

Privacy-preserving demographic filtering

E. Aïmeur G. Brassard
Dép. d'informatique et de R.O.
Université de Montréal
C.P. 6128, Succ. Centre-Ville
Montréal (Québec)
H3C 3J7 Canada
aimeur@iro.umontreal.ca
brassard@iro.umontreal.ca

J. M. Fernandez
Dép. de génie informatique
École Polytechnique
de Montréal
C.P. 6079, Succ. Centre-Ville
Montréal (Québec)
H3C 3A7 Canada
jose.fernandez@polymtl.ca

F. S. Mani Onana
Dép. d'informatique et de R.O.
Université de Montréal
C.P. 6128, Succ. Centre-Ville
Montréal (Québec)
H3C 3J7 Canada
manionaf@iro.umontreal.ca

ABSTRACT

The use of recommender systems in e-commerce to guide customer choices presents a privacy protection problem that is two-fold. We seek to protect the privacy interests of customers by trying to keep private their identity and demographic characteristics, and possibly also their buying preferences and behaviour. This can be desirable even if anonymity is used. Furthermore, we want to protect the commercial interests of the e-commerce service providers by allowing them to make recommendations as accurate as possible, without unnecessarily revealing valuable information they have legitimately accumulated, such as market trends, to third parties.

In this paper, we concentrate on recommender systems based on *demographic filtering*, which make recommendations based on feedback of previous users of similar demographic characteristics (such as age, sex, level of education, wealth, geographical location, etc.). We propose a system called ALAMBIC, which adequately achieves the above privacy-protection objectives in this kind of recommender systems. Our system is based on a semi-trusted third party in which the users need only have limited confidence. A main originality of our approach is to split user data between that party and the service provider in such a way that neither can derive sensitive information from their share alone.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*Information filtering*; H.3.5 [Information Storage and Retrieval]: Online Information Services—*Web-based services*; H.4.0 [Information Systems Applications]: General

General Terms

Security, Architecture

Keywords

E-Commerce, recommender system, privacy protection, demographic filtering, clustering, semi-trusted third party.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SAC'06, April 23-27, 2006, Dijon, France.

Copyright 2006 ACM 1-59593-108-2/06/0004 ...\$5.00.

1. INTRODUCTION

In e-commerce, recommender systems allow entities providing goods or services to guide the choices made by the users of such systems, i.e. potential customers. The recommendation can be issued by the merchant of the goods or services themselves or by intermediary brokers. In both cases, recommendations are issued from within a *catalogue* of items available through that merchant or broker. We use forthwith the generic term *Service Provider* (SP) to refer to the entity operating the recommender system and the owner of the respective catalogue. We use the generic term *service* for any item in this catalogue, whether it is a digital or physical good or a service to be provided.

Because of the size and complexity of the catalogue, recommender systems usually employ filtering techniques, such as content-based, collaborative, demographic or knowledge-based filtering to extract a reasonably small set of services to be recommended to the user. More detail about these filtering techniques can be found in [5, 25].

One important disadvantage of recommender systems, however, is that in order to obtain accurate recommendations the user might have to reveal to the SP information that is considered private, such as identity, demographic characteristics, previous buying behaviour, etc. On the other hand, the contents of the catalogue has intrinsic commercial value, and the SP should also try to protect it from competitors, and in particular from those who might be masquerading as potential customers and users of the recommender system. Thus, the problem of privacy protection in the use of recommender systems applies to both users and service providers.

In this paper, we address this problem in the context of recommender systems using *demographic filtering* (DF) techniques. In Section 2, we describe DF-based recommender systems and the privacy protection issues that they raise. We then introduce in Section 3 some of the privacy protection solutions introduced in other contexts and describe the required cryptographic primitives. We present in Section 4 the ALAMBIC system, our solution to protect privacy of both users and service providers. A discussion in Section 5 precedes the conclusion and future directions in Section 6.

2. PRIVACY AND DEMOGRAPHIC FILTERING

Recommender systems based on demographic filtering aim at categorising users based on their demographic information and recommend services accordingly. More precisely, demographic information is used to identify the types of users that like similar services. DF can be used by any SP who offers services by

using data on individual users. The key element of DF is that it creates categories of users having similar demographic characteristics, and tracks the aggregate buying behaviour or preferences of users within these categories. Recommendations for a new user are issued by first finding to which category he belongs and then by applying the aggregate buying preferences of previous users in that category.

Example - Part 1. *Today, a concerned citizen wanting to keep himself abreast and well informed on world affairs and politics has a myriad of sources of information from which to choose. Even if we restrict him to Internet sources, there are hundreds of sources ranging from organised Media, web sites of political parties and lobbies, interest groups, ad hoc posting in newsgroups, and even blogging and podcasting. Beyond the sheer number of such sources of information on a controversial topic, a user might have preferences on the source to be used, such as length and depth, political inclination, objectivity, means of delivery, etc. We will thus consider for our example a hypothetical search engine on Web items associated with world affairs and politics. Because it is reasonable to think that user preferences will be influenced by demographic characteristics, we will assume that this search engine uses a recommender system based on demographic filtering to rank the hits returned to a particular user.*

While there are many categorisation techniques that can be successfully used in this context, in this paper we shall use *Data Clustering* to illustrate these techniques and the proposed privacy-protection solutions introduced in Section 4. Note however that these solutions are completely independent of the categorisation technique used in a DF-based recommender system. Clustering has been the subject of much research, particularly in Statistics and Machine Learning (see for example [15]). It aims at forming clusters of similar objects. Objects in a given cluster are similar to each other, but different to objects in another cluster. A clustering algorithm thus needs a metric to compute the distance between two objects to indicate how similar or different they are. Therefore, objects are typically represented as vectors in a multi-dimensional space and distances are typically calculated using the Euclidean distance (or another instance of the Minkowski distance) considering all or a subset of the dimensions. Each cluster is represented by a centroid or a medoid¹, and sometimes radius and density information.

In demographic filtering, clustering is used to create the user categories mentioned above by considering the set of all previous customers. The objects are the users, and each dimension of the space represents one of their relevant demographic characteristics. For a given cluster C , its density represents the number of users in it and its radius is a measure of how demographically dissimilar they are. Then, the historical data on buying behaviour or preferences of each user in C is used to associate to the cluster C an *aggregate* buying behaviour. In its simplest form, this aggregate can simply consist of the list of services $P_C = \{p_1, p_2, \dots, p_c\}$ that were purchased or for which positive feedback was given by users in C . When a new user requires a recommendation, the recommender system computes the cluster to which he is closest, say C , and then produces as a recommendation the list of items P_C . In other words, DF-based recommender systems use *people-to-people* correlations to produce their recommendations. In this paper, we refer to the list of clusters, and for each cluster C the list P_C of items purchased/liked by users in C , as the *essence* of such a recommender

system, since constructing and maintaining (i.e. “distilling”) this information is all that is required in order to be able to make this kind of recommendations.

Of course, for such clusters to be used effectively in recommender systems, the aggregate buying preferences within a cluster must also show sufficient similarity. In our example, this means that the list P_C must be sufficiently small. Achieving this objective can be done in a variety of ways, such as merging or separating existing clusters, considering similarity distances on the combined space of demographic characteristics and buying preferences/behaviour, or by attempting to apply clustering techniques on the space of services. See [1] for instance. However, the specific way in which clusters are formed is immaterial to our privacy issues.

Example - Part 2. *Our concerned citizen happens to be a young male engineering student living in a Northern European country, of Arab origin, who likes soccer, drives a yellow car, and has indicated business and computer games as interests. The recommender has identified him to be in a cluster of well-to-do and educated young people in developed countries. Historically, people in this cluster have mostly clicked on links going to specialised in-depth print magazines, a certain subset of prolific bloggers, and multimedia clips from state-operated Western European broadcasters. When he queries the search engine for items about the war in Iraq in the last week, he will be provided (in order) with an in-depth article in *The Economist*, two posts by bloggers on the latest bombing and an audio clip from *Deutsche Welle*. Of course, there will be other hits on that topic, but they will appear after in the ordered list of matches.*

Unfortunately, in order for the clusters to be constructed and for a new user to be associated with the appropriate cluster, users must provide the SP with demographic information. The potential abuses from unscrupulous SPs are obvious. For example, a SP could pool its information with other SPs and/or governments. It could also sell this information. This could result in a serious violation of the user’s privacy. Such violations are prohibited by many governments but effective methods to enforce privacy laws are generally lacking. This problem is exacerbated when information is used about individuals without their knowledge of it. Should the user ever have the proof that his privacy has been violated by the SP, he could complain to the proper authorities, so that justice might be served. However, the complaint alone is not sufficient to restore the user’s privacy.

On the other hand, the privacy of data legitimately accumulated by the SP must also be protected. In a competitive market, pricing information contained in the catalogue might be sensitive and the SP might not want to reveal it to competitors. Furthermore, depending on the type of service, the catalogue itself might have value, such as in the case of information brokers. While the SP might be willing to reveal a few items and prices to a *bona fide* user, he must take precautions so that the information revealed cannot be used by competitors posing as users to obtain information on the catalogue or the pricing that they should not have.

Most importantly, however, once the SP has operated for a sufficiently long time that the essence distilled is good enough for use in a recommender system, the essence itself becomes of even higher value than the catalogue and should hence be protected, with even more vehemence, from potential competitors.

Technically, the essence belongs to the SP. It is in his best interest that it be updated accurately, because the quality of recommendations depends on it, and therefore so will the user’s satisfaction, and hence the SP’s business revenues. However, even though it is only a statistical aggregate, the essence should not be completely “opened” to the SP to prevent him from using “differential anal-

¹The centroid is a virtual point corresponding to the average of all the points in the cluster, while the medoid is the median point of the cluster.

ysis” on consecutive versions of the essence in order to infer the user’s demographic profile.

Our task is to provide an architecture for such generic demographic filtering systems, in which the privacy requirements outlined above are achieved. For that purpose, we identify two classes of privacy level:

1. Soft privacy: The user wants to keep his identity and demographic profile secret at all cost, but allows the SP to know which services he is interested in.

2. Hard privacy: The user wants to keep his identity and demographic profile secret at all cost, and furthermore, he does not allow the SP to know which services he is interested in.

Example - Part 3. Our young student might not want it known that he speaks Arabic. However, that might be relevant for the recommender to suggest items such as Al-Jazeera. Furthermore, in the hard-privacy case, he might not even want the SP, or anyone, to know that he is looking for information on recent events in Iraq. On the other hand, consider that the SP’s business model is to charge a certain amount for the number of hits returned, e.g. for 1 euro, you can purchase 25 queries that will return up to 3 hits each. Additional hits are available at additional cost. To maintain customer satisfaction, it is clear that the SP must use a recommender system to tailor which 3 hits to return to the user. Furthermore, he does not want to let the user browse all of them, because that would result in a loss of profit (operating a good search engine does cost money, after all). Finally, knowing which hits to provide to which users (the essence) is the information on which the whole business model is based. Losing that information to another search-engine provider would mean losing a significant competitive advantage.

3. RELATED WORK AND CRYPTOGRAPHIC PRELIMINARIES

In this section, we review some work related to the privacy-preservation paradigm. We also review the concepts of Secure Two-Party Computation (STPC) as well as Code Encryption and Code Obfuscation. We assume that the concepts of Public Key (or Asymmetric) Cryptosystems, Secret Key (or Symmetric) Cryptosystems and Public Key Infrastructure are well known.

3.1 Privacy Preservation

Much research has been done to address the problem of privacy preservation in Data Mining. In particular, several approaches have been classified using five dimensions [24]. *First*, data could be centralised or distributed. The *second* dimension is about the modification that can be applied on data, such as perturbation, aggregation, swapping, sampling, etc. The *third* dimension is concerned with algorithms that are used in data mining (for instance decision tree, association rule, clustering, rough sets, Bayesian networks). The *fourth* dimension considers whether it is the raw data or the aggregate data that must be hidden. Finally, in the *fifth* dimension, it is important to highlight how the privacy-preservation techniques are used for the modification of data. Here, techniques are heuristic-based, cryptography-based or reconstruction-based.

There have also been several approaches developed for privacy-preserving clustering. For example, one approach aims at using transformations to perturb the dataset before the algorithm is applied (see for example [20]). Another approach consists in designing algorithms that use secure multi-party computation (see for example [16]). In this paper, the clustering process is executed by the Alambic agent through the feedback it receives from users. Detail is given in Section 4.

Another example of domain with interest to researchers in privacy preservation is the Collaborative Filtering (CF) technique, which outputs recommendations of services for a given user, based upon the behaviour and the evaluations of the other users [22]. A scheme for privacy-preserving collaborative filtering is proposed in [7]. Here, a community of users can compute a public “aggregate” of their private data without revealing them. Each user takes part in the construction of the aggregate and gets personalised recommendations by using local computation. The problem with this scheme is that it is *on-line*, meaning that several users need to participate simultaneously in each recommendation process. Another scheme was proposed in [21], in which the user disturbs his demographic data, using *Randomised Perturbation* techniques, before sending them to the SP. In this way, the SP cannot compile truthful demographic information about the user. Although the data are disturbed, Polat and Du argue that their scheme should still allow for successful collaborative filtering. Our subsequent work on the ALAMBIC architecture offers another solution to the privacy-preserving collaborative filtering conundrum, in which we do not require several customers to be online simultaneously nor do we need to introduce random perturbations [2].

3.2 Secure Two-Party Computation

Secure Two-Party Computation (STPC) is concerned with the problem of evaluating a function $f(x, y)$ for which the first party, Alice, provides secret input x and the second party, Bob, provides secret input y , such that the output becomes known to both parties while the inputs x and y remain secret from Bob and Alice, respectively, except for what can be logically inferred from one’s private input and the joint output. The first general STPC protocol was given by Yao [26]. By assuming the intractability of factoring, he showed that every two-party interactive computational problem has a private protocol. Other solutions followed later [27; 17, etc]. Despite recent efforts to implement generic protocols for STCP [19], it remains most likely that efficient implementations will rely on simplifications based on the particular structure of the function to evaluate. In this paper, the SP and the user, assisted by the Alambic agent, are invited to execute a STPC if, in addition to having no information about the user demographic profile, the SP is not allowed to know the services of interest to the user, i.e. in the hard-privacy case (see Section 2).

3.3 Code Encryption and Code Obfuscation

Code encryption is a technique used to protect mobile code that is executed on remote and possibly untrusted computers. The security problems related to mobile code in general, and mobile software agents in particular, has been studied in [23]. For example, a mobile agent must be able to protect the integrity and the execution of its code, compute with secrets in public, and preserve the privacy of the code and internal data. Sander and Tschudin also introduced the notion of *Computing with Encrypted Functions*.

Code obfuscation [12] is a process that transforms a program so that it becomes more difficult to understand and more resistant to reverse engineering. The code resulting from this process is called *obfuscated code*. In practice, obfuscated code can be the result of one or more code transformations. Even though it differs significantly from the original code, the obfuscated code must produce the same result as the original despite a possible slow down.

In the solution proposed in this paper, the code of any given Alambic agent is encrypted and obfuscated in such a way that the SP is unable to gather information on its internal state or variables or on how it works. This prevents the SP from being able to alter

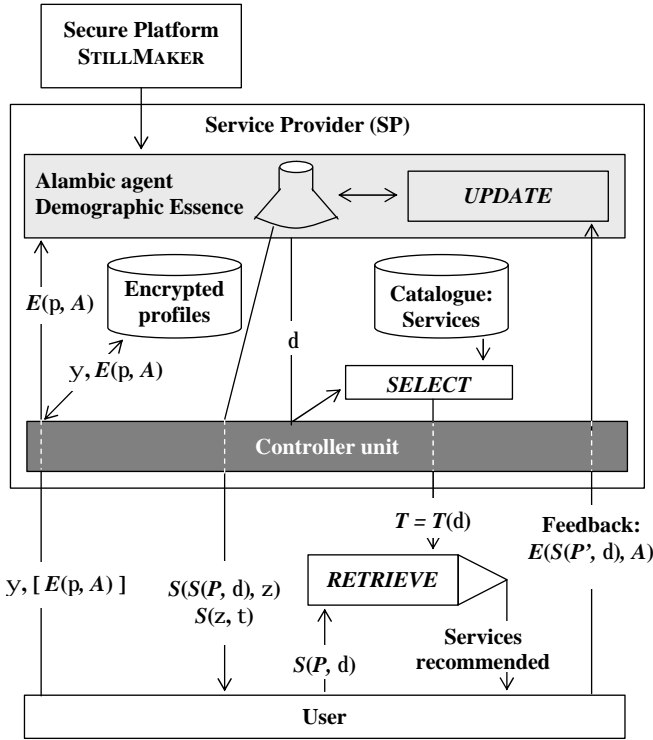


Figure 1: Architecture of ALAMBIC

the behaviour of the Alambic agent in a way that would reveal sensitive information.

4. THE ALAMBIC SYSTEM

In this section, we present ALAMBIC, our privacy-preserving demographic filtering system. First, we present its architecture. We then describe its main ingredients and detail the demographic filtering process. Finally, we show how the user’s privacy is preserved.

4.1 Architecture and components

The architecture of ALAMBIC is presented in Figure 1. We present in this section the three main components of the ALAMBIC system as well as the semi-trusted platform that generates the Alambic agents.

The SP: He maintains his catalogue and the encrypted user profile database. The catalogue contains the available services. The SP communicates with the Alambic agent and the user through a Controller Unit.

The Alambic agent: It resides inside the SP’s system, but its code is encrypted and obfuscated so that the SP cannot obtain more information than what can be deduced from the output of the *RETRIEVE* process, described in Section 4.2. The Alambic agent can be a piece of software that the SP downloads from the STILLMAKER (see below) and installs in his system. It can also be a STILLMAKER’s provided secure co-processor that the service provider adds to his system.

The user: He wishes to receive recommended services from the SP. He receives the Alambic agent’s public key from the SP to encipher his private information (demographic data), but this public key bears the STILLMAKER’s digital signature.

The STILLMAKER: The STILLMAKER is a Secure Platform that generates Alambic agents for SPs. Each Alambic agent is issued a separate private/public key pair and a public key certificate that is signed by the STILLMAKER by using a Public Key Infrastructure (PKI) with digital signature services that both the SP and the user recognize. The authentication of a given Alambic agent is equivalent to that of its public key. The user encrypts the data part of his profile with this public key. To give users confidence that the Alambic agents ensure their privacy, the STILLMAKER must use fourth-party validation, meaning that some independent organizations are invited to test and validate the generation of the Alambic agents by the STILLMAKER. In particular, the tests and validation must confirm that the obfuscation and encryption are performed correctly and that the Alambic agent’s private key is well hidden from the SP.

4.2 Description and main concepts

ALAMBIC is a demographic filtering system that provides a solution that achieves the privacy requirements described in Section 2. This solution is based on the following *division of trust principle*:

“Trust no one, but you may trust two”.

In other words, users will distribute their trust between the SP and the *Alambic agent*, in such a way that only a collusion between them would expose the user’s private data. Each SP has its own Alambic agent, which is provided by a STILLMAKER. The partial trust required of the agent (and therefore of the STILLMAKER) is not far fetched, as discussed in Section 5.

The main idea behind our solution thus resides in the introduction of the Alambic agent, which is a *semi-trusted* third party in which users only need to have limited confidence. The demographic essence described in Section 2 is handled by the Alambic agent because it requires knowledge of demographic information in order to make recommendations. This essence provides a clustering of users according to their demographic information, together with a mapping from these clusters to the preferences (represented in this paper by anonymous catalogue item indexes p_i) normally shown by these users.

The Alambic agent serves as an intermediary between the SP and the user. It makes recommendations based on information in the demographic essence. It also maintains it by receiving feedback from the user and updating the clusters and their mappings to the anonymous indexes accordingly.

We are now ready to describe ALAMBIC’s demographic filtering process.

4.3 The demographic filtering process

The execution process of ALAMBIC is shown through the interactions materialised by arrows in our architecture (Figure 1). These interactions are considered from left to right as time goes by. The steps of the execution process follow.

1. The SP provides the user with the public key certificate of the Alambic agent. This allows the user to learn the Alambic agent public key and make sure it is legitimate.
2. The user enciphers his demographic profile, π , with the Alambic agent public key, A , and obtains $E(\pi, A)$. For the sake of confidentiality, π contains in addition a secret key τ , used in Step 5 below. The user sends his pseudonym, ψ , together with $E(\pi, A)$, to the SP.
3. The SP inserts ψ and $E(\pi, A)$ in its encrypted profile database, and forwards $E(\pi, A)$ to the Alambic agent. [From

Table 1: The Alambic agent clustering table

Centroid	Anonymous indexes
C_1	p_1, p_2, p_7
C_2	p_5, p_8, p_{20}, p_t
...	...
C_n	p_{10}

Table 2: Structure of the SP's catalogue

Anonymous Index	Service Index	Description	...
p_1	s_1	...	
p_2	s_2	...	
...	
p_t	s_t	...	

now on and for future connections, the user “identifies” himself as ψ when he communicates with the SP].²

- The Alambic agent receives $E(\pi, A)$ from the SP. It decipheres $E(\pi, A)$ using its private key, A' , and obtains the user's demographic profile: $\pi = D(E(\pi, A), A')$, where D is the decryption algorithm.
- From π , the Alambic agent computes the distance between the user's demographic profile and the centroid of each cluster, to find the nearest cluster, $C = C_j(\pi)$, for a certain $j \in \{1, \dots, n\}$, where n is the number of clusters in the demographic essence. The Alambic agent thus produces a list, $P_C = \{p_1, \dots, p_c\}$, of anonymous indexes that users in C have chosen in the past (see Table 1). It picks at random two secret keys δ and ζ . The former serves in protecting the SP indexation procedure from the competition. The latter protects the list of anonymous indexes which is sent to the user: this protection is particularly useful in the case of hard privacy. Let S be a symmetric encryption scheme publicly known by all the parties. The Alambic agent applies a double encryption, $S(S(p_i, \delta), \zeta)$, of each anonymous index p_i of the list P_C , $i \in \{1, \dots, c\}$. It thus obtains the list:

$$S(S(P, \delta), \zeta) := \{S(S(p_1, \delta), \zeta), \dots, S(S(p_c, \delta), \zeta)\}.$$

The Alambic agent uses the secret key τ it receives from the user (see Step 2) to encipher ζ as $S(\zeta, \tau)$. It sends both $S(\zeta, \tau)$ and $S(S(P, \delta), \zeta)$ to the user. Finally, it also sends δ to the SP.

- The SP computes $S(p_k, \delta)$ for each anonymous index p_k , $k \in \{1, \dots, t\}$, where t is the total number of anonymous indexes in his catalogue (see Table 2). He then executes the procedure *SELECT*, which creates a table T , containing values $S(p_k, \delta)$, $k \in \{1, \dots, t\}$, as search keys, and the associated service indexes and descriptions. In other words, this table T contains the same information as the initial catalogue but is indexed through anonymous indexes masked by δ .
- The user decipheres $S(\zeta, \tau)$ and obtains $\zeta = S^{-1}(S(\zeta, \tau), \tau)$. He then computes the set $S(P, \delta) = S^{-1}(S(S(P, \delta), \zeta), \zeta) = \{S(p_1, \delta), \dots, S(p_c, \delta)\}$.

²While the use of a pseudonym ψ might make the system more convivial by allowing the storage of personalised preferences and encrypted profiles, it is not strictly required in ALAMBIC. In fact their use might detract from protecting user privacy. Some user studies show that users do not like being tracked when buying online, even if anonymously. This issue is addressed in [2].

- The user and the SP execute the procedure *RETRIEVE* as follows.

- In the soft-privacy case, the user only has to send the list $S(P, \delta)$ to the SP who compares this list with its table T and obtains recommended services for the user. In other words, in this case, the procedure *RETRIEVE* corresponds to selecting services in T that have encrypted and anonymous indexes in $S(P, \delta)$.
- In the hard-privacy case, the procedure *RETRIEVE* can be implemented with specific STPC solutions adequate for this task, such as SPIR (Symmetrically Private Information Retrieval) [11, 18, 13, 8] or BliS (Blind Search) [3]. These solutions enable the user to receive recommended services associated to the encrypted and anonymous indexes in $S(P, \delta)$, while the SP remains ignorant of such indexes.

In either case, the procedure *RETRIEVE* outputs recommended services for the user.

- Let $S(P', \delta)$ be the set of encrypted and anonymous indexes of services that the user has purchased (implicit feedback) or for which he has indicated a preference (explicit feedback). The user sends $S(P', \delta)$ to the Alambic agent for update purposes. $S(P', \delta)$ is sent in clear in the case of soft privacy and in its encrypted form $E(S(P', \delta), A)$ in the case of hard privacy. The updating process is discussed below.

Updating the demographic essence:

The essence update is based on the list $S(P', \delta)$ that the Alambic agent receives as the user's feedback. From elements in this list, the following situations could arise:

- $S(P', \delta) = S(P, \delta)$: The user is satisfied by all the services associated to the encrypted and anonymous indexes that he has received from the Alambic agent.
- $S(P', \delta) \subset S(P, \delta)$: The user is partially satisfied by the services associated to the encrypted and anonymous indexes that he has received from the Alambic agent.
- $S(P', \delta) \supset S(P, \delta)$: The user is satisfied by all the services associated to the encrypted and anonymous indexes that he has received from the Alambic agent. Moreover, he is satisfied by additional services that he queried by other means: for example, advice from a friend suggesting him to get a particular service. If the SP has other mechanisms to recommend services, such as *item-to-item* correlations, this could potentially come out with additional services that are correlated to those resulting initially from $S(P, \delta)$. In this case, the SP sends the service's description together with its encrypted and anonymous index, which the user adds to the list $S(P', \delta)$.
- $S(P', \delta) \cap S(P, \delta) = \emptyset$ and $S(P', \delta) \neq \emptyset$: The user rejects all the services recommended, but he receives and is satisfied by services that he queried by other means (as in the previous case 3).

In all of the above cases, the necessary modifications might involve changes to the list of items P_C of the cluster C to whom the user belongs, but it also changes C itself (changes in the centroid or medoid, density and radius) and might even trigger the merging, separation or creation of new clusters. For our purposes, all that matters is that all of these modifications can be embedded in the code of the Alambic agent and only require as input the (possibly encrypted) list of items $S(P', \delta)$.

5. PRIVACY AND APPLICABILITY OF ALAMBIC

We now address the compliance of the ALAMBIC system with respect to the privacy model of Section 2. Let us consider first the user's data: identity and demographic profile. There is no real need for the identity of the user ever to be revealed to any of the parties (the SP or the Alambic agent). In fact, the identity of the user can be maintained secret, even if the service needs to be purchased. While ALAMBIC does not address directly this issue, it is important to note that several solutions have been proposed and exist for blind search [3] and anonymous payments [10]. Anonymous delivery is relatively easily achievable [9, 14, 6] with electronic goods (e.g. music, software) that can be delivered over the Internet. And even in the case of physical goods, privacy-preserving delivery systems have been proposed [4] to obviate the need for the user to identify himself to the SP and protect other potentially private information such as his address.

As for the demographic data, there are three possible threats to their privacy by the SP. First, a *crypto-analytic attack* in which the SP could try to decrypt the encoded information sent by the user (or stored by him). This would require either breaking the public-key cryptosystem or falsifying a public key certificate. Second, an *inference attack* whereby gaining access to consecutive "clear" snapshots of the essence would allow the SP to make strong inferences on the demographic profile of the last user. However, since the essence update phase concerns only the Alambic agent, which does not reveal the outcome of that computation, such an analysis is not possible. In order for the SP to perform such analysis, he would have to "break" the Alambic agent's code.

Finally, and this must not be overlooked, the SP's private data is also protected. The contents of the catalogue (its contents, but also its sensitive attributes such as price, availability, etc.) are protected from the Alambic agent by the use of anonymous indexes, which do not identify the particular service or service type that a category of users might prefer. These anonymous indexes are meaningless to the Alambic agent but are used by the SP to index his catalogue of services. Furthermore, the fact that the controller unit monitors and controls all communications of the Alambic agent with the outside, makes it very difficult for the Alambic agent to voluntarily leak information about the essence to an outside party (e.g. one posing as a user).

In summary, all private data involved in ALAMBIC are well protected under the following assumptions:

- The underlying symmetric and public-key cryptography is secure and is used within a well-implemented Public Key Infrastructure.
- The STPC protocols used are secure and properly used and implemented (in case of hard privacy).
- The Alambic agent's code and internal data cannot be read or tampered with by the service provider, at any phase of the protocol. This may be verified by an independent accreditation authority.
- The semi-trusted third party and the SP can be trusted not to collude with each other in order to obtain the full user's demographic profile (the division of trust principle).

This last assumption is a non-technical one and no mathematical proof of its soundness can be given. Since the privacy protection features of our system, and hence its viability, depends on it, it is paramount to discuss whether it makes sense. We put forth the thesis that it is indeed a sound assumption, as long as the parties have clear vested interests and that those motivate them to perform what is expected of them in an honest fashion.

This is true for the SP, because it is in his best interest to provide as accurate and relevant a recommendation to the users of

the system as he can. This will result in a higher user satisfaction, better return business and eventually higher revenues. Thus, he is willing and motivated to implement a recommender system and furthermore to commit the resources necessary to comply with constraints imposed by law or by the users, for example privacy protection. For this reason, the essence should remain his property, even though he cannot use it or interpret it by himself. In addition, the ability for service providers to maintain and accurately update an as-inclusive-as-possible essence will provide them with a significant competitive advantage.

We see the semi-trusted third party that generates Alambic agents, i.e. the operator of the STILLMAKER platform, as a service provider himself, whose business survival depends on his ability and diligence in protecting the privacy of the public. He provides this service to the SP for a price, but is accountable to the public at large, and thus to any potential user of the SP's recommender system, to protect their privacy. This is not a far-fetched notion: it is already the case for those bound by professional secret and other privacy laws. This includes, for example, the figure of a notary public or equivalently the attorney-at-law (or solicitor), but also, to a certain extent, governments and banks. The other obvious example, in this era of Internet and e-commerce, are the Commercial Certification Authorities (such as *Verisign*, *Entrust*, etc.) whose business largely depends on the public's confidence in their honesty and professionalism. While there might be an incentive for them to collude with the service provider, we believe that modern society has already "dealt" with this problem and provides enough mechanisms (legal framework, checks and balances, and other means of deterrence) to prevent these situations, which are deemed sufficient by and for most (law-abiding) members of the society. We believe that the use of ALAMBIC (or a system similar to it) would make it possible and practical to have such parties intervene in the e-commerce process to protect the privacy of the public.

Last but not least, another fundamental premise is that the *users themselves* also have a vested interest in the accuracy and relevance of recommendations made by the service provider. Without this condition, they will not be willing to put in the extra effort, albeit small, to provide accurate input into such a system, but more importantly they will not be willing to assume the small residual risk associated with providing their private information into the ALAMBIC system. Finally, there might be particular application domains (e.g. recommendations of medical or pharmacological products) in which privacy is of such high importance, that users would be willing to share the additional cost of constructing and operating such a system with the SP, for example by paying fees to the semi-trusted third party.

6. CONCLUSIONS AND FUTURE WORK

We have described here a solution that allows SPs and users to collaborate in providing the latter with accurate and relevant recommendations of services, without compromising user privacy or SP commercial interests. While other approaches have been proposed to protect privacy for other filtering techniques, such as collaborative filtering [7, 21] or generic clustering [16], ours is the first that addresses the particularly thorny privacy issues surrounding demographic filtering. Furthermore, in comparison to these approaches our solution has marked advantages: a) it does not require previous users to be present when new recommendations are made, b) it does not introduce error-inducing perturbations in the data, and c) it uses simpler to implement cryptographic primitives.

We have restricted ourselves to recommender systems using demographic filtering techniques to provide user-to-user correlations (e.g. through user clustering). The main reason we chose to

do this is because, from the user point of view, we believe that most privacy requirements would be satisfied if their demographic information were protected. Nevertheless, we assert that it is possible to generalise the ALAMBIC architecture to provide privacy protection in other classes of recommender systems such as Collaborative Filtering and Content-Based Filtering [5]. This generalisation of the current paper is indeed considered in its sequel [2].

We believe we have also made a strong case for the viability of such systems in the e-commerce setting, by showing how it behaves all parties to collaborate. In particular, we argue that the main beneficiary for providing accurate recommendation is the SP. A key element in achieving this aim is the upkeep of an accurate aggregate market picture, which we have designated as “essence”. The inherent and potentially enormous value of this essence immediately suggests issues that future research should address. For one, a mechanism should be conceived that allows the SP to access and interpret the essence from time to time, while still protecting the user’s privacy. Secondly, security mechanisms must be designed and integrated in the system to guarantee the availability and integrity of the essence against both fortuitous and deliberate threats.

7. REFERENCES

- [1] E. Aïmeur, G. Brassard, H. Dufort, and S. Gambs. CLARISSE: A machine learning tool to initialize student models. In *Proceedings of Sixth International Conference on Intelligent Tutoring Systems: ITS '02*, pages 718–728, Biarritz, France, June 2002.
- [2] E. Aïmeur, G. Brassard, J. M. Fernandez, and F. S. Mani Onana. ALAMBIC: A privacy-preserving recommender system for electronic commerce. Manuscript available from the authors, November 2005.
- [3] E. Aïmeur, G. Brassard, and F. S. Mani Onana. Blind sales in electronic commerce. In *Proceedings of the 6th International Conference on Electronic Commerce (ICEC'04)*, pages 148–157, Delft, The Netherlands, October 2004.
- [4] E. Aïmeur, G. Brassard, and F. S. Mani Onana. Privacy-preserving physical delivery in electronic commerce. In *Proceedings of IADIS International Conference on e-Commerce*, Porto, Portugal, December 2005.
- [5] R. Burke. Hybrid recommender systems: Survey and experiments. *Customer Modeling and Customer-Adapted Interaction*, 4(12):331–370, 2002.
- [6] J. Camenisch and A. Lysyanskaya. A formal treatment of onion routing. In *Advances in Cryptology: Proceedings of CRYPTO 2005*, pages 169–187, Santa Barbara, CA, August 2005.
- [7] J. Canny. Collaborative filtering with privacy via factor analysis. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 238–245, Tampere, Finland, August 2002.
- [8] Y.-C. Chang. Single database private information retrieval with logarithmic communication. Available at eprint.iacr.org/2004/036/, accessed 1 November 2005, 2004.
- [9] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, February 1981.
- [10] D. Chaum. Security without identification: Transaction systems to make Big Brother obsolete. *Communications of the ACM*, 28(10):1030–1044, October 1985.
- [11] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan. Private information retrieval. In *Proceedings of 36th Annual IEEE Symposium on Foundations of Computer Science*, pages 41–51, 1995.
- [12] C. Collberg, C. Thomborson, and D. Low. A taxonomy of obfuscating transformations. Technical report 148, Department of Computer Science, University of Auckland, 1997.
- [13] Y. Gertner, Y. Ishai, E. Kushilevitz, and T. Malkin. Protecting data privacy in private information retrieval schemes. In *Proceedings of the 30th Annual ACM Symposium on the Theory of Computing*, pages 151–160, 1998.
- [14] D. M. Goldschlag, M. G. Reed, and P. F. Syverson. Onion routing for anonymous and private internet connections. *Communications of the ACM*, 42(2):84–88, February 1999.
- [15] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323, 1999.
- [16] S. Jha, L. Kruger, and P. McDaniel. Privacy preserving clustering. In *10th European Symposium on Research in Computer Security (ESORICS '05)*, Milan, Italy, September 2005.
- [17] J. Kilian. Founding cryptography on oblivious transfer. In *Proceedings of the 20th Annual Symposium on Theory of Computing*, pages 20–31, 1988.
- [18] E. Kushilevitz and R. Ostrovsky. Replication is not needed: Single database, computationally-private information retrieval. In *Proceedings of 38th Annual IEEE Symposium on Foundations of Computer Science*, pages 364–373, 1997.
- [19] D. Malkhi, N. Nisan, B. Pinkas, and Y. Sella. Fairplay – A secure two-party computation system. In *Proceedings of Usenix Security*, pages 9–13, August 2004.
- [20] S. Meregu and J. Ghosh. Privacy-preserving distributed clustering using generative models. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM'03)*, pages 211–218, Melbourne, Florida, November 2003.
- [21] H. Polat and W. Du. SVD-based collaborative filtering with privacy. In *The 20th ACM Symposium on Applied Computing, Track on E-commerce Technologies*, pages 13–17, Santa Fe, New Mexico, March 2005.
- [22] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom, and J. Riedl. Grouplens: An open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 Computer Supported Collaborative Work Conference*, pages 175–186, Chapel Hill, North Carolina, 1994.
- [23] T. Sander and C. F. Tschudin. Towards mobile cryptography. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 162–167, Oakland, USA, 1998. IEEE Computer Society Press.
- [24] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis. State-of-the-art in privacy preserving data mining. *ACM SIGMOD Record*, 33(1):50–57, 2004.
- [25] E. Vozalis and K. G. Margaritis. Analysis of recommender systems’ algorithms. In *Proceedings of the 6th Hellenic European Conference on Computer Mathematics and its Applications (HERCMA-2003)*, Athens, Greece, 2003.
- [26] A. C.-C. Yao. Protocols for secure computation. In *Proceedings of 23rd IEEE Symposium on Foundations of Computer Science*, pages 160–164, 1982.
- [27] A. C.-C. Yao. How to generate and exchange secrets. In *Proceedings of 27th IEEE Symposium Foundations of Computer Science*, pages 162–167, 1986.