Applied Intelligence manuscript No.
(will be inserted by the editor)

Strategies for avoiding preference profiling in agent-based e-commerce environments

Emilio Serrano · Jose M. Such · Juan A. Botía · Ana García-Fornes

Received: date / Accepted: date

Abstract Agent-based electronic commerce is known to offer many advantages to users. However, very few studies have been devoted to deal with privacy issues in this domain. Nowadays, privacy is of great concern and preserving users' privacy plays a crucial role to promote their trust in agent-based technologies. In this paper, we focus on preference profiling, which is a well-known threat to users' privacy. Specifically, we review strategies for customers' agents to prevent seller agents from obtaining accurate preference profiles of the former group by using data mining techniques. We experimentally show the efficacy of each of these strategies and discuss their suitability in different situations. Our experimental results show that customers can improve their privacy notably with these strategies.

Keywords privacy, agent-based e-commerce, preference profiling, interaction analysis, data mining

Emilio Serrano
Polytechnic University of Madrid
Madrid, Spain
E-mail: eserrano@gsi.dit.upm.es

Juan A. Botía
University of Murcia
Murcia, Spain
E-mail: juanbot@um.es

Jose M. Such
Lancaster University
Lancaster,United Kingdom
E-mail: j.such@lancaster.ac.uk

Ana García-Fornes
Polytechnic University of Valencia
Valncia, Spain
E-mail: agarcia@dsic.upv.es

1 Introduction

Privacy is of great concern in the era of global connectivity, in which everything is inter-connected anytime and anywhere, with more than 2 billion world-wide users with connection to the Internet as of 2011[1]. In the real world, everyone decides (at least implicitly) what to tell other people about themselves. In the digital world, users have more or less lost effective control over their personal data. Users are therefore exposed to constant personal data collection and processing without even being aware of it [17]. In this way, Garfinkel [18] suggests that nowadays users have only one option to preserve their privacy: becoming hermits and not using e-commerce sites, online social networks, etc. However, considering the increasing power and sophistication of computer applications (mainly due to new information technologies such as agent-based technologies) that offer many advantages to individuals, becoming a hermit may not really be an option. However, all of these advantages currently come at a significant loss of privacy [6].

Agent-based electronic commerce refers to electronic commerce in which agent technologies are applied to provide personalized, continuously running, semi autonomous behaviour [14]. Many studies have been made on this topic during the past two decades [19]. However, to our knowledge, privacy is seldom considered in agent-based e-commerce applications. This leads to applications that invade individuals' privacy, causing concerns about their use. Indeed, recent studies show that 90% of users are concerned or very concerned about privacy [63]. Moreover, almost 95% of web users admitted they have declined to provide personal information to web sites at one time or another when asked [24]. Thus, it is crucial for Multi-agent Systems to consider privacy in order to be of wide use [45]. This can potentially promote principals' trust in agent-based technologies. This trust is needed for principals to be willing to engage with and delegate tasks to agents [14].

Two information-related activities can represent a major threat for privacy: information collection and information processing [39]. These activities can lead to many privacy breaches [50, 55]. Information collection refers to the process of gathering and storing data about an individual. Personal data is transferred on-line even across the Internet. Without appropriate protection mechanisms a potential attacker could easily obtain information about principals without their consent. In order to avoid undesired information collection, sensitive personal information must be protected from access by any other third party that is different from the agent to which the information is directed to. Therefore, avoiding information collection requires security to control the access to personal information [36]. Current security-concerned Agent Platforms avoid undesired information collection for the messages exchanged by the agents running on top of them. For instance, Jade [25], Magentix [54], Magentix2 [57], AgentScape [38], SECMAP [60], Tryllian ADK [65], Cougaar [34], SeMoA [41], and Voyager [40] are security-concerned APs. These Agent

[1] `http://www.internetworldstats.com/stats.htm`

Platforms allow the encryption of messages before transferring them and the decryption of messages once they are received. As a result, if an agent A sends a message to an agent B using these technologies, A is sure that B will be the only one able to read this message.

Avoiding undesired information collection is a necessary condition to preserve privacy, but it is not sufficient. It prevents unauthorized third parties from accessing undesired information. If an agent A sends personal information to an agent B in a confidential fashion, external third parties will not be able to access it. However, agent B will obviously receive this personal information. The point is that agent B can then process the received personal information, unless specific measures for preventing information processing are adopted before sending this information [53].

Information processing refers to the use or transformation of data that have already been collected [52], even though this information has been collected by mutual consent between two parties. An example of information processing is profiling [22,13]. As stated in [22], profiling is "*the process of 'discovering' patterns in data that can be used to identify or represent a human or nonhuman subject (individual or group) and/or the application of profiles (sets of correlated data) to individuate and represent an individual subject or to identify a subject as a member of a group (which can be an existing community or a discovered category) and/or the application of profiles to individuate and represent individuals or groups*".

One of the most common types of profiling is called preference profiling (also called buyer profiling) in e-commerce environments [62,47], in which sellers obtain detailed profiles of their customers and tailor their offers regarding customers' tastes. Indeed, much of the agent-based e-commerce literature has been precisely focusing on processing customers' information to achieve more effective negotiation strategies from the point of view of the seller [4,8, 23,2]. All these approaches try to construct a detailed profile of the customer so that sellers can tailor their offers based on customers' tastes. These profiles can represent a serious threat to privacy. For instance, these profiles can be used to perform *price discrimination* [35]. Vendors could charge customers different prices for the same good according to the customers' profiles, i.e., if a seller knows that some good is of great interest to one customer, the seller could charge this customer more money for this good than other customers for the same good. For instance, in 2000, Amazon started to charge customers different prices for the same DVD titles [51]. When the story became public, Amazon claimed that this was part of a simple price test and discontinued this practice. Another example of privacy threat due to the use of these profiles is what is known as *poor judgment* [49]. This is when individuals are judged and subsequently treated according to decisions made automatically based on incorrect or partial personal data. For instance, companies usually divide their potential customers into similar groups based on customers' characteristics (known as customer segmentation). This practice can lead to exclusion of people from services based on potentially distorted judgments [52].

In this article, we focus on strategies for avoiding preference profiling in agent-based e-commerce scenarios. Specifically, these strategies can be applied by customers' agents that participate in agent-based e-marketplace to minimize the chances for sellers to perform successful information processing activities. Most of these strategies are based on the use of pseudonyms[2] and on different approaches for using/reusing/changing them. Specifically, six strategies are considered: the use of a unique pseudonym, a pseudonym per negotiation, a pseudonym change if model accurate enough, the use of a pseudonym per group, a pseudonym per preference, and introducing fake preferences. We experimentally illustrate the efficacy of these strategies in a case study of a wine trade system. Moreover, in a similar fashion, a customer can check the improvement in the privacy given by a specific strategy reviewed in this paper since she knows not only the requests happened in previous conversations, but also the set of future possible requests according to her preferences. The experimental results show that these simple strategies can evade the capacity of building a good *preference model*[3] (i.e., a model to predict if a product is wanted or unwanted by a customer according to her specific preferences) by using data mining techniques such as decision trees, rule-based classifiers, naive Bayesian classifiers, and multilayer perceptrons [64].

The remainder of the paper is organized as follows. Section 2 describes some related works. Section 3 presents our formal framework and how agents can build and evaluate preference profiles according to our formal framework. Section 4 review strategies for avoiding preference profiling. Section 5 describes the experiments we conducted and section 6 discusses the results obtained. Finally, section 7 presents some concluding remarks.

2 Related Work

Agent-based e-commerce has received much attention in the last two decades [14]. This is due to the fact that agent-based e-commerce offers many advantages with respect to traditional e-commerce, such as (semi-) autonomous behaviour so that agents perform transactions on behalf of their users [5]. There have been many studies in the agent-based e-commerce research field that deal with the problem of obtaining accurate models (or profiles) of customers' preferences. For instance, Hindriks and Tykhonov [23] present a generic framework based on Bayesian learning to learn preference models from opponent information. Buffett and Bruce [8] introduce a classification technique to approximate opponent's preferences over the domain of possible offers. Aydoğan and Yolum [4] propose a learning algorithm to build a preference model to understand

[2] A pseudonym is an identifier of a subject other than one of the subject's real names [37]. Pseudonyms have been broadly used by human beings in the real world. For instance, in the 19th century when writing was a male-dominated profession, some female writers used male names for their writings. Nowadays, in the digital world, there are a great number of pseudonyms such as usernames, nicknames, e-mail addresses, sequence numbers, public keys, etc.

[3] In this paper, we use the terms preference profile and preference model indistinctly.

consumer's needs and to offer services that respect consumer's preferences. Serrano et. al explore the use of data mining techniques [42], social network analysis [44], and graph theory [43]; to obtain theories which explain agents' interactions and their preferences. The studies presented above are aimed at using such preference profiles to have capital information about customers during negotiations with them. In particular, these studies are aimed at having preference models about customers to be able to adopt more effective negotiation strategies that can directly lead to revenue increases. However, these increases in revenue for sellers will be at the expense of customers's budget. That is, if a seller knows customers' preferences, the seller can take advantage of this to propose the deals that are most beneficial to her. The seller can take advantage of this in many different ways as stated in the introduction. For instance, if the seller knows that a product is your favourite she could charge more for this product to you (because she knows you will buy it anyway) than to other customers. Another example would be that the seller only offers a customer the most expensive product from all of the products she knows a customer likes. In order to put customers in a less imbalanced position during the negotiation, it is crucial that her preferences are hidden as much as possible.

Apart from being able to hold privileged information during negotiations, sellers could also use customers' profiles to perform tailored advertising (e.g., providing suggestions for products that customers may like) or personalise product searches. This could indeed be seen as beneficial for users. Nevertheless, recent studies show that most users reject tailored advertising. A survey done in 2009 considering 1000 US adult citizens that are internet users found that 86% reject this practice [59]. Another survey made in 2009 considering 3660 US citizens says that only 30% of them would be willing to disclose personal information like browsing behaviour captured by websites on which they have registered in order to improve user experience [58]. The results of these surveys point out that few users are really willing to lose privacy in exchange of personalisation or tailored advertising.

There have been some efforts in the last decades to minimize profiling. One of the most important has been the rise of Privacy-Enhancing Technologies (PETs) that have focused on preventing profiling, such as privacy-enhancing identity management [11]. The building block of privacy-enhancing identity management is Pseudonymity [20], which is the use of pseudonyms as identifiers [9]. In particular, Privacy-Enhancing Identity Management Systems (PEIMS) [11, 20] are systems that provide users with facilities to support the management of their pseudonyms (i.e. creation and selection of the pseudonyms to be used). According to [21], one of the main questions that is relevant for pseudonyms to avoid profiling is the amount of information that can be gathered by linking the data that have been disclosed under the same pseudonym. In this way, in [20] the authors point out pseudonyms should be changed from time to time to avoid profiling. However, there are few proposals of strategies to prevent profiling. Instead, users are left with the complete responsibility of deciding when and how they manage their pseudonyms. Moreover, the few

existing proposals of strategies usually base on generating a new pseudonym for each new transaction, what is known as transaction pseudonyms [9]. However, vendors usually try to avoid this kind of strategies by means of different approaches to achieve customer loyalty such as price discounts, allotment of points that can be used for future purchases, and so on [31].

Pseudonym-based techniques have also been used in agent technologies. Users connect to the IntelliShopper agent [33] using a pseudonym to avoid the link between the profiles that IntelliShopper has about customers and their real identity. Moreover, users can use different pseudonyms for IntelliShopper to have separate profiles for separate activities. However, the authors of this work leave the user with the responsibility for creating their pseudonyms. Moreover, they do not provide any pseudonym management facility.

There have been other approaches proposing the integration of pseudonyms into agent architectures and frameworks, such as Van Blarkom et al. [61]. In this way, Such et al. [56] present a pseudonym management model that has been implemented into an agent framework [57]. Warnier and Brazier [62] also present a proposal for supporting pseudonym management in an agent framework. Both proposals include the necessary mechanisms for agents to be able to manage their pseudonyms automatically but nothing is said about when a pseudonym should be changed or not, as pointed out in [55]. In this paper, we review strategies for agents to avoid preference profiling based on the use of pseudonyms.

3 Formal Framework

3.1 Agent Negotiation and Agents Preferences

In this section, we define a generic negotiation protocol that is described in figure 1. A customer requests a product with a message *request* that can be answered with a message *model* (product requested available), *alternative* (alternative to requested product is offered), or *not offer* (negotiation aborted by the seller). The messages *model* and *alternative* can be answered with a message *accept* (accepting the purchase of the product ordered by the customer or the alternative proposed by the seller), with a message *quit* (negotiation cut by the customer) or with a *request* (the customer makes a new product proposal to be acquired).

This protocol includes the main elements in a negotiation [48]: the proposal of a product or service and the ability to re-negotiate terms of this service by both the customer and the seller. There is a large number of negotiation protocols in the literature. Some of them do not include some of the functionality of this formal framework to make them simpler. For example, a negotiation implementation can ignore the possibility of iterating [15] or the counter-offers [15,16] (i.e., the protocol allows only to accept or reject the first offer). On the other hand, there are negotiation protocols covering more issues than the choice of product purchased, for example, checking that the product

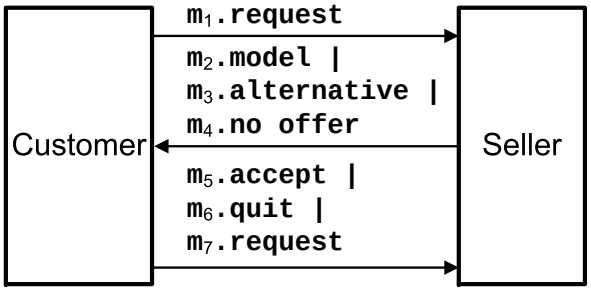

Fig. 1 Negotiation Protocol.

is indeed the requested [46]. However, this latter group includes the protocol described in this formal framework and, therefore, the strategies presented can be applied on them.

The protocol of our formal framework negotiates a type of service or product that is specified by a set of terms $T = t_1, t_2, ...t_n$, where n is the number of terms known and each term is a triple: term name *id*, relational operator *op*, and value *va* (e.g., $price = low$). Without loss of generality, we assume that the participants in the negotiation know the data type of each term. These terms T are the content of the messages *request* and *alternative*. Of course, the list of terms is not fixed at any time, hence, both customer and seller can add a term (increasing the value of n). Associating a protocol with a type of product, customers and sellers understand the possible terms of such class of goods. Limiting the discussion of the proposal to a specific type of product does not make it less general because different identifiers can be used depending on the type of product traded in the same protocol of figure 1. Another option is to extend the protocol with a variable "product" in the messages. Then the proposal would be reproduced for each value of the variable in isolation.

Let us define the preferences of a customer P as a set of preferred models Pm that are joined by the logical "or". Pm is a set of preferred terms pt that are joined by the logical "and". And pt preferred terms are triples in the same format as the terms of service (name of term *id*, relational operator *op*, and value *va*). More formally, $P = \{Pm_1, Pm_2, ...Pm_{|P|}\}$ and $Pm_i = \{pt_1, pt_2, ...pt_{|Pm_i|}\}$ with $1 \leq i \leq |P|$. The preferences set P can be viewed as a disjunctive list of products that the customer wants and the products as a conjunction of preferred terms. Therefore, an agent might have among their preferences for the purchase of books:

$$P = ((price = low) \wedge (year \geq 2008)) \vee (author = \text{“}Oscar\ Wilde''\text{)}$$

which means that the agent is interested in cheap and recent books or books by a particular author.

To complete the formal framework, let us define the mental processes that customer agents follow when interacting by the protocol. At the time of launch-

ing a request, either to initiate the conversation or to iterate in the negotiation, the customer chooses an element Pm_i of P and makes a product request specifying all the terms known in T satisfying Pm_i. In other words, the customer has general preferences but demands concrete products. The strategies evaluated in this paper prevent precisely from giving the general preferences to the seller. On the other hand, if the seller offers an alternative that meets the customer's preferences, in principle, this customer will accept the product even if it was not what the agent initially asked. Otherwise, the negotiation would be a simplified case of this formal framework where the seller does not offer alternatives [15]. The fact that the seller may propose not required alternatives is what gives the possibility of using information about customer preferences against this customer (although, even in the simplified case, the seller could use these preferences to send targeted advertising to the customers). Of course, the customer has always the freedom to cut a negotiation by the message *quit*, even if the seller has offered an acceptable alternative.

Strictly speaking, this is all we need to develop strategies for the protection of customer preferences: a negotiation protocol, a format for the content of messages and customers' preferences, and the mental processes that allow a customer to use the preferences to request specific products and accept (or reject) alternatives proposed by the seller. Note that although the role of the seller has not been defined, it is assumed that the generation of alternatives considers four factors [4]: availability of product in stock, utility function that determines the benefit, a function of proximity to provide alternatives similar to the product requested by the customer, and finally, a preference model that may have been built based on previous interactions. The next section explains how to build these preference model.

3.2 Building and evaluating preference models

Regarding the construction and evaluation of preference models, the essential difference with related works [2,4,8,23], see section 2, is that we adopt the point of view of the customer. Since the customer knows her real preferences, the construction and evaluation of these models can be performed by the interested party to check the efficacy of the strategies evaluated in this paper. On the other hand, from the point of view of the seller, this test is not feasible in real life because only a biased viewpoint of customer preferences is known.

Tuples or instances, that constitute the training data for the models, consist of an attribute for each term in T already known, see section 3. In case that negotiations do not include a value for a certain term in T, a label is included to indicate that the value is unknown ("?"), most data mining algorithms can deal with unknown values. Besides, if a negotiation includes a new term that has not appeared in previous negotiations, the term is included as a new attribute and a value of unknown is assigned in the previously generated tuples. Note that most data mining techniques require an equal number of attributes (features) for all tuples. Besides, the formal framework specification

minimizes the chances of an unknown value appears since a customer, despite having preferences based on specific terms, asks for complete products that give value to all the terms known by the customer.

In addition to the terms of the product negotiated T, the tuples contain an additional attribute with the class of the instance. This attribute can take values of $'+'$ and $'-'$ to indicate that the instance is a *positive example* (the terms define a product or service that a customer wanted) or a *negative example* (defining products that the customer did not want). Considering the protocol specified in the formal framework, the positive examples are given by: (1) the terms in the *request* messages, i.e., the initial one (message m_1 in figure 1) or after iterations in the loop (message m_7 in figure 1); and (2) the terms in the messages *alternative* if a message *accept* happens after the alternative. Similarly, a negative example is registered with the terms included in the messages *alternative* if a message *request* occurs after the alternative. Note that an alternative message followed by a *quit* is not necessarily a negative example (a customer can cut the negotiation after receiving an acceptable alternative, for example because it found a better deal). In summary, considering a pair of consecutive messages, m_1 and m_2, and a constructor $Instance$ having as parameters the terms negotiated and the class of the instance, the following rules generate the training data set I to build the preference models:

- if($m_1.performative = request \vee (m_1.performative = alternative \wedge m_2.performative = accept)$) then new Instance ($m_1.content$, "+");
- if($m_1.performative = alternative \ \wedge \ m_2.performative = request$) then new Instance ($m_1.content$, "-");

Since training data is labelled with a class (wanted or not wanted by the customer), building a preference model from the training data described above is a classification problem. Classification is an instance of supervised learning, i.e. learning where a training set of correctly-identified observations is available. Section 3.3 explores some suitable classifiers.

Finally, the reliability of the model must be defined. If the preference models were produced by sellers, the ignorance of the actual preferences would leave few reliable tools to evaluate these models. The *cross validation* [64] can be used to obtain an indicator of the extent to which the model classifies the training data adequately. Nevertheless, a customer can perform this evaluation more precisely because it knows not only the past instances but also all possible instances. An evaluation data set composed of an instance for each preferred model $Pm_i \in P$ is not enough to assess the model because: (1) the test data would contain no negative example, therefore, this measure of reliability cannot detect false positives (a negative example classified as positive); (2) the preferences admit operators different to $'='$, and therefore, it is not possible to create a single instance covering these preferences (e.g., $year \geq 2008$); and (3) an model "all instances are +" would offer 100% of accuracy. A more rigorous solution is to analyse the model to decide if it is logically equivalent to the preferences P. A decision tree, for example, can be seen as a set of classification rules linked by logical disjunction where each rule is obtained by joining the nodes on a path of each leaf to the root by the

conjunction. The obvious flaw in this approach is the computational cost and the difficulty of establishing this equivalence in some data mining techniques (e.g., neural networks). The compromise we adopt in this proposal is to generate a set of test data large enough and composed of: (1) a number of positive examples for each preferred model $Pm_i \in P$ which satisfy the terms in Pm_i (i.e., generating possible contents of the message *request*); and (2) a number of negative examples for each preferred model $Pm_i \in P$ whose terms do not meet any preference in P. The percentage of correctly classified instances is the degree of accuracy of the model and the objective of this paper is to make this degree as low as possible. Note that an extremely simple model (e.g, " all instances are +") offers a 50% of instances correctly classified with this evaluation approach.

3.3 Some suitable techniques to build preferences models

This section describes some data mining techniques which are suitable to build preferences models.

- *Decision trees.* One of the most popular learning techniques are the decision trees. A decision tree is a classifier expressed as a recursive partition of the instance space. These directed trees have a node called root (with no incoming edges), internal or test nodes (with one or more outgoing edges), and nodes called leaves or terminals (with no outgoing edges) [64]. Instances are classified by navigating them from the root to a leaf, the leaf indicates the class and the remaining nodes in the path indicate values of an attribute in the instance. Therefore, they classify a data set in a tree-shaped structure where the leaves are the classes and the nodes decide the value for an attribute. Decision trees algorithms work using a divide-and-conquer approach. First, an attribute is selected to be the root node and one branch is generated for each possible value splitting up the instances into subsets, one for every value of the attribute. Then the process can be repeated recursively for each branch but considering only those instances that can reach the branch according to the value of their attributes. The algorithm stops developing a part of the tree if all instances have the same classification (same leaf of the tree) [64].
- *Classification rules.* The antecedent or precondition of a classification rule is a series of attributes with values assigned, and the consequent or conclusion gives the class or classes that are applied to instances covered by that rule [64]. Preconditions are usually connected by the logic operator "and". The different rules of classification are supposed to be connected by the operator "or". Therefore, if any rule can be applied, the conclusion of this rule is given to the instance of the data set. Classification rules can be easily translated into decision trees: every antecedent of the rule is a node of the tree with a condition on the path from the root to that leaf, where the consequent of the rule is the class assigned by the leaf. Classification rules algorithms work using a coverage approach [64] which means that

a rule that covers instances in the class is identified (and excludes ones not in the class), these instances covered are separated, and later the algorithm continues classifying those that are left. As an application example, Márquez-Vera et al. [32] employ several rules based classifiers and decision trees to predict student failure at school.

- *Bayesian networks as a classifier.* Bayes' theorem shows how to determine the conditional probability of B given A knowing the conditional probability of A given B. Bayes's rule says that if you have a hypothesis H and evidence E related to that hypothesis, then $Pr[H|E] = \frac{Pr[E|H] \cdot Pr[H]}{Pr[E]}$, where $Pr[A]$ denotes the probability of an event A and $Pr[E]$ denotes the the probability of A conditional on another event B [64]. The Naive Bayes method assumes, naively, that attributes in a tuple are independent given the class. Therefore, if the evidence E is a combination of attributes values (let us split it in pieces of evidence for every attribute: E_1, E_2,...E_n), assuming they are independent, their combined probability is obtained by multiplying the probabilities. Hence, the probability of giving a class value h to an instance with the values E_1, E_2,...E_n in their n attributes is: $Pr[h|E] = \frac{(\prod_{i=0}^{n} Pr[E_i|h]) \cdot Pr[H]}{Pr[E]}$. Thus, Bayesian networks can be used as classifier calculating the previous expression for all class values and returning the class value which produces the maximum result, i.e. $argmax_h P(h|E)$ [64]. Bayesian networks classifiers are widely used to construct network intrusion detection approaches [27].
- *Multilayer Perceptron.* Neural networks are mathematical models inspired by biological neural networks and consist of set of interconnected artificial neurons. Neural networks are used to model complex relationships between inputs and outputs or to find patterns in data. A common type of neural networks is the so-called *feedforward* neural networks, where connections between the neurons do not form a directed cycle. The simplest kind of feedforward neural network is the single-layer perceptron network, which consists of a single layer of output nodes that are fed directly from the inputs via a series of weights. The main weakness of the perceptron is that it can only solve linearly separable problems. This issue is addressed by the multilayer perceptron, which connects simple perceptron-like models in a hierarchical structure so that non-linear decision boundaries can be represented [64]. The multilayer perceptron consists of three or more layers of nonlinearly activating artificial neurons. Some application examples are the use of neural networks to build up a map within an unknown environment [26] or the financial forecasting [28].

Note that all the techniques discussed are classifiers since the data is labelled with a class (wanted or not wanted by the customer). Besides, a dataset large enough is considered. Therefore, the use of unsupervised, semi-supervised or one-shot learning algorithms would achieve less accurate preference models.

The comprehensive comparison of the best data mining technique to obtain a preference model is beyond the scope of this paper. Works in this line [4] have compared the use of sophisticated techniques aimed at learning pref-

erences (e.g., *Candidate Elimination Algorithm reviewable*, RCEA) with standard mining techniques (e.g., ID3 decision tree) concluding that decision trees obtain the same accuracy when classifying a product as a positive or negative example. Besides, customers can generate models using all the techniques available and consider the most accurate model as an indicator of the extent to which the seller may have learned their preferences.

4 Strategies for Avoiding Preference Profiling

In this section, we present some strategies that customers' agents can use to preserve their principals' privacy. Most of these strategies based on the use of pseudonyms. Thus, we assume that agents will be running on top of a privacy-enhancing agent framework that provides pseudonym-management facilities, such as Magentix2 [57] and AgentScape [62]. We also assume that payments are carried out using some kind of anonymous payment mechanism and deliveries are carried out using some anonymous delivery system. Hence, credit card numbers and delivery addresses do not need to be disclosed when an agent acquires a product and the only identifiying information from negotiation to negotiation is the pseudonym used by the customer[4]. Finally, we also assume the use of an underlying anonymous communication technology (e.g. TOR [12]) so that the IP addresses and other whereabouts are hidden.

The reviewed strategies are detailed below:

4.1 Unique pseudonym

The first strategy consists of using a unique pseudonym. The customer's agent uses a pseudonym that hides the real identity of its principal (the customer). However, the customer's agent never changes its pseudonym and does not use any other pseudonym. Specifically, in our framework, the customer uses the same pseudonym for all of the runs of the negotiation protocol. According to [21], one of the main questions that is relevant for pseudonyms to avoid profiling is the amount of information that can be gathered by linking the data that have been disclosed under the same pseudonym. Social security numbers in the USA are a clear example of a pseudonym that it is usually used for a long time and in different contexts. This allows different pieces of personal information disclosed (even in different contexts) to be linked to each other. This strategy is mainly included for control reasons in the experiments that we performed and that are detailed in the evaluation section (section 5).

[4] For instance, a payment system based on the untraceable electronic cash presented by Chaum et al. [10] could be used for anonymous payments. For anonymous deliveries, the privacy-preserving physical delivery system presented by Aïmeur et al. [3] could be used.

4.2 Pseudonym per negotiation

In the second strategy, the customer changes its pseudonym to a newly created one for each new negotiation with the seller. It can be seen as just the opposite to the previous strategy (unique pseudonym for all of the negotiations). The main aim of this strategy is to make harder for the seller to relate different negotiations with the same customer to each other, i.e., the seller could think that it is negotiating with different customers each time. This strategy is what is known as transaction pseudonyms in the privacy-enhancing identity management literature [9]. Moreover, it is usually regarded as the most privacy-preserving strategy for avoiding preference profiling. However, very few works try to practically show the real efficacy of this strategy. In the experiments section (section 5) we compare its efficacy to the efficacy of the other strategies explained in this section.

Vendors usually try to avoid this strategy. They usually conduct different approaches to achieve customer loyalty such as price discounts, allotment of points that can be used for future purchases, and so on [31]. Therefore, customers may be interested in strategies between *Unique Pseudonym* and *Pseudonym per negotiation*.

4.3 Pseudonym change if model accurate

A strategy that is between *Unique Pseudonym* and *Pseudonym per negotiation* is that of changing a pseudonym before the seller gets an accurate preference profile. That is, the customer does not use a different pseudonym per negotiation but can reuse the same pseudonym for a number of negotiations, and then, change this pseudonym before the seller can get an accurate preference profile.

In this strategy, the customer constructs itself one (or more) preference model(s) with one (or more) learning technique(s) before a new transaction. The training data for constructing the preference model are the past negotiations in which the customer used the current pseudonym. The customer changes its pseudonym for the next negotiation if the preference model constructed (or the most accurate one from all of the preference models constructed) has accuracy greater than some threshold of tolerance defined by the customer.

Section 3.2 explains how to evaluate the preference models to obtain the accuracy in terms of correctly classified instances in the validation phase of the preference model constructed. Several observations must be explained for this strategy. (1) As explained in section 3.2, the accuracy is usually greater than 50%, so the threshold should be set above this value. (2) The threshold is not an upper bound of the accuracy that can be reached by the seller. This is because the customer verifies the accuracy before starting a new negotiation. Therefore, if this extra negotiation is long enough, lots of data could be provided and accuracy over the threshold fixed by the customer could be achieved. (3) This

strategy is computationally very expensive since it involves the construction of a preference model for each data mining technique considered and for each negotiation. There are several possibilities to improve the efficiency: build the model only after several negotiations, using data mining techniques which can add information to the model without rebuilding it, etc. (4) The expected behaviour of this strategy involves sharp declines in the information provided to sellers corresponding with the moments of change of pseudonym.

4.4 Pseudonym per group

Another possible strategy is for a group of customers to share the same pseudonym, i.e., to make a coalition. The more customers join the coalition and the more different their preferences are, the more effective this strategy should be. This is because there are more chances for the seller to find contradictory preferences, i.e., the seller will find contradictions in the customer's favourite products. It could seem that the obvious drawback of this strategy is that it obtains privacy with regard to sellers sacrificing privacy with regards a group of customers. However, this should not imply any problem if the group is composed of customers that trust each other. Moreover, even when customers do not trust each other there are other solutions — such as using an anonymous communication mechanism like TOR [12] that can be added to agent frameworks as shown in [29] — that will make the seller but also the customers in the group unable to know which customer bought which product — recall that we are also assuming that payments and deliveries are anonymous (see beginning of section 4).

4.5 Pseudonym per preference

Another strategy is to use a different pseudonym for each preference. This is aimed at splitting the whole preference profile of a customer in many little preference profiles that only cover a part of the complete customer's preferences. For the case of intelligent agents, this strategy was first envisioned by Van Blarkom et al. [61]. They proposed to place what they called an Identity Protector between the agent and the rest of its potential partners to achieve that aim. However, they did not really propose any specific technique or mechanisms to do so. We base this strategy on the fact that preferences of a customer can usually be seen as a disjunctive list of specific preferences. Thus, we propose that customers use a different pseudonym for each disjunctive.

In our framework (see section 3), preferences P are composed of a disjunctive list of preferred models Pm_i. One strategy to avoid excessive knowledge of our preference is to use a pseudonym for each $Pm_i \in P$ with $1 \leq i \leq |P|$. The more preferred models Pm_i, the more effective this strategy is. Let us explain some observations. (1) If the customer is negotiating with a pseudonym associated with a preferred model Pm_i and the seller offers a product that does

not fit Pm_i but is acceptable considering the totality of the preferences P, the customer should just ignore such a product, and buy it when she is using the pseudonym for her part of the preferences corresponding to such product. (2) This strategy is particularly useful to detect abusive prices imposed by the seller according to customers' preferences. This is because the customer can use a pseudonym to query prices of products associated with other identities.

4.6 Fake preferences

Finally, an intuitive solution to the problem presented in this paper is simply lying about the preferences. That is, the customer tries to cheat on the seller by providing fake preferences. The customer can apply this strategy by holding a unique pseudonym through all of its negotiations with the seller. Then, the customer can perform actions to provide fake preferences depending on the specific framework. In our framework, the customer may insert false positives examples in the data base of the seller requesting not wanted terms, since the protocol specified in the formal framework, see section 3, allows to fake a change of mind using the message *quit*. Of course, the use of other negotiation protocols may restrict this possibility. If the seller ignored all the products requested which are not finally purchased, this strategy would be equivalent to *Unique pseudonym*. This paper assumes that the seller includes all these fake changes of mind in the data mining model built, as occurs in many real cases. In Amazon for example, requesting information about a book does not involve a sale and the system does not forget the products in which we have been interested. Note that the remaining strategies also include the preferences given in all messages even if the protocol shown in figure 1 does not end successfully (i.e., with the *accept* message). Note also that a strategy of inserting false negatives instead of false positives is not practical because the customer does not know in advance the alternatives that the seller will provide.

5 Evaluation

This section provides experimental results on the effect of the strategies explained in section 4 for preserving privacy in a multi-agent negotiation based on the formal framework explained in section 3. In particular, we conducted several agent-based simulations using the MASON Multiagent Simulation Toolkit[5]. In these simulations, customer and seller agents follow the negotiation protocol described in section 3.1 to negotiate the purchase of wines [1,4]. The seller agent builds preferences models based on the techniques explained in sections 3.2 and 3.3. Besides, customers agents use the strategies presented in section 4 to prevent the seller from obtaining detailed models of their preferences.

[5] MASON website `cs.gmu.edu/~eclab/projects/mason/`

5.1 Experimental Setting

Agents negotiate types of wine [1,4]. In our simulations, a wine is specified by the following terms: colour; body; flavour; sugar; and country. The possible values for each of these attributes are shown in table 1. These fields and their values come from a well-known ontology [1] and the only difference is a simplification of the term "region" in the "country" field.

Attribute	**Values**
Colour	red, rose, white
Body	light, medium, full
Flavour	delicate, moderate, strong
Sugar	dry, offDry, sweet
Country	France, Portugal, Spain, Italy, USA, Germany, Australia, NewZeeland

Table 1 Wine Attributes.

We performed 100 simulations with a single seller agent negotiating 1000 times with 50 different customers, $\{C_1, C_2...C_{50}\}$. Each customer C_i with $1 \leq i \leq 50$ has a set of preferences $P_{i\%10}$ according to the following list with ten different profiles:

- $P_0 = ((colour = red) \wedge (body = full)) \vee ((colour = white) \wedge (body = light))$
- $P_1 = ((colour = red) \wedge (body = light)) \vee ((colour = white) \wedge (body = medium))$
- $P_2 = ((sugar = sweet) \wedge (country = France)) \vee ((sugar = dry) \wedge (country = Spain))$
- $P_3 = ((sugar = dry) \wedge (country = France)) \vee ((sugar = dry) \wedge (country = Italy))$
- $P_4 = ((body = light) \wedge (flavour = delicate)) \vee ((sugar = dry)) \wedge (country = Portugal))$
- $P_5 = ((body = full) \wedge (flavour = moderate)) \vee ((sugar = sweet) \wedge (country = Portugal))$
- $P_6 = ((colour = red) \wedge (body = medium) \wedge (flavour = moderate)) \vee ((colour = rose) \wedge (body = light) \wedge (sugar = dry)) \vee ((colour = white) \wedge (body = full) \wedge (sugar = dry))$
- $P_7 = ((colour = rose) \wedge (body = medium) \wedge (flavour = moderate)) \vee ((colour = white) \wedge (body = light) \wedge (sugar = dry)) \vee ((colour = red) \wedge (body = full) \wedge (sugar = dry))$
- $P_8 = ((colour = red) \wedge (body = medium) \wedge (flavour = moderate)) \vee ((colour = rose) \wedge (body = full) \wedge (country = Italy))$
- $P_9 = (country = USA) \vee (country = France)$

Besides the 50 customers agents, there is one seller agent which builds preferences models based on the past interactions with the customers. The data mining techniques used by the seller to build preferences models are all specific algorithms of the techniques explained in section 3.3: *J48*, a decision tree; *NNge*, an algorithm of classification rules; *NaiveBayes*, a Bayes Network learning algorithm; and *MultilayerPerceptron*, a multilayer perceptron classifier. The four implementations are open source [7] and have been used with its default parameters to allow the interested reader to reproduce the results obtained. Note that the paper is not focused on comparing different data mining techniques but on calculating the benefit of the strategies presented over an accurate model obtained by data mining. These four algorithms are employed to prove the generality of the approach presented.

```
body = light
|   colour = red: - (28.0)
|   colour = rose: - (23.0)
|   colour = white: + (51.0)
body = medium: - (98.0)
body = full
|   colour = red: + (49.0)
|   colour = rose: - (25.0)
|   colour = white: - (27.0)
```

Fig. 2 Preference model using J48. The notation `a=v : +/-` denotes that "if `a` has value `v` the target is classified as `+/-`". Every leaf includes the number of instances classified in parentheses. If this number includes a fraction, the denominator is the number of instances incorrectly classified in the training data by the rule from the leaf to the root.

An example of preference model built using J48 is displayed in figure 2. In particular, it displays a preference model which has been built with the information provided by the customer C_1 throughout 1000 negotiations and using the control strategy, unique pseudonym. The reader can check that the decision tree is logically equivalent to the preferences of this customer, P_1 (detailed above).

As explained in section 3.2, the assessment of the preference models is performed by a simple validation using a test set with: a number of positive examples for each preferred model $Pm_i \in P$ which satisfy the terms in Pm_i; and a number of negative examples for each preferred model $Pm_i \in P$ whose terms do not meet any preference in P. The percentage of correctly classified instances (CCI) of the test set by the preference model is the accuracy. For these experiments, the test set is composed of 1000 positive examples of 1000 negative examples.

Finally, some considerations for the implementation of the strategies in these experiments are detailed here. *Pseudonym if model accurate* has been configured with a threshold of 75%. That is, the customer changes its pseudonym if the agent calculates a CCI greater than this value for the current pseudonym before starting any negotiation. *Pseudonym per group* considers a group with the 50 customers in each negotiation. *Fake preferences* try to insert 10 false positives examples as requests in each conversation (if the seller offers the product requested but unwanted, the customer quits).

5.2 Experimental Results

Figure 3 shows the results for the six strategies and the four mining techniques studied. The figure shows that, as explained at the end of section 3.2, that 50% of correctly classified instances (CCI) is the best possible outcome since the simplest model of preferences achieves this result. For "no strategy", the decision tree only needs 200 interactions to get accuracy over 99.9%. These results are repeated for other data mining techniques with the exception of the Bayesian classifier that only gets about 85% of CCI. Therefore, we can

establish a CCI of 100% as worst possible outcome, since the seller is able to successfully predict all customer preferences. Once the best and worst case have been considered, the effect of the strategies explained in this paper are discussed and ordered from the worst to the best results:

- *Pseudonym if model accurate* This strategy obtains the worst results for all techniques except for the Bayesian classifier where *Pseudonym per preference* is slightly worse.
- *Pseudonym per preference* achieved 73% of CCI in all techniques from 200 negotiations.
- *Pseudonym per group* gets the best results with the Bayesian classifier, about 60% of CCI, and no technique reaches 70%.
- *Pseudonym per negotiation* gets the best results with the decision tree, the optimal result (50% of CCI). In the other techniques, it gets results slightly worse than *Fake preferences*.
- *Fake preferences* achieves the best results (except for the decision tree). Specifically, the number of CCI ranges from 50%, which is the best case possible (with Bayes), and 57% (with NNge).

In the following section, we thoroughly discuss the results obtained in these experiments.

6 Discussion

With the results obtained, we can conclude that the strategy *Fake preferences* is the best one for the case study presented because it improves the results of *Pseudonym per negotiation* and it does not require a change of pseudonym that may involve loss of privileges given by the seller. Of course, as discussed in section 4, there are cases where the application of this strategy is not feasible and where some of the proposed alternatives are useful. The worst strategy studied is *Pseudonym if model accurate* because the worst CCI is reached. Moreover, as explained in section 4, this strategy leads to abrupt changes in the information available to build the preference model.

The customer knows its own preferences and can create a data set to properly test the model constructed. Nevertheless, the seller has to resort to methods such as cross-validation [64]. What is the seller's perspective when the above strategies are applied to customers?. Figure 4 shows the results of cross validation for each strategy and using J48 as learning algorithm. *Pseudonym per negotiation* does not even appear, since the cross-validation with 10 folds needs at least 10 instances. *Pseudonym if model accurate* is the strategy that, on average, more effectively discourages the seller about the quality of the models built. *Pseudonym per group* obtains about 70% of CCI. *Fake preferences* 80% and *Pseudonym per preference* over 99% of CCI from 200 negotiations. In conclusion, while *Fake preferences* is the best strategy to hinder the seller from knowing customer preferences, *Pseudonym per preference* is the best strategy

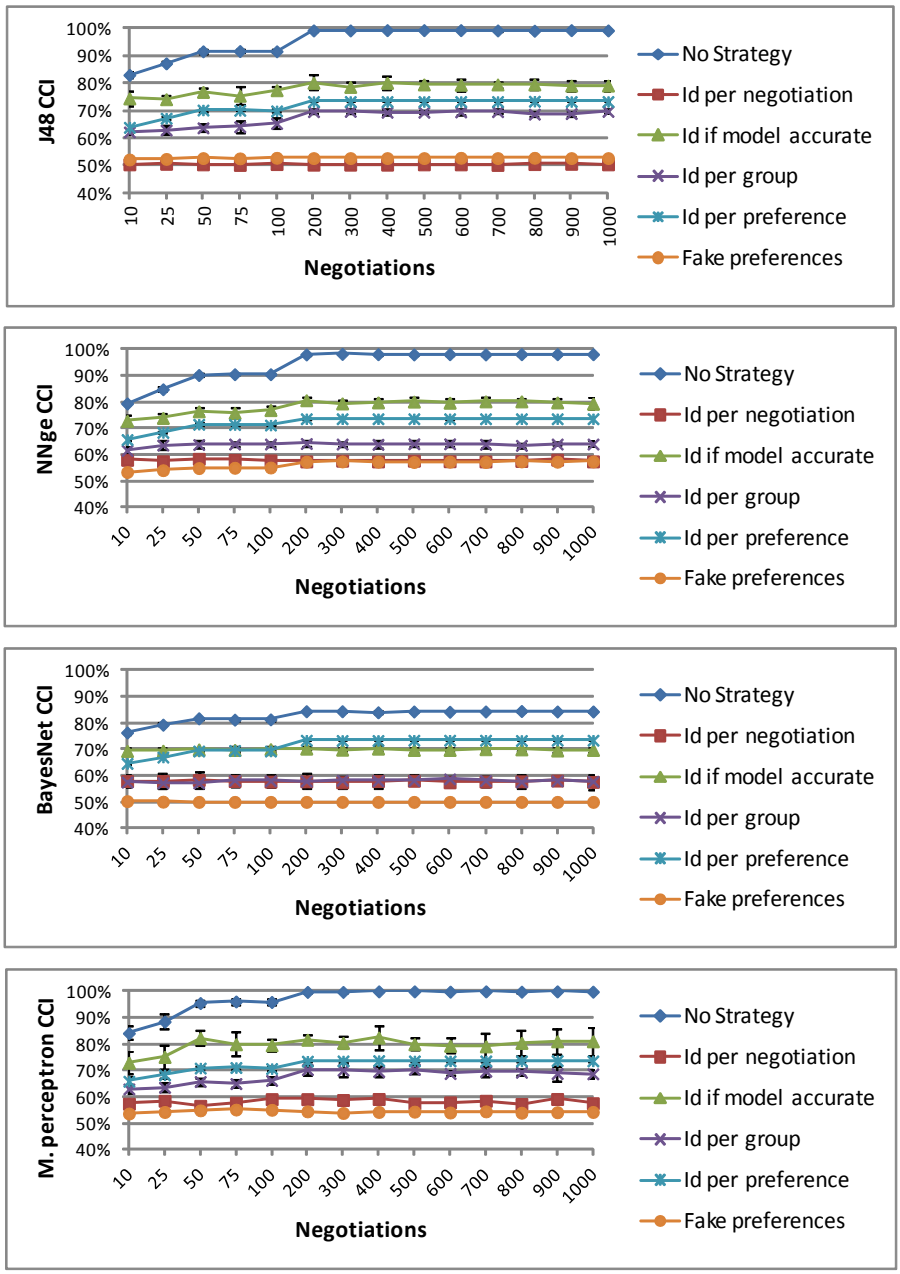

Fig. 3 Results for the 6 strategies from the perspective of the customer. Four data mining techniques are used: decision tree, rule-based classifier, naive Bayesian classifier and multi-layer perceptron. The lines show the mean for 100 experiments evaluating the preference model by simple validation (training data composed of 1000 negative cases and 1000 positive cases). The bar lines show the standard deviation of the mean, often imperceptible. The Y axis shows the correctly classified instances for a specific data mining technique, and the X axis the number of negotiations.

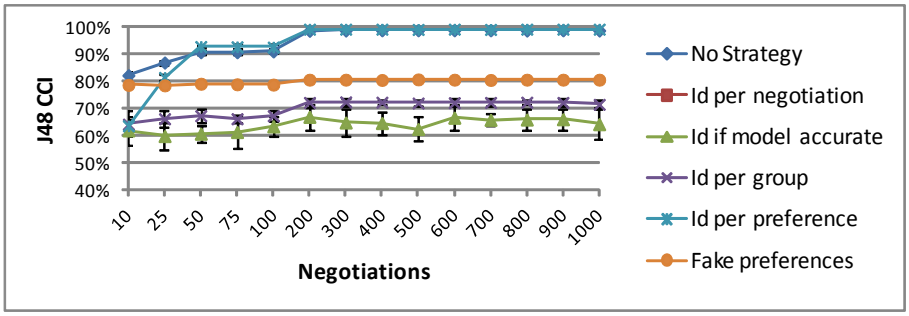

Fig. 4 Results for the 6 strategies from the perspective of the seller. The lines show the mean for 100 experiments evaluating the preference model by cross validation. The bar lines show the standard deviation of the mean, often imperceptible. The Y axis shows the correctly classified instances for a specific data mining technique, and the X axis the number of negotiations.

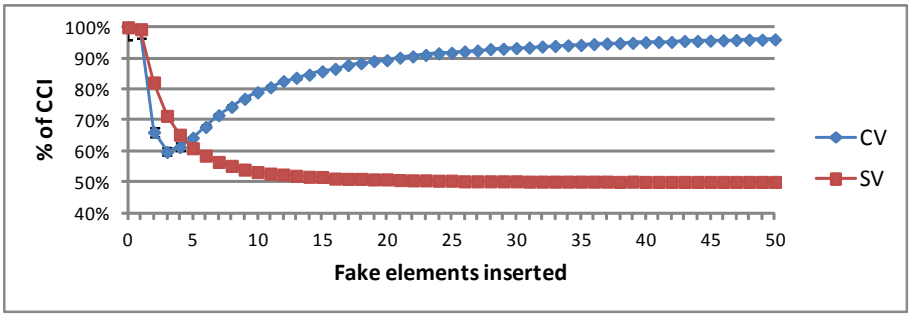

Fig. 5 Results for 100 negotiations using *Fake preferences* with a different number of false positives examples and J48 as learning algorithm. The lines show the mean for 100 experiments evaluating the preference model by simple and cross validation. The Y axis shows the correctly classified instances using simple validation or cross validation, and the X axis the number of fake elements inserted.

to make sellers think that excellent information about of those preferences has been gained.

Since the strategy recommended, *Fake preferences*, requires a parameter (the number of fake elements inserted), experiments have been conducted to study the influence of this parameter on the results, see figure 5. The results are close to the optimum for 10 false positives examples when the simple validation is used: accuracy of 53%. Using over 19 fake examples the accuracy is under 51%. The experiments also show that using more fake examples, the perception of the seller (which uses cross validation) is that the model is more accurate since much more examples have been provided.

In the same vein, figure 6 shows the effect of the threshold parameter in the *Pseudonym if model accurate* strategy. The chart shows that under 50%, this strategy behaves just like *Pseudonym per negotiation*, i.e. the simple validation offers around 50% of correctly classified instances. However, it has

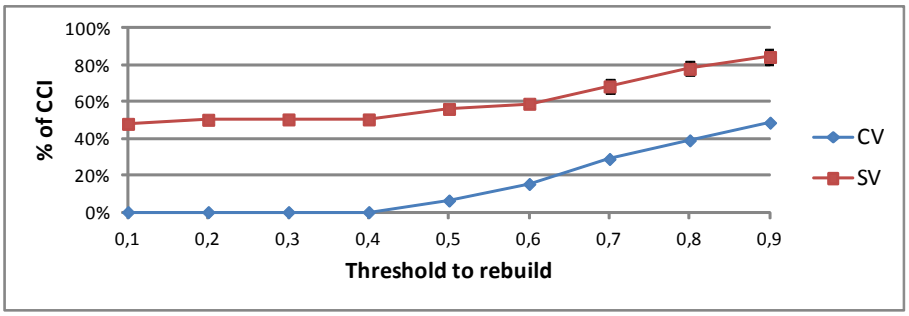

Fig. 6 Results for 100 negotiations using *Pseudonym if model accurate* giving different values for the threshold employed to decide when the model is rebuilt. The lines show the mean for 100 experiments evaluating the preference model by simple and cross validation. The bar lines show the standard deviation of the mean, often imperceptible. The Y axis shows the correctly classified instances using simple validation or cross validation, and the X axis the number of fake elements inserted.

the added computational cost of having to build a preference model before each negotiation. In these cases, the cross validation cannot be conducted because at least 10 instances in the training data are needed. Values equal or greater than 50% disclose more information about the preferences since the model is rebuilt with less frequency.

Finally, some experiments have been conducted to study the effects of the strategies studied when customers' preferences change. Specifically, each customer C_i with $1 \leq i \leq 50$ changes her preferences from $P_{i\%10}$ to $P_{i+1\%10}$, according to the list of profiles given in section 5. This change happens in the negotiation number 200, when, as shown in figure 3, a lack of strategy allows the seller to classify correctly almost 100% of the instances. Figure 7 details the results obtained for each strategy. When no strategy is used, the CCI goes from 99% with 199 instances to 46% with 200. This occurs because, although the training data only changes in one instance, the evaluation data (generated with the new customer's preferences) varies drastically. The strategies with the best behaviour without preferences changes (see figure 3), *Pseudonym per negotiation* and *Fake preferences*, do not present any significant change because they never allow the seller to reach a sound theory about their preferences. Consequently, these strategies are not affected if the preferences are not stable. This is also the case of *Pseudonym per group* but for a different reason. Since the preferences are exchanged according to the same list of profiles and in this strategy all customers share the same pseudonym, no change is perceived in the training data. Finally, *Pseudonym per preference* presents an abrupt CCI reduction (from 73% to 23%) when preferences change and this value is not recovered after 500 negotiations (CCI reaches only 72%). This is caused mainly because the number of preferred models (Pm_i, see section 4.5) can be different in the new customer profile. If the new profile has more preferred models, new

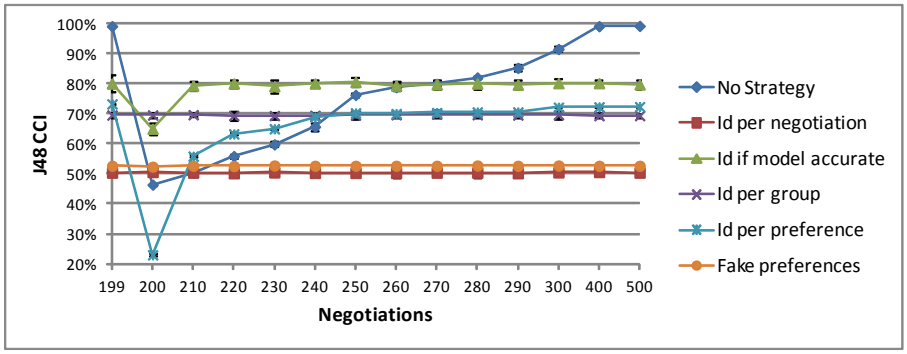

Fig. 7 Results for the 6 strategies from the perspective of the customer when preferences change in negotiation 200. The lines show the mean for 100 experiments evaluating the preference model by simple validation (training data composed of 1000 negative cases and 1000 positive cases). The bar lines show the standard deviation of the mean, often imperceptible. The Y axis shows the correctly classified instances for a specific data mining technique, and the X axis the number of negotiations.

pseudonyms are employed and the seller does not have any information about them. If the new profile has less preferred models, some pseudonyms are not used after the change, they do not generate new data for the seller, and they offer very low number of CCI because the evaluation data set has changed with the preferences.

Given the experiments detailed above, the use of *Fake preferences* is also recommended when preferences are not stable. However, *Pseudonym per preference* presents the best result at the moment of the change and after a few negotiations (less than 10 in these experiments). Therefore, another option not explored in this paper, is to alternate between different strategies during the history of the interaction with the seller. For example, *Pseudonym per preference* could be employed right after a preferences change, and after a while, the customer might start using *Fake preferences*.

Of course, the efficacy of the strategies and the quality of the preference model obtained by the seller depends on many factors, including the semantics exchanged (in this case wine) and the complexity of the preferences profiles of customers. Nonetheless, the results for this case study allow us to extrapolate a number of interesting properties of the strategies presented and hint at their potential in real domains.

7 Conclusions and future works

Agent-based e-commerce has received much attention in specialized literature. Specifically, this paper has shown that a large number of works deal with the problem of obtaining accurate models (or profiles) of customers' preferences in order to provide the sellers with more effective negotiation strategies. Since

these profiles can represent a serious threat to the privacy of customers, this paper focuses on reviewing and testing strategies to avoid the construction of these preference models. These strategies are mostly based on the use, reuse, or change of pseudonyms.

To illustrate the suitability of the strategies for a wide scope of contexts, this paper has detailed a generic framework to implement a negotiation between agents, define the preferences of the products negotiated by the customers, build a preference model using a generic data mining classifier, and evaluate this model from customers' viewpoint.

The six strategies evaluated are: the use of a unique pseudonym, a pseudonym per negotiation, a pseudonym change if model accurate enough, the use of a pseudonym per group, a pseudonym per preference, and introducing fake preferences.

The efficacy of these strategies have been tested in a wine trade system. The conclusion is that adding fake preferences is the best approach to hinder a seller from building an accurate preference model using the four data mining techniques considered (a decision tree, a rule-based classifier, a naive Bayesian classifier, and a multilayer perceptron). On the other hand, if the customer tries to make the seller think that her model is extremely accurate (when it is not), the use of a pseudonym per preference gets outstanding results. Finally, the use of the pseudonym per preference also gives the best results when preferences change, although fake preferences overcomes it after a few negotiations.

Given the number of experiments conducted, the results obtained are statistically sound. We strongly believe that the experimental setting has the potential to produce very similar results with human users instead of agents. However, in order to prove this, further work performing experiments with actual human users is necessary.

As pointed out in different studies [58,59], there is a minority but important number of people that would be willing to lose privacy in exchange of tailored advertising or personalised searches. As future work, we plan to use some of the existing privacy-utility tradeoff frameworks [30] that support decisions about whether disclosing personal information would be worth the privacy that would be lost. Thus, the idea would be to decide whether or not to apply the strategy that maximises the privacy-utility tradeoff at each point in time, i.e., it preserves as much privacy as possible while maximising the utility gained.

Finally, our main future work is to test the strategies presented in more real scenarios. Even when there are plenty of possibilities for using an agent-based e-commerce environment, we cannot convince enough customers to implement the strategies we have presented in this paper and to produce enough sales for an assessment with statistical significance. On the other hand, agent-based simulations, as the ones employed in this paper, are precisely conceived to study situations where studying the reality is not feasible or simply too costly. However, to improve the evaluation realism, we are harvesting data sets from sales produced in electronic commerce environments. Although this data is static, we plan to preprocess it to reflect the implementation of the

strategies presented in this paper. For example, by giving distinct customer identifications in each tuple, we can study the effect of "Unique pseudonym". Other strategies will be considerably more challenging. To test "Pseudonym per preference", previously, we will have to discover the different customers' preferences based on the data set. Besides, the error occurred in this process will be propagated to the evaluation. Other strategies will require assumptions which lead the experiments, basically, to the simulation approach presented in this paper. "Fake preferences" will require to simulate fake sales and to introduce them in the data set. Besides, this assessment will always rely on the assumption that we know the learning algorithms employed by sellers agents.

Acknowledgments

This research work is supported by the Spanish Ministry of Science and Innovation in the scope of the Research Projects TSI-020302-2010-129, TIN2011-28335-C02-02 and through the Fundación Séneca within the Program 04552 / GERM / 06.

References

1. *Wine*, 2009. http://www.w3.org/TR/2004/REC-owl-guide-20040210/wine.rdf.
2. F. Abedin, K. Chao, N. Godwin, and H. Arochena. Preference ordering in agenda based multi-issue negotiation for service level agreement. In *Advanced Information Networking and Applications Workshops, 2009. WAINA'09. International Conference on*, pages 19–24. IEEE, 2009.
3. E. Aïmeur, G. Brassard, and F. S. M. Onana. Privacy-preserving physical delivery in electronic commerce. In *Proceedings of IADIS International Conference on e-Commerce*, pages 25–33, 2005.
4. R. Aydoğan and P. Yolum. Learning opponent's preferences for effective negotiation: an approach based on concept learning. *Autonomous Agents and Multiagent Systems*, 2012. In Press.
5. A. Bahrammirzaee, A. Chohra, and K. Madani. An adaptive approach for decision making tactics in automated negotiation. *Applied Intelligence*, pages In press. Published Online, DOI: 10.1007/s10489–013–0434–8, 2013.
6. J. Borking, B. Van Eck, P. Siepel, and D. Bedrijf. *Intelligent software agents: Turning a privacy threat into a privacy protector*. Registratiekamer, The Hague, 1999.
7. R. R. Bouckaert, E. Frank, M. Hall, R. Kirkby, P. Reutemann, A. Seewald, and D. Scuse. *Weka manual (3.7.1)*, June 2009. http://prdownloads.sourceforge.net/weka/WekaManual-3-7-1.pdf?download.
8. S. Buffett and B. Spencer. Learning opponents' preferences in multi-object automated negotiation. In *Proceedings of the 7th international conference on Electronic commerce*, ICEC '05, pages 300–305, New York, NY, USA, 2005. ACM.
9. D. Chaum. Security without identification: transaction systems to make big brother obsolete. *Commun. ACM*, 28:1030–1044, 1985.
10. D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In *CRYPTO '88: Proceedings on Advances in cryptology*, pages 319–327, New York, NY, USA, 1990. Springer-Verlag New York, Inc.
11. S. Clauβ, D. Kesdogan, and T. Kölsch. Privacy enhancing identity management: protection against re-identification and profiling. In *DIM '05: Proceedings of the 2005 workshop on Digital identity management*, pages 84–93, New York, NY, USA, 2005. ACM.

12. R. Dingledine, N. Mathewson, and P. Syverson. Tor: The Second-Generation Onion Router. In *13th USENIX Security Symposium*, pages 303–320, San Diego, CA, USA, 2004.
13. V. Eyharabide and A. Amandi. Ontology-based user profile learning. *Applied Intelligence*, 36(4):857–869, 2012.
14. M. Fasli. *Agent Technology For E-Commerce*. John Wiley & Sons, 2007.
15. A. Fip. *FIPA Contract Net Interaction Protocol Specification*. FIPA, 2001.
16. A. Fip. *FIPA Iterated Contract Net Interaction Protocol Specification*. FIPA, 2001.
17. S. Fischer-Hübner and H. Hedbom. Benefits of privacy-enhancing identity management. *Asia-Pacific Business Review*, 10(4):36–52, 2008.
18. S. Garfinkel. *Database nation: the death of privacy in the 21st century*. O'Reilly & Associates, Inc., Sebastopol, CA, USA, 2001.
19. J. Gwak and K. Sim. A novel method for coevolving ps-optimizing negotiation strategies using improved diversity controlling edas. *Applied Intelligence*, 38(3):384–417, 2013.
20. M. Hansen, P. Berlich, J. Camenisch, S. Clau, A. Pfitzmann, and M. Waidner. Privacy-enhancing identity management. *Information Security Technical Report*, 9(1):35 – 44, 2004.
21. M. Hansen, A. Schwartz, and A. Cooper. Privacy and identity management. *IEEE Security & Privacy*, 6(2):38–45, 2008.
22. M. Hildebrandt and S. Gutwirth. *Profiling the European Citizen: Cross-Disciplinary Perspectives*. Springer Publishing Company, Inc., 2008.
23. K. Hindriks and D. Tykhonov. Opponent modelling in automated multi-issue negotiation using bayesian learning. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems - Volume 1*, AAMAS '08, pages 331–338, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.
24. D. Hoffman, T. Novak, and M. Peralta. Building consumer trust online. *Communications of the ACM*, 42(4):80–85, 1999.
25. JADE Board. Jade security guide. `http://jade.tilab.com`, 2005.
26. J.-G. Kang, S. Kim, S.-Y. An, and S.-Y. Oh. A new approach to simultaneous localization and map building with implicit model learning using neuro evolutionary optimization. *Applied Intelligence*, 36(1):242–269, 2012.
27. K.-C. Khor, C.-Y. Ting, and S. Phon-Amnuaisuk. A cascaded classifier approach for improving detection rates on rare attack categories in network intrusion detection. *Applied Intelligence*, 36(2):320–329, 2012.
28. K.-J. Kim and H. Ahn. Simultaneous optimization of artificial neural networks for financial forecasting. *Applied Intelligence*, 36(4):887–898, June 2012.
29. L. Korba, R. Song, and G. Yee. Anonymous communications for mobile agents. In *Proceedings of the 4th International Workshop on Mobile Agents for Telecommunication Applications*, MATA '02, pages 171–181, 2002.
30. A. Krause and E. Horvitz. A utility-theoretic approach to privacy and personalization. In *Proceedings of the Twenty-Third Conference on Artificial Intelligence (AAAI-08)*, 2008.
31. J. Lee, J. Kim, and J. Y. Moon. What makes internet users visit cyber stores again? key design factors for customer loyalty. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '00, pages 305–312, New York, NY, USA, 2000. ACM.
32. C. Márquez-Vera, A. Cano, C. Romero, and S. Ventura. Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data. *Applied Intelligence*, 38(3):315–330, 2013.
33. F. Menczer, W. N. Street, N. Vishwakarma, A. E. Monge, and M. Jakobsson. Intellishopper: a proactive, personal, private shopping assistant. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 3*, AAMAS '02, pages 1001–1008, 2002.
34. A. E. Newman. Cougaar developers' guide. `http://www.cougaar.org`, 2004.
35. A. Odlyzko. Privacy, economics, and price discrimination on the internet. In *Proceedings of the 5th international conference on Electronic commerce*, ICEC '03, pages 355–366, New York, NY, USA, 2003. ACM.

36. M. Petkovic and W. Jonker, editors. *Security, Privacy and Trust in Modern Data Management (Data-Centric Systems and Applications)*. Springer-Verlag, 2007.
37. A. Pfitzmann and M. Hansen. A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. http://dud.inf.tu-dresden.de/Anon_Terminology.shtml, Aug. 2010. v0.34.
38. T. B. Quillinan, M. Warnier, M. Oey, R. Timmer, and F. Brazier. Enforcing security in the agentscape middleware. In *Proceedings of the 2008 workshop on Middleware security*, MidSec '08, pages 25–30. ACM, 2008.
39. K. Rannenberg, D. Royer, and A. Deuker, editors. *The Future of Identity in the Information Society: Challenges and Opportunities*. Springer Publishing Company, Incorporated, 2009.
40. Recursion Software Inc. Voyager security guide. `http://www.recursionsw.com/`, 2008.
41. V. Roth and M. Jalali-Sohi. Concepts and architecture of a security-centric mobile agent server. In *ISADS*, 2001.
42. E. Serrano, J. J. Gómez-Sanz, J. A. Botia, and J. Pavón. Intelligent data analysis applied to debug complex software systems. *Neurocomputing*, 72(13-15):2785 – 2795, 2009.
43. E. Serrano, A. Muñoz, and J. Botia. An approach to debug interactions in multi-agent system software tests. *Information Sciences*, 205(0):38 – 57, 2012.
44. E. Serrano, A. Quirin, J. Botia, and O. Cordón. Debugging complex software systems by means of pathfinder networks. *Information Sciences*, 180(5):561 – 583, 2010.
45. E. Serrano, M. Rovatsos, and J. Botia. A qualitative reputation system for multiagent systems with protocol-based communication. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 1*, AAMAS '12, pages 307–314, Richland, SC, 2012. International Foundation for Autonomous Agents and Multiagent Systems.
46. E. Serrano, M. Rovatsos, and J. A. Botía. Data mining agent conversations: A qualitative approach to multiagent systems analysis. *Information Sciences*, 230(0):132 – 146, 2013.
47. B.-E. Shie, P. Yu, and V. Tseng. Mining interesting user behavior patterns in mobile commerce environments. *Applied Intelligence*, 38(3):418–435, 2013.
48. C. Sierra, N. R. Jennings, P. Noriega, and S. Parsons. A framework for argumentation-based negotiation. In M. P. Singh, A. S. Rao, and M. Wooldridge, editors, *ATAL*, volume 1365 of *Lecture Notes in Computer Science*, pages 177–192. Springer, 1997.
49. H. J. Smith and S. J. Milberg. Information privacy: measuring individuals' concerns about organizational practices. *MIS Quarterly*, 20:167–196, June 1996.
50. D. Solove. A taxonomy of privacy. *University of Pennsylvania Law Review*, 154(3):477, 2006.
51. S. Spiekermann. Individual price discriminaton - an impossibility? In *International Conference for Human-Computer Interaction (CHI'2006), Workshop on Privacy and Personalization*, 2006.
52. S. Spiekermann and L. F. Cranor. Engineering privacy. *IEEE Transactions on Software Engineering*, 35(1):67–82, 2009.
53. J. M. Such. *Enhancing Privacy in Multi-agent Systems*. PhD thesis, Departament de Sistemes Informàtics i Computació, Universitat Politècnica de València, 2011.
54. J. M. Such, J. M. Alberola, A. Espinosa, and A. García-Fornes. A Group-oriented Secure Multiagent Platform. *Software: Practice and Experience*, 41(11):1289–1302, 2011.
55. J. M. Such, A. Espinosa, and A. García-Fornes. A Survey of Privacy in Multi-agent Systems. *Knowledge Engineering Review*, 2013. In press.
56. J. M. Such, A. Espinosa, A. García-Fornes, and V. Botti. Partial identities as a foundation for trust and reputation. *Engineering Applications of Artificial Intelligence*, 24(7):1128–1136, 2011.
57. J. M. Such, A. García-Fornes, A. Espinosa, and J. Bellver. Magentix2: a privacy-enhancing agent platform. *Engineering Applications of Artificial Intelligence*, 26(1):96–109, 2013.
58. TRUSTe and TNS. *2009 study: Consumer attitudes about behavioral targeting*. 2009.

59. J. Turow, J. King, C. Hoofnagle, A. Bleakley, and M. Hennessy. Americans reject tailored advertising and three activities that enable it. *Available at SSRN 1478214*, 2009.
60. S. Ugurlu and N. Erdogan. An overview of secmap secure mobile agent platform. In *Proceedings of Second International Workshop on Safety and Security in Multiagent Systems*, 2005.
61. G. van Blarkom, J. Borking, and J. Olk, editors. *Handbook of Privacy and Privacy-Enhancing Technologies: The Case of Intelligent Software Agents*. College bescherming persoonsgegevens, 2003.
62. M. Warnier and F. Brazier. Anonymity services for multi-agent systems. *Web Intelligence and Agent Systems*, 8(2):219–232, 2010.
63. A. Westin. Social and political dimensions of privacy. *Journal of Social Issues*, 59(2):431–453, 2003.
64. I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (The Morgan Kaufmann Series in Data Management Systems)*. Morgan Kaufmann, 1st edition, Oct. 1999.
65. H. Xu and S. M. Shatz. Adk: An agent development kit based on a formal design model for multi-agent systems. *Journal of Automated Software Engineering*, 10:337–365, 2003.