Towards Learning Privacy Policies

Arosha K Bandara
Department of Computing
The Open University, MK7 6AA, UK
a.k.bandara@open.ac.uk

Alessandra Russo & Emil C Lupu
Department of Computing
Imperial College London, SW7 2AZ, UK
{a.russo, e.c.lupu}@imperial.ac.uk

With the proliferation of personal computing devices users are creating a variety of digitized personal information, from personal contact databases and multimedia content to context data such as location, activity and mood. Preventing unintended disclosure of such information is a key motivator for developing privacy management frameworks. It is equally critical that protecting privacy does not prevent users from completing essential tasks. For example, whilst a user does not generally make his location available, he *may* want to disclose it to the taxi company with which he has booked a journey. It is impractical for the user to always specify his preferences in advance and it is impossible to forecast every possible scenario. Furthermore, conflicts may arise due to apparently conflicting requirements or due to the diversity of situations encountered. Therefore there is a need to be able to learn privacy requirements from the user's behaviour and decisions and to be able to analyse privacy requirements for consistency. Otherwise the overhead of using the privacy management system could cause it to be disabled completely. These issues are illustrated in the following scenario:

"Alice and Bob's son Charles is involved in many after-school activities. Concerned for his safety whilst travelling to and from these activities, Charles' parents buy him a new mobile phone that has a GPS tracking feature together with a Privacy Manager (PM) tool. To prevent Charles from unintentionally disclosing is location to others, Bob configures the PM with a policy that states that only Alice and Bob can read Charles' location information.
One day Charles needs a lift home and uses a taxi firm, 'zCar', that allows customers to send SMS requests containing their location. However, when Charles tries to send a pick-up request, his PM informs him that this would violate his location privacy policy. Charles chooses to override his policy and soon a taxi arrives to take him home. The next time Charles needs a lift, he uses another firm offering the same service,'qCab', and is again forced to override his policy. Over time, Charles' PM learns this behaviour and suggests a new policy that will disclose his location to taxi firms whenever he requests a pick-up."

Current efforts in privacy management have focussed on notations for privacy policy specification (e.g. EPAL [1] and P3P [2]) and on user interaction design for privacy management [3]. However, little has been done to support automated analysis and learning of privacy policies. We advocate an approach based on Inductive Logic Programming (ILP) for automatic learning of privacy policies. ILP is preferred over statistical learning techniques because it produces rules (privacy policies) which are comprehensible to the user and amenable to automated analysis.

Our approach uses a simple formal representation of privacy policies together with a representation of the privacy decisions made automatically by the system or manually by the user. For example the predicate $policy(allow, zCar, read, charles, loc, pickup)$ specifies that $zCar$ was allowed to read Charles' location for the purpose of a pick-up. The scenario only has positive examples, E^+ of privacy decisions, but we can encode negative examples, E^- if needed. Our formalism also includes predicates that encode entity types, e.g. $is\_a(zCar, taxiFirm)$ specifies that $zCar$ is a taxi firm. With this information, we use ILP to perform observation-based predicate learning and infer rules, that entail E^+ but not E^-, having $policy(\ldots)$ predicates at the head and $is\_a(\ldots)$ predicates in the body.

We validate our approach by using $Progol$ to demonstrate the learning functionality in the above scenario. Future work will focus on more complex privacy policy scenarios that use non-observable predicate learning [4] and other ILP procedures (e.g. *HAIL*).

References

[1] P. Ashley, S. Hada, G. Karjoth, C. Powers, and M. Schunter. *EPAL 1.1*. URL: http://tinyurl.com/35xpon,(1.10. 2003).
[2] L. Cranor. *Web Privacy with P3P*. O'Reilly, USA, 2002.
[3] J. Karat, C. Karat, C. Brodie, and J. Feng. Privacy in information technology: Designing to enable privacy policy management in organizations. *IJHCS*, 63(1-2):153–174, 2005.
[4] S. Muggleton and C. Bryant. Theory completion using inverse entailment. In *10th Int. Conf. on Inductive Logic Programming*, pages 130–146, 2000.