employed classical information theory [26] to measure the inference chance in the realm of multilevel databases [27, 28]. Our approach adapts their work for social computing environments.

The challenge of social inferences cannot be addressed adequately enough by existing techniques because of the following reasons: (a) an inferred user attribute may not be stored in the social application database; (b) background knowledge available to the inferrer outside of this database is often the premise for inferences; (c) information revealed in the past through this application can enable historical inferences; (d) the sensitivity of user information may be of dynamic nature; and (e) social inferences do not necessarily result from deductive reasoning.

AI is widely used in information assurance and confidentiality, as well as system integrity and availability. Many intrusion detection and prevention systems employ neural networks, decision trees, and Bayesian networks [9, 10]. However, social inferences remain very under-researched and, to the best of our knowledge, the social inference problem has not been researched before with AI techniques.

In Section 3, we first modify and extend Denning and Morgenstem's formulation [27] in order to then introduce inference functions that can predict the risk of social inferences in mobile and social applications. In Section 4, we first describe our mobile-phone based actual field study. This study then populates our database for the creation of a decision/classification tree. This tree aids the process of understanding how to combine and compare instantaneous and historical inference functions to predict the risk of social inferences. Our results with analysis are presented in Section 5.

3. Prediction and Classification of High Risk Situations

In this section we review the social inference problem, the relation between social inferences and information entropy, and our entropy-based framework that models users' background knowledge to predict the social inference risk. We then define our instantaneous and historical inference functions.

3.1. Social Inference Risk Prediction

Our theoretical framework is based on this fact: as we collect more information about a user, such as his/her contextual situation, our uncertainty about other attributes, such as his/her identity may be reduced; consequently, this process increases the probability of our correctly guessing some user attributes. This uncertainty can be measured by *information entropy*. *Information*, as used in information theory for telecommunications [26], is a measure of the decrease of uncertainty in a signal value at the receiver site. Here we use the fact that the more uncertain or random an event (outcome) is, the higher the *entropy* it possesses. In the realm of the inference problem under study, as the inferrer collects more information about an entity or attribute (such as another user or a location), the number of possible entities that match known sets of attributes decreases; this results in reduced information entropy.

To explain this in more detail, we refer to an example from our user experiment in [12]. This experiment provides an example for the herein presented work. Bob engages in an online communication with Alice. At the start of communication Bob does not know anything about his chat partner. He is not told the name of the chart partner or anything else about her, so all users are equally likely to be his partner (the user probability is uniformly distributed for this chat session). Thus, the information entropy has the highest possible value. After Alice starts chatting, her language and chat style help Bob determine her gender and home country [12]. At this point, users of the same gender and nationality are more likely to be his chat partner. Thus, the probability for Bob to guess his chat partner is no longer uniformly distributed over the users and the entropy decreases. After a while, Alice reveals that she is Hispanic and also plays for the university's women's soccer team. Bob, who has prior knowledge of this soccer team, knows that it has only one Hispanic member. This allows Bob to infer Alice's identity at physical appearance granularity. In summary, social inferences happen when newly collected information reduces an inferrer's uncertainty about an attribute to a level that she/he could deduce that attribute's value for an entity/user. Collected information includes not only the information provided to users by the system, but also other information available outside of the application database or background knowledge.

We formally define the social inference problem as follows [11, 12]. Information Φ can be inferred from information Q if knowing Q reduces the uncertainty about Φ by bringing the entropy of Φ down to a risky *threshold*. Q can be safely known by user $\boldsymbol{A}$ if $\boldsymbol{A}$ is permitted to know everything that can be inferred from Q. This condition can be expressed as follows: $\forall\Phi,\ [(H(\Phi|Q) < threshold \wedge PK_A(Q)) \Rightarrow PK_A(\Phi)]$, where $H(\Phi|Q)$ is the conditional entropy of Φ given Q and $PK_A(Q)$ means $\boldsymbol{A}$ is permitted to know Q .

Q includes the potential inferrer's (i.e. $\boldsymbol{A}$) background knowledge as well as answers to all of his/her earlier queries facilitated by this social application. Before $\boldsymbol{A}$ knows Q (which means A has no relevant knowledge), Φ can take any of its possible values with equal probability, thus yielding maximum entropy from ***A's*** perspective. The maximum entropy of Φ, $H_{\max}$, is calculated as follows:

$$H_{\max} = -\sum_{1}^{X} P.\log_2 P = -\log_2(1/X) \qquad \textbf{(1)}$$

where P=1/X and X is the maximum number of entities (users) related to the application. We assume that $\boldsymbol{A}$ does not have any relevant prior knowledge (background- or queries-based).

After estimating all the information available to the inferrer (i.e. Q), we can calculate the conditional information entropy of attribute Φ as defined in information theory:

$$H_{c1} = H(\Phi \mid Q) = -\sum_{i=1}^{V} P1(i).\log_2 P1(i) \qquad \textbf{(2)}$$

where H_{c1} is the *instantaneous entropy* of Φ. V is the number of possible values for attribute Φ. $P1$(i) is the probability that the ith possible value is thought by the inferrer to be the correct one. $P1(i)$ is the posterior probability of each value given Q.

In the case of historical inferences, Q includes the answers to previous queries starting at the current time and going back a long amount of time equal to T (predetermined value). We distinguish this case using Q' for the available information and H_{c2} for the *historical entropy* of Φ:

$$H_{c2} = H(\Phi \mid Q') = -\sum_{i=1}^{V'} P2(i).\log_2 P2(i) \tag{3}$$

Let us now illustrate the effectiveness of our entropy-based model using our earlier example from the user study. Alice is engaged in an initially-anonymous on-line chat with Bob. After a while her chat style may enable Bob to guess her gender and home country. Then, she reveals her Hispanic heritage and gender, as well as her affiliation with the soccer team. Let Φ be Alice's identity at name or face granularity. Before the last chat step, Q may comprise her gender and home country, thus changing the probability distribution of possible values as below:

$$P1(i) = \begin{cases} \varsigma.\sigma/X3 + \varsigma.(1\text{-}\sigma)/(X1) + (1-\varsigma).\sigma/(X2) & \text{for users of the same gender and country} \\ \varsigma.(1\text{-}\sigma)/(X1) + (1-\varsigma).(1\text{-}\sigma)/V & \text{for users of only the same gender} \\ (1-\varsigma).\sigma/(X2) + (1-\varsigma).(1\text{-}\sigma)/V & \text{for users of only the same country} \\ (1-\varsigma).(1\text{-}\sigma)/\mathrm{V} & \text{for the rest of users} \end{cases}$$

where V is the total number of potential users for this social application, ζ is the probability of Bob correctly guessing Alice's gender, σ is the probability of Bob correctly guessing her home country [12], $X1$ is the number of female users,, and $X2$ is the number of users having Alice's ethnicity.

After Alice actually reveals her gender and team membership, Q is modified to account for the newly revealed information (gender, ethnicity, and soccer team player) and relevant background knowledge possessed by Bob. Since the personal profiles were found to be part of the inferrer's background knowledge in such applications, V is now the number of users that satisfy Q, which is the number of Hispanic female soccer players. At this point, V=1, $P1(i)$=1, and the entropy is at its minimum level.

We now define the instantaneous and historical inference functions based on the corresponding entropies, as follows.

$$INF_1(\mathrm{Q} \rightarrow \Phi) = (H_{\max} - H_{c1}) / H_{\max} \tag{4}$$

where $INF_1(\mathrm{Q} \rightarrow \Phi)$ is the *instantaneous inference function*, $H_{\max}$ is the maximum entropy as defined in (1), and H_{c1} is the instantaneous entropy as defined in (2). The maximum entropy is used for normalization, that is. $INF_1(\mathrm{Q} \rightarrow \Phi)$ always lies between 0 and 1. When it is too close to 1, there is a high inference risk. Similarly, the *historical inference function* is defined as:

$$INF_2(Q \rightarrow \Phi) = \left\{ \begin{array}{ll} \dfrac{H_{\max} - H_{c2}}{H_{\max}}, if & H_{c2} < threshold \\ \lambda \qquad\qquad\quad , if & H_{c2} \geq threshold \end{array} \right\} \tag{5}$$

where $H_{\max}$ is the maximum entropy as defined in **(1)**, and H_{c2} is the historical entropy as defined in **(2)**. The historical inference function produces values between 0 and 1 until the entropy reaches the threshold. The threshold is preset based on each user's anonymity preferences [11, 12]. When the entropy reaches this threshold, $INF_2(\mathrm{Q} \rightarrow \Phi)$ starts counting the new queries involving Φ and producing values higher than one.

3.2. Classifying High Risk and Normal Situations

We have previously conducted a laboratory experiment in the domain of computer-mediated communications and verified that information entropy is the best predictor of inference risks [12]. More specifically, if the instantaneous entropy is lower than the preset threshold, then the situation is of high risk for inferences. We also suggested setting the entropy threshold based on each user's anonymity preferences. If a user prefers to be indistinguishable from **U**-1 other users, V in **(2)** can be replaced by **U** to obtain the entropy value for the instantaneous entropy threshold. The threshold value can then substitute H_{c1} in **(4)**. The higher the value derived for the historical inference function, the higher the inference risk. Actually, after λ=K queries there may be high inference risk. However, it is not obvious at what point (value of K) a situation is classified as high risk based on the historical inference function. It is also unclear whether a combination of relatively high values for the instantaneous and historical functions can indicate a high risk situation. In addition to addressing these questions, we aim to verify that adding more features to the risk-decision process results in little or no gain. Also, the same classification accuracy cannot be achieved without entropy calculations. The procedure is explained in the next section.

4. Risk Classification Method

A classification tool such as a decision tree can be employed to answer the above questions for high risk situations. In this section, we first summarize results from the actual user study that provided the test data to train our classification tree. We then explain the decision tree generation process.

4.1. Step 1- Field Study and Data Collection

We collected data through a mobile-phone field study. All study subjects were students of our medium-sized urban university who were offered 40 dollars for answering a pre-study survey. Then, they carried our Windows-based mobile phones for four weeks, and answered several questionnaires while using the phone. One hundred sixty nine students registered for the study. One hundred twenty nine of them were active participants throughout our four-week experiment.

In phase I of the study, the subjects entered their contact information, demographic information (such as age and gender), and answered questions related to their physical appearance (such as height and body type). Phase II involved installation of a location estimation system that continuously tracks the users' location on campus. In phase III, we installed our '*Nearby*' application to display on each phone the nicknames of the users in the vicinity of a phone's user. Each user initially chose a nickname to display, instead of using his/her real name. Phase IV involved the last part of our data collection process using pop-up questionnaires conforming to the Context-Aware Experience Sampling Method (CA-ESM) [29]. With CA-ESM, a questionnaire popped up every time a subject changed location and stayed in the new location for at least five minutes or when she/he had not answered a questionnaire for at least two hours. The questionnaires asked subjects how often they visited the location they were currently at, how many people they then saw in their physical vicinity, and how many of them were their friends or acquaintances. In subsequent questions, the subjects were asked questions about the nicknames they saw with the nearby application, what they could guess about the identity of each nickname owner, and how they could map them to people in their vicinity. They elaborated on their guesses by mentioning potential names or physical characteristics of nearby nickname owners.

An identity inference happens when an inferrer is able to correctly map a nickname shown on his/her phone to a nearby person. For example, the inferrer's cell phone repeatedly shows that "Superman" is nearby and the inferrer repeatedly sees the same person among all nearby people. That person must then be "Superman".

4.2. Step 2- Generating a Classification and Regression Tree

The data collected from the field study included many features (alternatively called independent variables or internal nodes) for each instance of a pop-up questionnaire for each subject, including:

- Place-related variables such as type of current place and the frequency of visiting this place.
- Proximity-related variables such as the number of nearby people, number of nearby same application users, and number of nearby friends.
- Time-related variables such as the number of pop-up questionnaires completed by the subject up to the time of this questionnaire and the number of days passed since the beginning of the study.
- Demographic variables for this subject and the nearby users such as their education level, gender, and ethnicity. This information is derived by the system using data obtained in phase I.
- Finally, the instantaneous and historical inference functions were evaluated for each nearby user.

Note that not all the above features are necessarily relevant, but like any other regression analysis (including many intrusion detection systems) we monitor and try to use all of the available system features. Some of the features may be redundant or may contribute little (if anything) to the detection process [13]. Hopefully the decision tree will help us clear up any confusion. Over 3000 questionnaires were answered in our study. The subjects' answers and their elaborations on their guesses were then compared to the demographic and physical information collected in the pre-study survey in order to verify if the subject was able to correctly identify a nearby nickname or narrow it down to just two users (where one of them is the correct identity). Correct guesses were classified as *high risk* situations while the others were classified as *normal* situations. This binary decision process is driven by the tree contents.

The classification trees for the subjects were generated automatically using MatLab. Gini's diversity index [30] was used to choose an outgoing tree branch. For reliability purposes, nodes had to have 100 or more observations to be split.

Three sets of classification trees were generated. The first set involved trees trained and tested using the inference functions as independent variables. Trees in the second set were trained and tested with all other categories of independent variables discussed above (i.e. excluding our inference functions). Finally, trees in the third set were trained and tested with all five categories of independent variables explained above (i.e. by incorporating all the variables used in the first two tree sets). The procedure is shown in Fig. 1. The variable denoting a correct_guess was always used as the dependent variable. Classification results are explained in the next section.

5. Classification Results

We produced two instantaneous inference functions: *inst_inf_1* is the value of the instantaneous inference function ($INF_1(Q \rightarrow \Phi)$) in equation **(4)** where the number of possible values for a nearby user's identity, V, is set to the number of nearby users using the application. *inst_inf_2* is the value of the instantaneous inference function where the number of possible values for a nearby

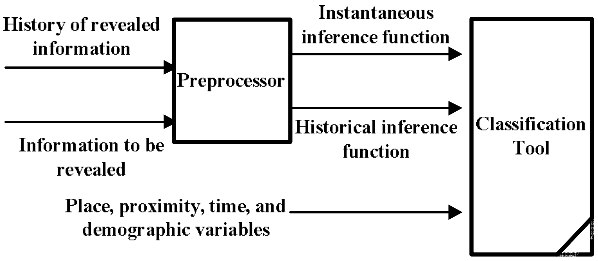
History of revealed information
Information to be revealed
Preprocessor
Instantaneous inference function
Historical inference function
Place, proximity, time, and demographic variables
Classification Tool

Fig 1. Block-diagram for classification process (with all possible inputs)

user's identity, V, is set to the number of all nearby people. The value of the historical inference function, *hist_inf*, was calculated considering the history of co-proximity of the subject and the nearby user, for up to two weeks before the current questionnaire popped up.

First, only inference functions were used as independent variables. The tree structure and the rate of correct classification changes based on the ratio of costs[1] between false positives[2], C_P, and false negatives[3], C_n. We changed the cost of false negatives, C_n as compared to the cost of false positives, C_P, and obtained the upper curve depicted in Fig. 3. As shown in the figure, for C_n=8*C_P the true positive rate is 85% and the true negative rate is 74%. Since correct guesses were made rarely with the questionnaires (about 12%), the false negative must be given a higher cost to produce a large true positive rate. An instance of the tree for C_n=6.*C_P is shown in Fig. 2. This tree uses the inference functions both individually and in combination. The tree basically implies that a situation is of high risk when either the instantaneous or the historical inference is too high (hist_inf>threshold T1 or inst_inf>threshold T2), or they are both relatively high (hist_inf> T4 or inst_inf>T2 where T4<T1).

In the second phase, only the time-, place-, and proximity-related information and demographic features explained in Section 4 were used as independent variables. Note that the proximity-related features include the number of nearby application users and the number of nearby people; the latter implies the number of possible values for a user's identity, V, in calculating the instantaneous inference functions. However, no feature directly measures the historical inference function. The correct classification rate of the decision tree as a function

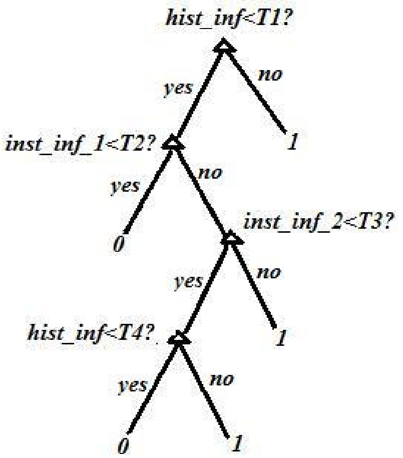

Fig. 2. A classification tree for C_n=6*C_P (phase 1).

of C_n is shown in the lower curve of Fig. 3. For a given true negative rate, the true positive rate is on average 30% lower than the true positive rate in the previous phase. An instance of the tree for C_n=6*C_P is shown in Fig. 4. It has a higher depth than the tree obtained in phase one. In the final phase, all five categories of variables involved in the first two phases were used as independent variables. The difference in the correct classification rate was less than 0.5%.

6. Analysis of the Results and Conclusions

We provided an overview of the social inference problem and presented our method of estimating the inference risk for mobile and social applications. Our method introduces the instantaneous and historical information entropy functions. Social inferences result from low information entropy. In the case of an identity inference, the user at

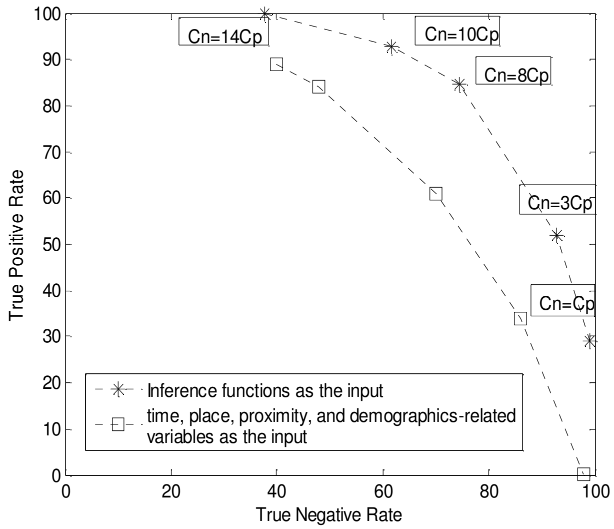

Fig. 3. Correct/true classification rates for various ratios of false-positive and false-negative costs.

[1] *T*he cost of correctly classifying a situation into a specific class should be minimum. By default in MatLab, this cost is 1 if the classification is wrong and zero otherwise.

[2] For a false positive, no inference occurs but the tree classifies the situation as high risk.

[3] For a false negative, an identity inference occurs but the tree classifier identifies the situation as normal.

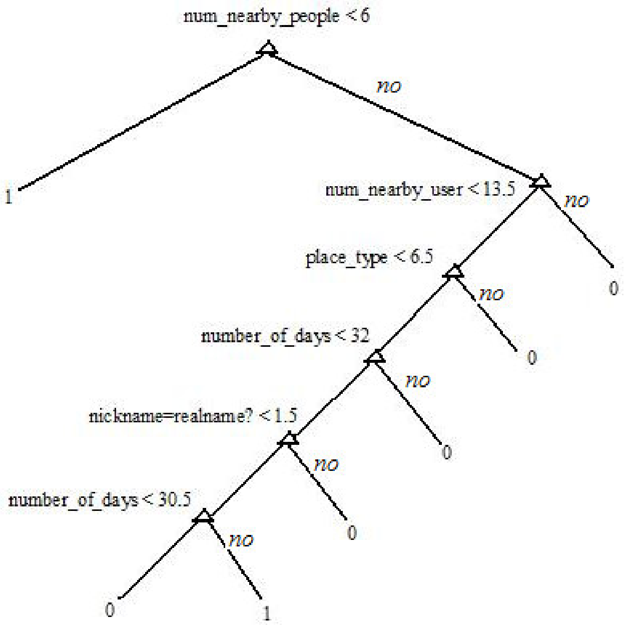

Fig. 4. A classification tree for C_n=6*C_P (phase two).

risk is uniquely identified by the inferrer. Therefore, when a classification tool is used for high risk situations, the tool either needs an input variable that measures uniqueness or an appropriate set of input variables from which low information entropy can be detected. The second case can be very difficult or even impossible to obtain for the following reasons.

- Lack of adequate training data involving unique situations and outliers. After a user, ***A***, reveals a subset of attributes, say $\{e_1, e_2..., e_n\}$, the information entropy is a function of the number of people for whom $e´_1=e_1$ and $e´_2=e_2$... and $e´_n=e_n$, and the respective probabilities considering all these users. To capture whether the entropy falls below the preset threshold, the classification tool must have enough training data to include unique combinations of user attributes which are unlikely to happen. For example, it is highly unlikely that the {Hispanic, female, soccer player} set (in the example of Section 3) appears exactly as is in the training data.
- Difficulty of modeling the *history* of information that a potential inferrer collects about another user ***B*** in relation to all the other users. E.g., in the proximity-based application of this paper, the proximity of the inferrer to ***B*** in three different situations should not lead to an identity inference if many common users are involved in these situations. However, it can lead to an identity inference if user commonality is small among these situations. The only strong estimator of this risk is the historical inference function which is not necessarily correlated with any observable feature.

To choose the best features for classifying the situations and investigating the effectiveness of the decision algorithm, we then used a classification tree to identify high risk and normal situations. We observed the following results.

- Using the inference functions as internal nodes produces a rather simple and comprehensive tree structure.
- Feeding the tree with all the features that can be directly measured (place, proximity, time and demographic variables) in addition to the inference functions does not make a significant improvement in the classification rate.
- Providing all these features without the inference functions does not yield the same performance as in the former two cases.

Most intrusion detection systems monitor and use all of the available system features. Some of the features may be redundant or may contribute little (if anything) to the detection process [13]. This results in excessive computational complexity with little gain. We have shown that including more features in addition to our inference functions does not actually improve the true classification rate. Also, using all the available features without entropy calculations increases the complexity of the tree without achieving comparable success. Therefore, we obtained the simplest and most accurate design with a preprocessor that calculates the inference functions and a classification tree having as inputs the instantaneous and historical inference functions. The tree uses these functions separately and collectively to appropriately classify the situations.

7. Acknowledgements

This research is partially supported by the National Science Foundation Grant NSF IIS DST 0534520 and NSF CNS 0454081. The opinions expressed are those of the authors and may not reflect those of the NSF.

8. References

[1] W. Stallings, *Cryptography and Network Security Principles and Practices*: Pearson Prentice Hall.

[2] D. Hong, M. Yuan, and V. Y. Shen, "Dynamic Privacy Management: a Plugin Service for the Middleware in Pervasive Computing," in *ACM 7th international conference on Human computer interaction with mobile devices & services* 2005, pp. 1-8.

[3] A. Gal and V. Atluri, " An Authorization Model for Temporal Data " in *ACM Conference on Computer and Communication Security*, 2000.

[4] M. Langheinrich, "Privacy by Design – Principles of Privacy-Aware Ubiquitous Systems," in *Third*

International Conference on Ubiquitous Computing (UbiComp 2001). , 2001, pp. 273-291.

[5] S. Motahari, C. Manikopoulos, R. Hiltz, and Q. Jones, "Seven privacy worries in ubiquitous social computing," in *ACM International Conference Proceeding Series; Proceedings of the 3rd symposium on Usable privacy and security* 2007, pp. 171-172.

[6] J. Cornwell, I. Fette, G. Hsieh, M. Prabaker, J. Rao, K. Tang, K. Vaniea, L. Bauer, L. Cranor, J. Hong, B. McLaren, M. Reiter, and N. Sadeh, " User-controllable security and privacy for pervasive computing.," in *Proceedings of the 8th IEEE Workshop on Mobile Computing Systems & Applications*, 2007.

[7] A. Brodsky, C. Farkas, and S. Jajodia, "Secure databases: constraints, inference channels, and monitoring disclosures," *IEEE Transactions on Knowledge and Data Engineering,* vol. 2, pp. 900-919, 2000.

[8] D. E. O'Leary, "Some Privacy Issues in Knowledge Discovery: The OECD Personal Privacy Guidelines," *IEEE Expert: Intelligent Systems and Their Applications* vol. 10, pp. 48-52, 1995.

[9] P. K. Harmer, P. D. Williams, G. H. Gunsch, and G. B. Lamont, "An Artificial Immune System Architecture for Computer Security Applications," *IEEE Transactions on Evolutionary Computation,* vol. 6, 2002.

[10] M. Botha, R. V. Solms, K. Perry, E. Loubser, and G. Yamoyany, "The Utilization of Artificial Intelligence in a Hybrid Intrusion Detection System," in *ACM International Conference Proceeding Series*, 2002.

[11] S. Motahari, S. Ziavras, M. Naaman, M. Ismail, and Q. Jones, "Social Inference Risk Modeling in Mobile and Social Applications," in *IEEE International Conference on Information Privacy, Security, Risk and Trust.* 2009.

[12] S. Motahari, S. Ziavras, R. Schular, and Q. Jones, "Identity Inference as a Privacy Risk in Computer-Mediated Communication," in *IEEE Hawaii International Conference on System Sciences (HICSS-42)*, 2008.

[13] S. Chebrolua, A. Abrahama, and J. P. Thomas, "Feature deduction and ensemble design of intrusion detection systems," *Computers & Security,* vol. 24, pp. 295-307, 2005.

[14] L. Cranor, M. Langheinrich, M. Marchiori, and J. Reagle, "The Platform for Privacy Preferences 1.0 (P3P1.0) Specification," in *W3C recommendation*, 20020416, 2002.

[15] M. Ackerman and L. Cranor, "Privacy Critics: UI Components to Safeguard Users' Privacy," in *CHI 99*, 1999.

[16] S. Lederer, "Designing Disclosure: Interactive Personal Privacy at the Dawn of Ubiquitous Computing," in *Computer Science Division*: University of California at Berkeley, 2003.

[17] P. Osbakk and N. Ryan, "Context, CC/PP, and P3P," in *UbiComp 2002 Adjunct Proceedings*, Göteborg, Sweden, 2002.

[18] W. Stallings, *Cryptography and Network Security Principles and Practices.*: Pearson Prentice Hall., 1999.

[19] T. F. Lunt, "Current Issues in Statistical Database Security," *IFIP Transactions, Results of the IFIP WG 11.3 Workshop on Database Security V: Status and Prospects* vol. A-6, pp. 381-385, 1991.

[20] S. Jajodia and C. Meadows, *Inference Problems in Multilevel Secure Database Management Systems*. Los Alamitos, California, USA IEEE Computer Society Press, 1995.

[21] P. D. Stachour and B. Thuraisingham, "Design of LDV: A Multilevel Secure Relational Database Management," *IEEE Transactions on Knowledge and Data Engineering,* vol. 2, pp. 190-209, 1990.

[22] S. Dawson, S. D. Capitani, and d. V. P. Samarati, "Specification and Enforcement of Classification and Inference Constraints " *IEEE Symposium on Security and Privacy,* 1999.

[23] J. Zhan and S. Matwin, "A Crypto-Based Approach to Privacy-Preserving Collaborative Data Mining," in *Sixth IEEE International Conference on Data Mining Workshops*, 2006, pp. 546-550.

[24] A. Machanavajjhala, J. Gehrke, and D. Kifer, "ℓ-Diversity: Privacy Beyond k-Anonymity," in *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE 2006)*, 2006.

[25] L. Sweeney, "Technical Report LIDAPWP4, Uniqueness of simple demographics in the U.S. population.," Laboratory for International Data Privacy, Carnegie Mellon University, Pittsburgh, PA 2000.

[26] C. E. Shannon, "Prediction and entropy of printed English," *The Bell System Technical Journal,* vol. 30, pp. 50-64, 1950.

[27] M. Morgenstern, "Security and Inference in Multilevel Database and Knowledge-Based Systems," in *International Conference on Management of Data archive, Proceedings of the 1987 ACM SIGMOD international conference on Management of data* 1987, pp. 357-373.

[28] D. E. Denning and M. Morgenstern, "Military database technology study: AI techniques for security and reliability," 1986.

[29] J. A. S. Joel M. Hektner, Csikszentmihalyi, Mihaly., *Experience Sampling Method*: Sage Publications, 2006.

[30] L. Breiman, J. Friedman, R. Olshen, and C. Stone, *Classification and Regression Trees*. Boca Raton: Chapman & Hall, 1993.