Computational Intelligence, Volume 25, Number 4, 2009

POYRAZ: CONTEXT-AWARE SERVICE SELECTION UNDER DECEPTION

Murat Şensoy,[1] Jie Zhang,[2] Pinar Yolum,[1] and Robin Cohen[2]

[1]*Boğaziçi University, Istanbul, Turkey*
[2]*Universtiy of Waterloo, Waterloo, Ontario, Canada*

The increasing number of service providers on the Web makes it challenging to select a provider for a specific service demand. Each service consumer has different expectations for a given service in different contexts, so the selection process should be consumer-oriented and context-dependent. Current approaches for service selection typically have consumers receive ratings of providers from other consumers, where the ratings reflect the consumers' overall subjective opinions. This may be misleading if consumers have different contexts and satisfaction criteria. In this paper, we propose that consumers objectively record their experiences, using an ontology to capture subtle details. This can then be interpreted by consumers according to their own criteria and contexts. We then integrate a method for addressing consumers who lie about their experiences, filtering them out during service selection. We demonstrate the value of our approach through experiments comparing our model with three recent rating-based service selection models. Our experiments show that using the proposed approach, service consumers can select the service providers for their needs more accurately even if the consumers have different criteria, they change the contexts of their service demands over time, or a significant portion of them are liars.

Key words: electronic commerce, service selection, trust and reputation, semantic Web.

1. INTRODUCTION

The number of service providers on the Web is increasing dramatically (Paolucci and Sycara 2004). As a result, a service consumer is confronted with a large number of service providers that advertise the same service. This is particularly relevant in the context of electronic commerce, for instance. Although those providers advertise their service offerings on the Web, there is no guarantee that they comply with what they advertise. This is a natural result of the fact that the Web is not operated by a central authority that can monitor all participants' activities and ensure that everyone acts properly.

Even if the providers produce services that comply with their service definitions, those definitions usually cover only the functional properties of the services they offer. However, the satisfaction of consumers is not only related to functional properties of the services they get but also non-functional properties such as service quality (Zeng et al. 2004). Therefore, only considering the advertised description of a service, it is difficult for consumers to select satisfactory service providers for a given service demand.

Moreover, for exactly the same supplied service, each consumer may have a different degree of satisfaction, because each service consumer has different expectations and satisfaction criteria for a given service in different contexts. Therefore, while selecting an appropriate service provider among alternatives for a service consumer, one should carefully consider the satisfaction criteria and the context of the consumer.

Although semantic Web technologies such as ontologies and ontological reasoning (Feigenbaum et al. 2007) are promising for intelligent, consumer-oriented, and context-aware selection of service providers, the most widely used service selection approaches depend on capturing and manipulating ratings (Jøsang, Ismail, and Boyd 2007). In rating-based approaches, the consumers rate the service providers and share their ratings with other consumers. Then, the shared ratings are aggregated to determine the most satisfactory service providers. Rating-based approaches suffer from two weaknesses. First, ratings do not

Address correspondence to Murat Şensoy, Department of Computer Engineering, Boğaziçi University, Istanbul, Turkey; e-mail: sensoymu@boun.edu.tr

carry any semantic information. That is, just looking at a rating, one cannot understand the rationale of the rating. Second, ratings reflect the satisfaction criteria and taste of the raters. If the taste of a consumer is highly different than that of the raters, the ratings may seriously mislead the consumer (Liang and Shi 2008). For example, a service consumer may give a low rating to a service provider who delivers a book 2 days late. If the delivery date is not significant for a second service consumer, the first service consumer's low rating will not be significant either.

Semantic Web technologies provide a common framework that allows semantic data to be shared (Feigenbaum et al. 2007). For example, using an ontology, it is possible to semantically describe the interactions between a consumer and a provider in detail. Then, this representation of past dealings with the provider can easily be interpreted by other consumers. Accordingly, we previously developed an approach for distributed service selection that allows consumers to represent their *experiences* with the service providers using ontologies (Şensoy and Yolum 2007). An experience captures the outcome of an interaction between a customer and a provider, and can be thought of as a record of what service the customer has requested and received in return. In this way, experience-based approaches allow the objective facts of the experiences (other than subjective opinions, i.e., ratings) to be communicated to the other party. A consumer who receives other consumers' particular experiences can interpret what they have experienced with the providers and evaluate the providers using her own satisfaction criteria and context. We experimentally showed that experience-based approaches outperform rating-based approaches in terms of the achieved satisfaction in reliable environments where consumers honestly share their experiences (Şensoy and Yolum 2007).

Our previous work assumes that consumers always exchange their experiences honestly. However, in many settings, a consumer may prefer to be dishonest about their past dealings with providers. For example, consumers may provide untruthful experiences to promote the providers. This is referred to as "ballot stuffing" (Dellarocas 2000). Consumers may also cooperate with other providers to drive a provider out of the system. This is referred to as "bad-mouthing." We show that in deceptive environments where there are liars among the consumers, the experience-based service selection significantly fails. Zhang and Cohen (2006) have proposed an approach to allow consumers to model the trustworthiness of other consumers. This centralized rating-based method combines consumers' personal observations with others' information about providers and public knowledge of the others held by the system. In our work, we adapt this approach to evaluate the trustworthiness of consumers, in a distributed setting on the basis of the consumers' shared *experiences*, which is context-aware and able to handle consumer subjectivity.

In summary, we propose POYRAZ, an integrated approach for context-aware service selection in deceptive environments. POYRAZ effectively combines (1) a service selection engine that makes context-aware service selections using the shared consumer experiences and (2) an information filtering module that computes trustworthiness of the consumers and identifies deceptive experiences. This module filters out deceptive experiences, before they are used for the selection of service providers. We evaluate the performance of POYRAZ using extensive experiments in different settings. We compare it with three recent rating-based service selection approaches from the literature. Our experiments show that using POYRAZ, service consumers can successfully select satisfactory service providers even if a significant ratio of consumers are liars, and even if the satisfaction criteria and the context of consumers vary considerably over time. Moreover, POYRAZ significantly outperforms the rating-based approaches in those settings.

The rest of this paper is organized as follows. Section 2 explains our approach for representing experiences using an ontology and making service selection using those experiences in depth. Section 3 presents our method for the computation of consumers' trustworthiness and filtering out deceptive experiences received from untrustworthy consumers. Section 4 experimentally evaluates our approach for context-aware service selection with comparisons to well-known rating-based service selection approaches. Section 5 discusses our work with reference to the literature and presents an overview of the significance of our contributions. Last, we offer concluding remarks and outline directions for further research in Section 6.

2. CONTEXT-AWARE SERVICE SELECTION

Consider a multiagent system in which consumer agents (service consumers) help their users to find useful service providers. We assume that those service consumers know the satisfaction criteria of their users for a specific service demand. In this setting, the main task of the service consumers is to search for service providers to handle their users' service demands. For this purpose, the consumers record their interactions with the service providers objectively in detail within an experience structure and then share their experiences (Şensoy and Yolum 2007). An experience structure contains a consumer's service demand and the provided service in response to this service demand.[1] Actually, an experience expresses the interaction experienced between the consumer and the provider regarding a specific service demand at a specific time. So, any consumer receiving an experience can evaluate the service provider according to its own criteria using the objective data in the experience. This approach overcomes the subjectiveness of the rating-based approaches. However, expression of experiences requires the representational power of ontologies.

2.1. Experience Ontology

In order to express their experiences with the service providers, service consumers use a common Web ontology language (OWL) for a specified service domain. This ontology covers the fundamental concepts (such as demand, service, commitment, and experience), which exist in the base level ontology and domain specific concepts and properties, which exist in the domain level ontology. Using these concepts and properties, a service consumer can express its service demands and experiences.

The base level ontology (Figure 1) consists of the domain-independent infrastructure of the experience ontology. The main class in the base level ontology is the *Experience* class. Instances of this class represent the experiences of service consumers in the system. As in real life, an experience in the ontology contains information about what a service consumer has requested from a service provider and what the service consumer has received at the end. To conceptualize the service demand and the received service of the consumer, *Demand* and *Service* classes are included in the base level ontology. Both the demand and the supplied service concepts are descriptions of a service for a specific domain and hence share a number of properties. These shared properties are captured in the *Description* class in the base level ontology. The domain level ontology contains extensions to this class. Domain-dependent properties of the *Description* class can be used to describe service demands, supplied services, responsibilities, and fulfillments of sides during transactions. These properties are shown in domain level ontology.

Each *Description* class contains an owner and a date field. For a demand, the owner is a service consumer and for a service, the owner is a service provider. The date value keeps

[1] Henceforth in this paper, an *experience structure* is simply referred to as an *experience*.

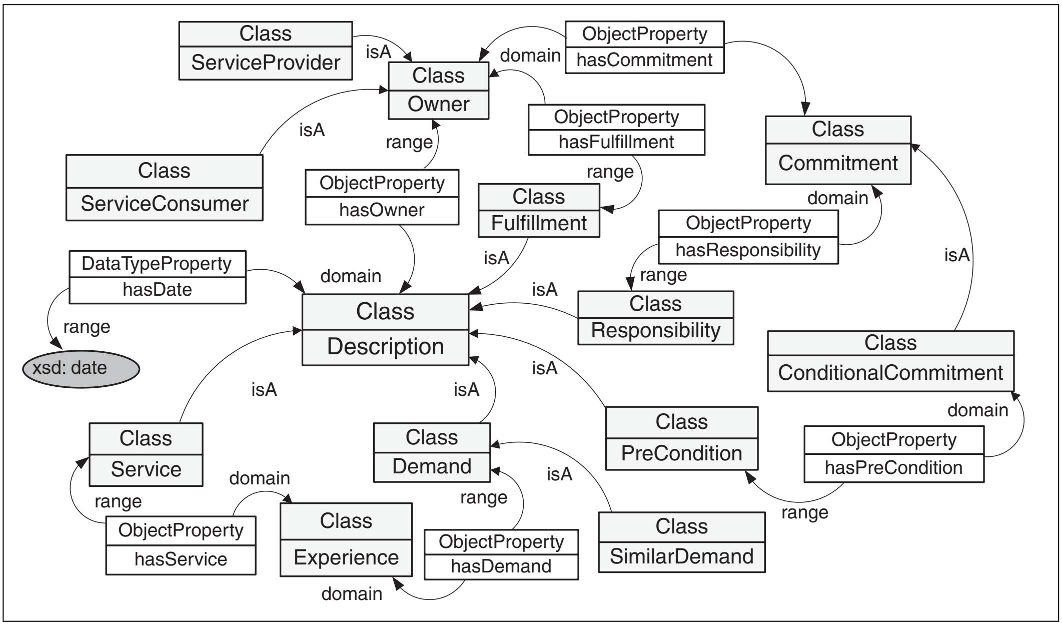

FIGURE 1. Base-level ontology.

the date of demanded service or the provided service. An owner may have commitments toward others to carry out responsibilities (Singh 1999). A commitment always has an instance of responsibility. This means that the owner of the commitment is responsible for the realization of conditions described in the responsibility instance. Example 1 demonstrates a simple responsibility instance. *Commitment* and *Responsibility* classes are used to express commitments and responsibilities, respectively, in the experience ontology. Fulfillments are accomplishments of responsibilities and are denoted with the *Fulfillment* class. Owners of responsibilities or fulfillments can be service consumers or providers depending on the context.

Example 1. Consider a service provider who is responsible for delivering particular goods to New York City with a shopping cost of $5. In the ontology, this can be represented as an instance of a *Commitment* class, where the instance of the *Responsibility* of the commitment has a *toLocation* property referring to New York City and has a *hasShipmentCost* property referring to $5.

Transactions between the consumers and providers are usually based on business contracts. The contracts can be represented by conditional commitments. Unlike commitments, conditional commitments have preconditions. For example, a conditional commitment $CC(X, Y, P, Q)$ denotes that if the precondition P is carried out by Y, X will be committed to carry out responsibility Q. In this definition, Y is the owner of the precondition and X is the owner of responsibility. *ConditionalCommitment* and *Precondition* classes are used in the ontology to specify conditional commitments and preconditions. Conditional commitments can be used to represent contracts and offers made by service consumers and providers. An example case is demonstrated in Example 2.

Example 2. A service consumer can offer to pay an additional $100 for 1 week early delivery. If the provider makes the shipment 1 week early, the consumer is committed to pay

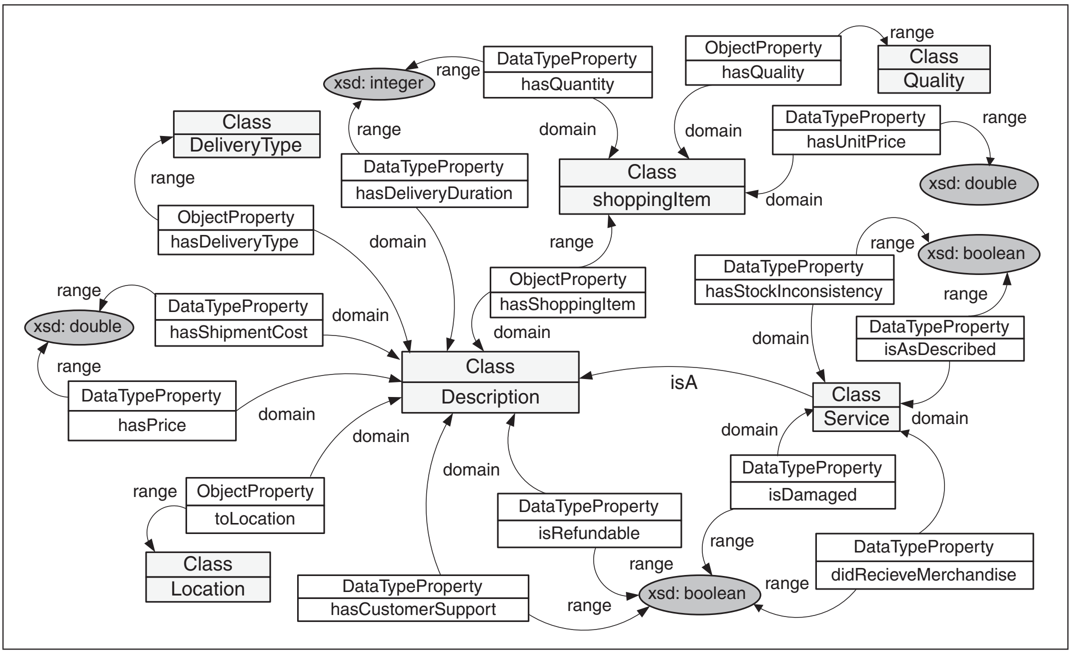

FIGURE 2. Domain-level ontology for online shopping.

$1,100 for a product whose actual value is only $1,000. Service providers can also make offers using conditional commitments.

As the base-level ontology deals only with domain-independent concepts, a second ontology is necessary to capture domain-dependent concepts and properties. The domain-level ontology is developed for this purpose. The core class of domain-level ontology is the *Description* class, which refers to the same *Description* class in the base-level ontology. Domain-specific properties of *Description* class are used to describe service demands, supplied services, responsibilities, and fulfillments of parties during transactions. A domain-level ontology for online shopping is shown in Figure 2. This ontology contains domain-specific concepts such as *ShoppingItem*, *Location*, *DeliveryType*, and *Quality*, as well as domain-specific properties such as *hasShoppingItem*, *toLocation*, *hasDeliveryType*, *hasDeliveryDuration*, *hasShipmentCost*, and *hasPrice*. Those concepts and properties are used to describe consumers' experiences in the online shopping domain.

2.2. Decision Making Using Experiences

Service consumers maintain, exchange, and interpret experiences related to the providers. These experiences are expressed using the OWL ontology in Section 2.1. Therefore, they can be interpreted easily by the agents using an OWL reasoner such as *Pellet* (Sirin et al. 2007). In Figure 3, an example experience is shown. This experience is explained in Example 3.

Example 3. In her experience (represented in Figure 3), the buyer states that she ordered an IBM ThinkPad T60 notebook from a seller named *TechnoShop* on October 15, 2007. She requested the merchandise to be delivered to New York within 14 days. The seller received $700 for the product and delivered the merchandise within 7 days without requesting any

```
<owlx:Individual owlx:name="ExperienceInstance">
  <owlx:type owlx:name="Experience" />
  <owlx:ObjectPropertyValue owlx:property="hasDemand">
    <owlx:Individual owlx:name="demandInstance" />
  </owlx:ObjectPropertyValue>
  <owlx:ObjectPropertyValue owlx:property="hasService">
    <owlx:Individual owlx:name="serviceInstance" />
  </owlx:ObjectPropertyValue>
</owlx:Individual>
```

```
<owlx:Individual owlx:name="demandInstance">
  <owlx:type owlx:name="Demand" />
  <owlx:ObjectPropertyValue owlx:property="hasOwner">
    <owlx:Individual owlx:name="MuratSensoy" />
  </owlx:ObjectPropertyValue>
  <owlx:DataPropertyValue owlx:property="hasDate">
    <owlx:DataValue owlx:datatype="&xsd;Date">2007-10-15</owlx:DataValue>
  </owlx:DataPropertyValue>
  <owlx:ObjectPropertyValue owlx:property="hasShoppingItem">
    <owlx:Individual owlx:name="#IBM_ThinkPad_T60" />
  </owlx:ObjectPropertyValue>
  <owlx:ObjectPropertyValue owlx:property="toLocation">
    <owlx:Individual owlx:name="NewYork" />
  </owlx:ObjectPropertyValue>
  <owlx:DataPropertyValue owlx:property="hasDeliveryDuration">
    <owlx:DataValue owlx:datatype="&xsd;Integer">14</owlx:DataValue>
  </owlx:DataPropertyValue>
</owlx:Individual>
```

```
<owlx:Individual owlx:name="serviceInstance">
  <owlx:type owlx:name="Service" />
  <owlx:ObjectPropertyValue owlx:property="hasOwner">
    <owlx:Individual owlx:name="TechnoShop" />
  </owlx:ObjectPropertyValue>
  <owlx:ObjectPropertyValue owlx:property="hasShoppingItem">
    <owlx:Individual owlx:name="#IBM_ThinkPad_T60" />
  </owlx:ObjectPropertyValue>
  <owlx:DataPropertyValue owlx:property="hasDeliveryDuration">
    <owlx:DataValue owlx:datatype="&xsd;Integer">7</owlx:DataValue>
  </owlx:DataPropertyValue>
  <owlx:DataPropertyValue owlx:property="recivedMerchandise">
    <owlx:DataValue owlx:datatype="&xsd;boolean">true</owlx:DataValue>
  </owlx:DataPropertyValue>
  <owlx:DataPropertyValue owlx:property="isRefundable">
    <owlx:DataValue owlx:datatype="&xsd;boolean">false</owlx:DataValue>
  </owlx:DataPropertyValue>
  <owlx:DataPropertyValue owlx:property="hasCustomerSupport">
    <owlx:DataValue owlx:datatype="&xsd;boolean">false</owlx:DataValue>
  </owlx:DataPropertyValue>
  <owlx:DataPropertyValue owlx:property="hasShipingCost">
    <owlx:DataValue owlx:datatype="&xsd;Integer">0</owlx:DataValue>
  </owlx:DataPropertyValue>
  <owlx:DataPropertyValue owlx:property="hasPrice">
    <owlx:DataValue owlx:datatype="&xsd;Integer">700</owlx:DataValue>
  </owlx:DataPropertyValue>
</owlx:Individual>
```

FIGURE 3. An experience that is about buying a notebook from a seller named *TechnoShop*.

extra money for shipping. However, the delivered product was not refundable and *TechnoShop* did not provide any customer support.

When a consumer has a new service demand and only a few or no direct previous interactions with the service providers, it needs to collect information about the service providers from other consumer agents. This information is used to compute the expected behavior of the providers for the current service demand of the consumer. Behaviors of the providers may change considerably in different contexts. For example, while a provider delivers bicycles on time, it may not deliver cars without any delay. This implies that a consumer may model the behaviors of service providers with respect to its specific service demand. Therefore, the consumers who have had similar service demands in the past may provide more useful information about the providers. Those consumers are contacted to provide the information related to the service providers.

In rating-based service selection approaches, the collected information is the ratings of providers. Ratings reflect the subjective opinion of the raters. Therefore, ratings may mislead the consumers in the cases where the satisfaction criteria of the consumers using these ratings are different from the satisfaction criteria of those that provide the ratings (as shown in Example 4). Unlike ratings, experiences do not reflect the subjective opinion of their creators. Therefore, any consumer receiving an experience can evaluate the service provider according to its own criteria using the objective data in the experience.

Example 4. Consider the experience in Figure 3 and assume that there are two different consumers (*Bob* and *Lucy*) who received this experience. For Bob, delivery duration and price are crucial whereas customer support or being refundable are not important. On the other hand, for Lucy, being refundable and having customer support are indispensable. Therefore, for Bob, *TechnoShop* is a very good provider and deserves a good rating, because it delivers products within 1 week without requesting any extra money. However, for Lucy, *TechnoShop* is not preferable. However, by plain ratings, Bob's positive ratings of *TechnoShop* would have misled Lucy.

In experience-based service selection, first a consumer collects related experiences from other consumers. For example, if the consumer needs to buy a notebook, it searches for the experiences that are related to "buying notebooks" (for details, see Appendix A). After collecting related experiences, the consumer evaluates each experience using its satisfaction criteria. Each consumer has an internal taste function F_{taste} (namely satisfaction criteria) to evaluate its transactions with the service providers in the context of its service demands. This function takes as its argument an experience (a pair of service demanded, service received) and returns as its output $\{0, 1\}$, where 0 means that the received service within a transaction is not satisfactory for the consumer while 1 means that it is satisfactory. In real life, the taste of a consumer may change over time. Hence, this function should be time dependent. We assume that taste function of the consumer is unknown to other consumers. In a real-life application, a consumer agent can easily elicit its taste function from its human user using a user interface. Once the consumer has the taste function, it can easily compute its expected level of satisfaction for a specific transaction given the service demand and the supplied service within the transaction. Hence, using the taste function, the consumer can also interpret an experience and compute its level of satisfaction using the data in the experience. In other words, the consumer can produce its expected level of satisfaction for the experience by asking itself how satisfied it would be, had it lived the experience under consideration.

Using the collected experiences about service providers, a consumer can model the service providers to estimate which of the providers produce a satisfactory service for a specific service demand. For this purpose, the consumer uses a machine learning technique, parametric classification (Duda, Hart, and Stork 2001), as follows. Demand and service specifications within experiences are received in the form of ontologies, but then they are converted into the internal representation of the service consumer. Demand and commitment information in each experience is represented as a vector. Each field in this vector is extracted from the experience ontology. These fields correspond to property values in the experience ontology such as service price. Supplied service for this demand is classified as satisfied or dissatisfied with respect to satisfaction criteria of the consumer using the taste function and ontological reasoning (Pan 2007). The (*vector*, *class*) pairs are used as training set, where possible classes are satisfied and dissatisfied. For each class, covariance and mean are extracted from the training set. Then, a discriminant function is defined to compute the probability of satisfaction (Duda et al. 2001). The service consumer performs this computation for every service provider and chooses the provider with the highest satisfaction probability.

Equations (1)–(3) formulate this computation using a Gaussian distribution function. In these equations, C_i refers to the ith class and d represents the number of dimensions of the demand vector X. Note that there are two classes: the first class is *satisfied* and the second class is *dissatisfied*. For the ith class, mean and covariance are represented by μ_i and Σ_i, respectively. The mean μ_i is a vector with d dimensions and refers to the mean of the demand vectors in C_i. That is, each element of μ_i refers to the mean of the corresponding dimension of the demand vectors in C_i. The covariance Σ_i is a matrix of $d \times d$ dimensions and each element of it refers to the correlation between the corresponding dimensions of the demand vectors in C_i.

Equation (1) formulates the class likelihood $p(X \mid C_i)$: the probability that the demand X is observed in class C_i. In the equation, T is the transpose operator. Using Bayes's Rule, equation (2) formulates the posterior probability $p(C_i \mid X)$: the probability that the demand X is in class C_i. In equation (2), $p(X)$ refers to the probability that demand X is observed and it is computed as $p(X) = p(X \mid C_1) + p(X \mid C_2)$ in this case. Similarly, $p(C_i)$ refers to the prior probability that the class C_i is observed. Last, the discriminant function for the ith class, $g_i(X)$, is formulated as in equation (3) (Duda et al. 2001). In this way, we are performing a maximum posteriori estimation. The higher the computed $g_1(X)$ value is, the more likely the provider under consideration satisfies demand X:

$$p(X \mid C_i) = \frac{\exp\left[-\frac{1}{2}(X-\mu_i)^T \Sigma_i^{-1}(X-\mu_i)\right]}{(2\pi)^{2/d}\,|\Sigma_i|^{1/2}}, \tag{1}$$

$$p(C_i \mid X) = \frac{p(X \mid C_i)p(C_i)}{p(X)}, \tag{2}$$

$$g_i(X) = \log[p(C_i \mid X)] + \log[p(X)] = \log[p(X \mid C_i)] + \log[p(C_i)]. \tag{3}$$

Consider that Bob in Example 4 wants to buy a notebook. For this purpose, first he collects experiences about the notebook providers and then estimates the probability of satisfaction for each provider as described above. In this example, Bob needs to compute the probability that *TechnoShop* produces a satisfactory service. Initially, Bob uses his satisfaction criteria to evaluate the supplied services within the collected experiences about *TechnoShop*. He labels each experience as satisfied or dissatisfied. Using a Gaussian distribution function, Bob estimates the probabilities that his current demand is observed among the satisfied demands and dissatisfied demands as in equation (1). Those probabilities are denoted as $p(X \mid C_1)$ and $p(X \mid C_2)$, respectively, where X is the data vector representing Bob's current demand about buying a notebook. Then, using Bayes' rule, Bob estimates the probability that *TechnoShop* produces a satisfactory service given his current service demand, denoted as $p(C_1 \mid X)$. Last, Bob calculates the discriminant function $g_1(X)$ to quantify the preferability of *TechnoShop* using equation (3) and uses this value to decide about *TechnoShop*.

Until now, we assume that all the consumers share their experiences honestly. Because of personal or commercial reasons, some malicious consumers may want to defame or advertise some providers by producing and propagating deceptive experiences. If the consumers in the system cannot differentiate between truthful and deceptive experiences, deceptive experiences may mislead them. This situation strongly imposes the requirement of using a mechanism to filter out deceptive experiences during service selection.

3. FILTERING OUT DECEPTIVE EXPERIENCES

In this section, we describe how deceptive experiences are determined and filtered out during service selection. From now on, we call consumers who share their experiences or ratings with others "advisors." Different approaches have been proposed for determining and filtering out deceptive information from advisors during service selection, e.g., TRAVOS (Teacy et al. 2006) and the iterated filtering approach of beta reputation system (BRS) (Whitby, Jøsang, and Indulska 2005). A brief description of this research can be found in Section 4. These approaches usually use limited sources of information. For example, TRAVOS only makes use of consumers' personal observations (private information) for evaluating trustworthiness of an advisor, while BRS uses only ratings from others (public information) to filter out unreliable ratings. In general, approaches using public information are designed to be centralized and assume that the majority of advisors are honest. Approaches using only personal observations may fail in settings where consumers do not have enough personal observations.

We propose an approach for a consumer to estimate the trustworthiness of an advisor by combining the two different sources of information: private and public credits of the advisor. The private credit of the advisor is calculated by the consumer, based on the experiences the advisor supplies of providers with whom the consumer has already had some interaction. If private credit cannot be calculated with confidence, a public credit is calculated, based on the advisor's experiences with all providers in the environment. A weighted combination of the private and the public credits is derived, based on the estimated reliability of the private credit value. This combined value then represents the trustworthiness of the advisor. After that, the experiences received from the less trustworthy consumers are finally regarded as deceptive and filtered out during service selection. Note that during those calculations, we only consider the experiences related to the current demand of the consumer. This is because only those experiences are used for the service selection, so the context of those experiences is the same as the current one. In other words, trustworthiness of advisors is calculated in a context-dependent way. This enables an advisor to be regarded as trustworthy in one context while the advisor may be regarded as untrustworthy in another context.

3.1. Private Credit of Advisors

Our approach allows a consumer C to evaluate the private credit of an advisor A by comparing their experiences for their commonly encountered providers $\{P_0, P_1, \ldots, P_m\}$. For each of the commonly encountered provider P_i, A has the experience vector E_{A,P_i} and C has the experience vector E_{C,P_i}. The experiences in E_{A,P_i} and E_{C,P_i} are ordered according to their recency. The experiences are then partitioned into different elemental time windows. The length of an elemental time window may be fixed (e.g., 3 days) or adapted by the frequency of the transactions with the provider P_i, similar to the way proposed in (Dellarocas 2000), where the length is smaller when the frequency of the transactions is high, and larger when the frequency is low. It should also be considerably small so that there is no need to worry about the changes of providers' behavior within each elemental time window.

We define a pair of experiences (e_{A,P_i}, e_{C,P_i}), such that e_{A,P_i} is one of the experiences in E_{A,P_i}, e_{C,P_i} is one of the experiences in E_{C,P_i}, and e_{A,P_i} *corresponds* to e_{C,P_i}. Two experiences, e_{A,P_i} and e_{C,P_i}, are correspondent only if the experience e_{C,P_i} is the most recent experience in its time window, and the experience e_{A,P_i} is the closest and prior to the experience e_{C,P_i}. We consider experiences provided by C after those by A, in order to incorporate into C's experiences anything learned from A, before taking an action. According to the solution proposed in Zacharia, Moukas, and Maes (1999), by keeping only the most recent experiences, we can avoid the issue of advisors' "flooding" the system. No matter how many experiences are provided by one advisor in a time window, we only keep the most recent one. Then, we count the number of experience pairs for P_i, denoted as N_{P_i}. The total number of experience pairs for all commonly encountered providers (N_{all}) will be calculated by summing up the number of experience pairs for each commonly encountered provider as follows:

$$N_{\text{all}} = \sum_{i=0}^{m} N_{P_i}. \tag{4}$$

For each pair of experience (e_{A,P_i}, e_{C,P_i}), the consumer C converts e_{A,P_i} and e_{C,P_i} to its satisfaction levels based on its own taste function F^C_{taste} as follows:

$$l_{A,P_i} = F^C_{\text{taste}}(e_{A,P_i}), \qquad l_{C,P_i} = F^C_{\text{taste}}(e_{C,P_i}). \tag{5}$$

Note that for the purpose of simplicity, we assume the satisfaction level is binary (satisfied, dissatisfied) in the current work. Possible ways of extending our approach to accept satisfaction levels other than binary ones will be investigated as future work. We define the experience pair (e_{A,P_i}, e_{C,P_i}) as a positive experience pair if l_{A,P_i} is the same as l_{C,P_i}. Otherwise, the pair is called as a negative experience pair.

We examine experience pairs for all commonly encountered providers. Suppose there are N_p number of positive pairs. The number of negative pairs will be $N_{\text{all}} - N_p$. The private credit of the advisor A is estimated as the probability that A will provide truthful experiences to C. By truthful experiences, we mean the experiences whose converted satisfaction levels are the same as the ones of the personal experiences of C. Because there is only incomplete information about the advisor, the best way of estimating this probability is to use the expected value of the probability. The expected value of a continuous random variable is dependent on a probability density function, which is used to model the probability that a variable will have a certain value. Because of its flexibility and the fact that it is the conjugate prior for distributions of binary events (Russell and Norvig 2002), the beta family of probability density functions is commonly used to represent probability distributions of binary events [see, e.g., the generalized trust models BRS (Jøsang and Ismail 2002) and TRAVOS (Teacy et al. 2006)]. Therefore, the private credit of A can be calculated as follows:[2]

$$\alpha = N_p + 1, \qquad \beta = N_{\text{all}} - N_p + 1,$$
$$R_{\text{pri}}(A) = E[Pr(A)] = \frac{\alpha}{\alpha + \beta}, \tag{6}$$

[2] Note that we assume a uniform distribution for the initial prior distribution.

where $Pr(A)$ is the probability that A will provide truthful experiences to C, and $E[Pr(A)]$ is the expected value of this probability variable.

3.2. Public Credit of Advisors

If the consumer C has a few or no personal experiences about the providers that the advisor A has experience with, then private credit of A cannot be computed by C with confidence. In this case, the consumer C calculates A's public credit in addition to its private credit. For this purpose, experiences given by A are examined to determine if they are consistent with the majority of the experiences given by the other advisors for the same providers. Consistency of an experience e_{A,P_i} with the majority is computed as follows. First, the consumer C determines the experiences provided by other advisors about the same provider, P_i. Suppose that n other advisors $(A_0, \ldots, A_{n-1})$ also have given C their experiences about the provider P_i. Let one of the experiences given by the advisor A_j be e_{A_j,P_i}, where $0 \leq j < n$ and e_{A_j,P_i} correspond to e_{A,P_i}. In other words, similar to the calculation of private credit, e_{A,P_i} and e_{A_j,P_i} are within the same time window, e_{A_j,P_i} is prior to e_{A,P_i}, and they are the most recent experiences in the corresponding time window. Hence, we guarantee that the conflicts between the experiences in our calculations are not due to the behavior change of the providers, but instead due to dishonest reporting. Second, those experiences provided by other advisors about P_i are converted to the consumer C's satisfaction levels, using equation (5). In this case, we use 1 to represent satisfactory experiences and 0 to represent dissatisfactory experiences. Then, we calculate the average satisfaction level (avg) as in equation (7). The experience e_{A,P_i} of the advisor A is considered a consistent experience if $|F^C_{\text{taste}}(e_{A,P_i}) - avg| \leq \phi$; otherwise, e_{A,P_i} is considered as an inconsistent experience. In our calculations, $0 < \phi < 0.5$ is the maximum acceptable deviation from the majority:

$$avg = \frac{\sum_{j=0}^{n-1} F^C_{\text{taste}}(e_{A_j,P_i})}{n}. \tag{7}$$

Suppose that the advisor A provides in total N'_{all} experiences for the current demand of C. If there are N_c consistent experiences among those experiences, the inconsistent experiences provided by A will be $N'_{\text{all}} - N_c$. In a similar way as estimating the private credit, the public credit of the advisor A is estimated as the probability that A will provide consistent experiences for the current demand of C. It can be calculated as follows:

$$\alpha' = N_c + 1, \qquad \beta' = N'_{\text{all}} - N_c + 1,$$

$$R_{\text{pub}}(A) = \frac{\alpha'}{\alpha' + \beta'}. \tag{8}$$

This indicates that public credit of an advisor is high as long as it gives experiences consistent with the experiences of the majority.

3.3. Trustworthiness of Advisors

In order to estimate the trustworthiness of the advisor A, we combine the private credit and the public credit values. The private credit and the public credit values are assigned different weights. The weights are determined by the reliability of the estimated private credit value. For this purpose, we first determine the minimum number of experience pairs needed for C to be confident about the calculated private credit of A. The Chernoff Bound theorem (Mui, Mohtashemi, and Halberstadt 2002) provides a bound for the probability that the estimation error of private credit exceeds a threshold, given the number of pairs. Accordingly, the minimum number of pairs can be determined by an acceptable level of error and a confidence measurement as follows:

$$N_{\mathrm{min}} = -\frac{1}{2\varepsilon^2}\ln\frac{1-\gamma}{2}, \tag{9}$$

where $\varepsilon \in (0, 1)$ is the maximal level of error that can be accepted by C, and $\gamma \in (0, 1)$ is the level of confidence consumer C would like to attain. If the total number of experience pairs used for the calculation of the private credit is larger than or equal to N_{min}, the consumer C is confident about the calculated private credit value. Hence, this value is used as the trustworthiness of A. However, if the used experience pairs are less than N_{min}, the consumer C combines the private and the public credit values as a weighted sum. The weight (or reliability) of the private credit value can be measured as follows:

$$w = \begin{cases} \dfrac{N_{\mathrm{all}}}{N_{\mathrm{min}}} & \text{if } N_{\mathrm{all}} < N_{\mathrm{min}}; \\ 1 & \text{otherwise.} \end{cases} \tag{10}$$

The trustworthiness of A is calculated by combining the private and public credit values as follows:

$$Tr(A) = wR_{\mathrm{pri}}(A) + (1 - w)R_{\mathrm{pub}}(A). \tag{11}$$

4. EVALUATION

In order to demonstrate the performance of the proposed methods, we implement a simulator and conduct simulations on it. The simulator is implemented in Java. KAON2[3] is used as the OWL-DL reasoner. In our experiments, there are various settings and for each setting, simulations are repeated 10 times in order to increase the reliability. We average the performance of various approaches throughout the simulations and their mean values are reported in the figures. Although we estimate and report the mean values, these mean values may not reflect the true mean values. The reason is that the estimated mean values may vary from sample to sample. Hence, we compute a confidence interval that generates a lower and upper limit for the mean values. This interval estimate gives an indication of how much uncertainty is in our estimate of the true mean values. The narrower the interval is, the more precise our estimate is. In order to compute confidence intervals of the mean values, a t-test can be used when the number of samples is small (e.g., 10 samples). Therefore, our simulation results are analyzed with a t-test for a 95% confidence interval, as suggested in (Montgomery 2001). Our tests show that with a 95% probability, the reported mean values for the average percent of success in service selection deviates at most 3%. This implies that our results statistically significant and our conclusions may not change much for different runs of simulations.

The main purpose of our simulations is to measure the performance of our approach in selecting an appropriate service provider in different settings. In the implementation of POYRAZ, we set the maximum acceptable deviation from the majority $\phi = 0.1$, the acceptable level of error $\varepsilon = 0.4$ and the confidence measurement $\gamma = 0.6$ during the

[3] http://kaon2.semanticweb.org

calculations of advisors' trust values. After calculating the trust values, we regard an advisor as a liar if its trustworthiness is less than 0.5. We also implement different service selection approaches from the literature and compare them with POYRAZ. Those approaches are explained briefly in the next section.

4.1. Service Selection Approaches for Benchmarks

There are many rating-based service selection approaches in the literature. We use three of those approaches to make benchmark comparisons with our approach. Those approaches are explained briefly below. In order to make more reliable comparisons, the rating-based approaches and POYRAZ use the same information sources in our experiments. While POYRAZ uses *experiences*, the rating-based approaches use *ratings* from the same sources (advisors).

1. *FIRE: An integrated trust and reputation model for open multiagent systems.* It is a trust and reputation model consisting of four components (Huynh, Jennings, and Shadbolt 2004): interaction trust, witness reputation, role-based trust, and certified reputation. Role-based trust and certified reputation components are not related to our work. Hence, in this work, we only consider the interaction trust and witness reputation components. The interaction trust component models a consumer's trust of a provider using only the direct interactions between the consumer and the provider. Here, FIRE uses the direct trust component of another well-known trust and reputation system, REGRET (Sabater and Sierra 2001). On the other hand, the witness reputation component uses only the ratings from other consumers to compute the reputation of the provider. In FIRE, each rating is a tuple in the following form: $r = (c; p; i; t; v)$; where c and p are the consumer and the provider that participated in the interaction i, respectively, and v is the rating c gave p for the term t (e.g., price, quality, and delivery). The range of v is $[-1, +1]$, where -1 means absolutely negative, +1 means absolutely positive, and 0 means neutral or uncertain. In this way, FIRE enables consumers to rate each attribute of a service independently. Unfortunately, FIRE does not have any mechanism for filtering out unfair ratings. After computing the direct trust and the witness reputation, FIRE calculates the overall trust of the provider as a weighted sum of those values.
2. *Beta reputation system.* The beta reputation system (BRS) is proposed by Jøsang and Ismail (2002). It estimates reputations of service providers using a probabilistic model. This model is based on the beta probability density function, which can be used to represent probability distributions of binary events. In this approach, consumers propagate their ratings about providers. A rating of the consumer c to the provider p is in the form of $r = [g, b]$, where g is the number of c's good interactions with p and b is the number of c's bad interactions with p. Ratings from different consumers about the same provider are combined by simply computing the total number of good interactions and the total number of bad interactions with the provider. These two numbers are used to compute the parameters of a beta distribution function that represents the reputation of the provider. To handle unfair ratings provided by other consumers (advisors), Whitby et al. extend the BRS to filter out those ratings that do not comply with the significant majority of the ratings by using an *iterated filtering approach* (Whitby et al. 2005). Hence, this approach assumes that significant majority of the advisors honestly share their ratings.
3. *TRAVOS: Trust and reputation in the context of inaccurate information sources.* This approach is proposed by Teacy et al. (2006). Similar to BRS, it uses beta probability density functions to compute consumers' trust on service providers. The main difference between BRS and TRAVOS is the way they filter out unfair ratings. While BRS uses

TABLE 1. Dimensions of Service Space and Their Ranges

Dimension name	Type	Range
hasShoppingItem	Integer	1–1,000
toLocation	Integer	1–100
hasDeliveryType	Integer	1–6
hasDeliveryDuration	Integer	1–60
hasShipmentCost	Double	0–250
hasPrice	Double	10–11,000
hasUnitPrice	Double	1–100
hasQuantity	Integer	1–100
hasQuality	Integer	1–10
isRefundable	Boolean	0–1
hasConsumerSupport	Boolean	0–1
didReceiveMerchandise	Boolean	0–1
hasStockInconsistency	Boolean	0–1
isAsDescribed	Boolean	0–1
isDamaged	Boolean	0–1

the majority of ratings to filter out unfair ratings about the providers, TRAVOS uses the personal observations about those providers to detect and filter out unfair ratings. Hence, unlike BRS, TRAVOS does not assume that the majority of ratings are fair.

4.2. Simulation Environment

In our simulations, service characteristics of a service provider are generated as follows. First, a service space is defined so that all possible services are represented within this service space. Dimensions of the service space and their ranges are tabulated in Table 1. Each service provider has a multidimensional region called service region in this service space. This region is randomly generated. The service space and the service regions have 15 dimensions. A service region covers all of the services produced by the service provider. If a consumer who is located in Istanbul orders *two* books titled *Anagrams* from the service provider, the service that the provider delivers will be constructed as follows. The properties that are specified (shopping item id, quantity, and location) will be fixed. For the remaining attributes, the service provider will choose random values making sure that the values stay in the range of its service region. So, for this example, the degree of freedom for generating services will be reduced to 12.

Given the service constraints, the simulation environment generates a demand of a service consumer as follows. A demand space is constructed for the consumer by removing the dimensions of the service space that do not belong to *Demand* class. Then, a random region in this demand space is chosen. The center of this region represents the demanded service. In response to this demand, the chosen provider supplies a service. If the provided service for this demand stays within the margins of demand region, the service consumer is satisfied; otherwise, she is dissatisfied. This is the implementation of F_{taste} function in our evaluations. The simulation environment guarantees that each demand can be satisfied by exactly one service provider. Next, the simulator creates the *similar demand criteria* for the demand of the service consumer. This is again done by creating a new region (*similar demand region*). Essentially, this is the demand region after some dimensions have been removed. The number of dimensions to be removed and these dimensions are chosen randomly. Service demands staying within the margins of the similar demand region are classified as similar demand by the consumer.

The simulation environment is set up with 10 service providers and 200 service consumers. Only one of the service providers can satisfy a given service demand. Simulations are run for 100 epochs, where an epoch refers to a discrete time slot during which each consumer may request at most one service. When the simulations start, agents do not have any prior experiences with service providers. At each epoch, with a probability of 0.5, a consumer requests a service for its current service demand. Then, it collects experiences related to the similar service demands from other consumers in order to use for service selection. For this purpose, a P2P search mechanism is used (Şensoy and Yolum 2007), which enables a consumer to locate others with similar service demands. An overview of this protocol is provided in Appendix A. In our simulations, we force consumers to make service decisions based on the information from others rather than their own previous experiences. In this way, we can test the abilities of our approach better against subjectivity, unreliability, and context-awareness.

4.3. Simulation Parameters and Evaluation Metrics

In our simulations, we try to mimic real-life scenarios. Therefore, we have parameterized our simulation environment considering some of the important factors in real life. The factors are subjectivity, variations on context, and deception. We briefly explain our parameters related to the factors below.

1. *Subjectivity:* Consumers having similar demands may have different satisfaction criteria. This means that for the same demand and the same supplied service, two consumers may have different degrees of satisfaction (e.g., ratings) depending on their satisfaction criteria. This is the subjectivity of the consumers. In the experiments, we define subjectivity as a parameter $(R_{\text{sub}j})$, which determines the ratio of consumers having similar demands but conflicting satisfaction criteria. For example, if $R_{\text{sub}j} = 0.5$, half of the consumers having the same or similar demands have conflicting satisfaction criteria (tastes). In the experiments, only one provider satisfies a service demand of a consumer. Now, consider two consumers with the same demand and assume that $\{P_0, \ldots, P_9\}$ are the providers in the environment. Therefore, if those two consumers have the same taste, both of them give a good rating for the same provider P_i and they give bad ratings for the other nine providers. However, if those two consumers have conflicting satisfaction criteria, the first consumer gives a good rating to a provider P_i, and the second consumer gives a good rating to another provider P_j, where $P_i \neq P_j$. In this setting, the first consumer gives a bad rating to P_j and the second consumer gives a bad rating to P_i. On the other hand, both of the consumers give bad ratings to the other eight providers P_k, where $k \neq i$ and $k \neq j$. Therefore, ratings of the consumers are consistent for those providers, even though their ratings are conflicting for P_i and P_j.
2. *Variation on context:* As frequently seen in real world, each service consumer changes its service demand after receiving a service. This is done with a predefined probability (P_{CD}). After changing its demand, the service consumer collects information for its new service demand. This parameter is introduced to mimic variations on the context of service demands in real life.
3. *Liars:* Another parameter in the simulations is R_{liar}, which defines the ratio of liars in the consumer society. Liars modify their experiences before sharing, so that they mislead the other consumers the most. This is achieved by disseminating bad experiences about

the good providers and good experiences about the bad providers. Behaviors of the liars are summarized as follows. If an experience of a liar contains a satisfactory service, the liar modifies the experience before sharing with others so that the received service within the experience looks like it has not satisfied the demand of the customer. For example, if the liar demanded a notebook within 7 days from a provider in the past and it is delivered on time, the liar states in its experience that the merchandise was not received or the notebook was delivered within 120 days. On the other hand, if an experience contains an unsatisfactory service, the liar modifies the experience before sharing so that the received service looks like it has satisfied the demand of the customer. For example, if the liar demanded a notebook within 7 days from a provider in the past, but delivery was made after 30 days, the liar states in his or her experience that the notebook was delivered within 7 days.

In this work, we use rating-based service selection approaches for benchmark comparisons. Hence, we define the behavior of liars once again for rating-based service selection approaches. In this case, liars give bad ratings to good service providers and good ratings to bad service providers. Formally, if the true rating of a liar to a provider is r, the liar modifies the rating as $r' = 1 - r$ before sharing with the other consumers. This formulation for lying is frequently used in the trust and reputation literature (Yu and Singh 2003; Whitby et al. 2005; Teacy et al. 2006).

4. *Evaluation metrics:* Our main performance metric is success in service selection. We measure it as the percentage of the satisfactory service selections made by the consumers. Intuitively, in deceptive environments, the success in service selection should be correlated with the amount of filtered deceptive information during service selection. As the amount of unfiltered deceptive information increases, the performance of service selection approaches is expected to decrease. Our supplementary performance metric is the error in identifying liars among the advisors during service selection.

4.4. Experimental Results

POYRAZ is an integrated system that is composed of an experience-based service selection approach described in Section 2 and a trust model to filter deceptive information as proposed in Section 3. In this section, we first experimentally evaluate POYRAZ as an integrated service selection approach and then we compare alternative models of trust for POYRAZ to provide additional validation for the deception filtering approach that we employ.

4.4.1. Experimental Results to Validate POYRAZ. In this part of our experiments, we demonstrate the performance of our approach in three steps. First, we examine our approach in deceptive environments when there is no subjectivity and variation on context. Second, we investigate the performance change when subjectivity is included in the experiments. Third, we consider reliable environments with no subjectivity and inspect the performance of our approach when consumers are allowed to change the context of their service demands.

1. *Deceptive environments without subjectivity and variation of context:* In this setting, consumers do not change their service demands ($P_{\text{CD}} = 0.0$). Therefore, the context of their service selections does not change during the experiments. Moreover, consumers with the similar demands have the same taste ($R_{\text{sub}} = 0.0$), so the consumers with similar service demands are satisfied with the same providers. In order to learn the effect of deception in this setting, we repeat our experiments for different percentages of liars.

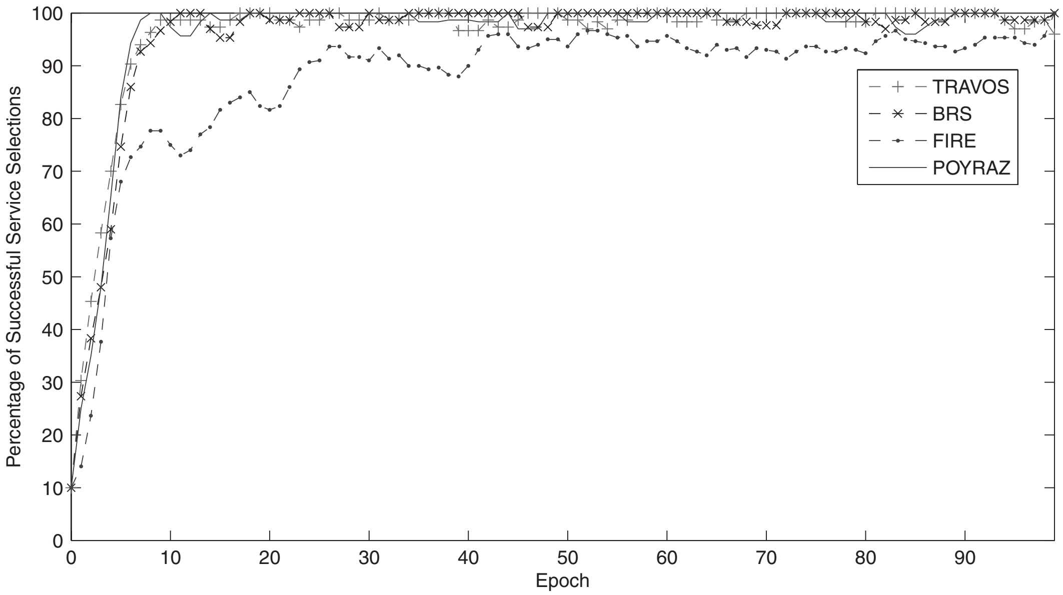

FIGURE 4. Percentage of successful service selections for $R_{\text{liar}} = 0.2$. There is no subjectivity or variation on context during the experiment ($R_{\text{sub}} = 0.0$ and $P_{\text{CD}} = 0.0$).

Figure 4[4] shows the percentage of successful service selections in one of our experiments where only 20% of consumers are liars ($R_{\text{liar}} = 0.2$). In this setting, performances of POYRAZ, TRAVOS, and BRS are almost the same. Those approaches can successfully determine satisfactory service providers. However, performance of FIRE is considerably lower than that of the other approaches. The main reason for this performance difference is the fact that FIRE does not have a mechanism for detecting and filtering out unfair ratings given by the liars. Figure 5 shows another experiment in the same setting where 50% of the consumers are liars ($R_{\text{liar}} = 0.5$). As shown in the figure, performances of POYRAZ and TRAVOS decrease slightly, when the percentage of liars is increased from 20% to 50%. However, in this case, the performances of FIRE and BRS decrease dramatically.

Those experiments show that some of the service selection approaches are significantly affected by deceptive information disseminated by the liars in the society. In order to see the effect of deceptive information more clearly, we conduct simulations for different ratios of liars, by varying the value of the R_{liar} parameter. Figure 6 shows the average percentage of successful service selections through the experiments for different ratios of liars. The figure shows that the performances of POYRAZ and TRAVOS do not decrease significantly as the percentage of liars increases in the society. Although TRAVOS is slightly more sensitive to the ratio of liars than POYRAZ, both of these approaches have a very good performance in determining satisfactory service providers. Unlike POYRAZ and TRAVOS, FIRE, and BRS are extremely sensitive to the percentage of liars. In particular, the performance of BRS decreases more dramatically than the performance of FIRE for $R_{\text{liar}} > 0.3$. Although FIRE seems to be better than BRS for higher percentages

[4] Note that in Figures 4, 5, and 8, we show the results only for one individual simulation instead of the average results in order to explain the variation in service satisfaction over time with an example from our simulations.

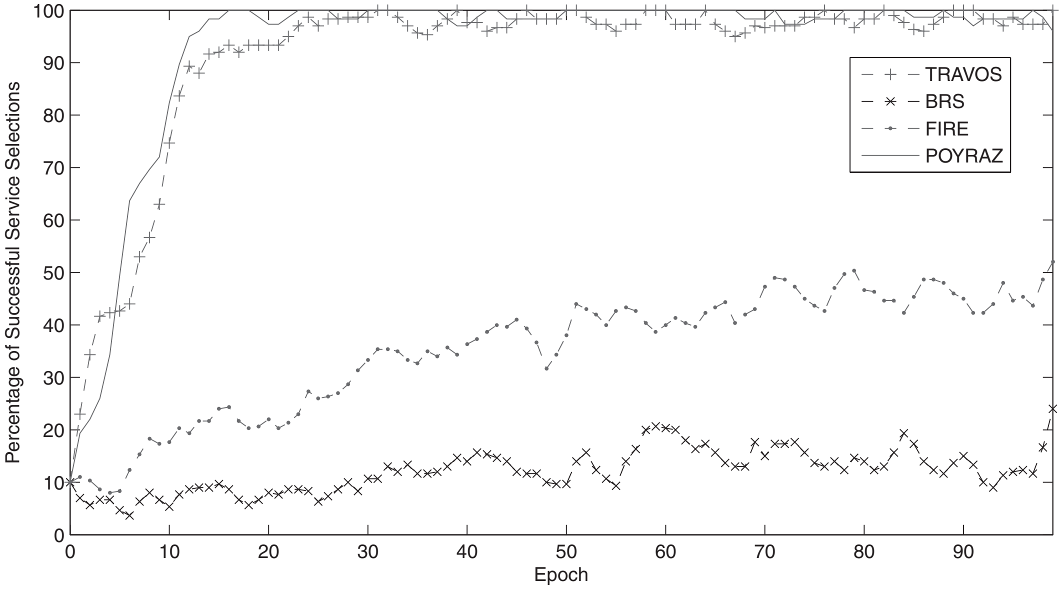

FIGURE 5. Percentage of successful service selections for $R_{\text{liar}} = 0.5$. There is no subjectivity or variation on context during the experiment ($R_{\text{sub}} = 0.0$ and $P_{\text{CD}} = 0.0$).

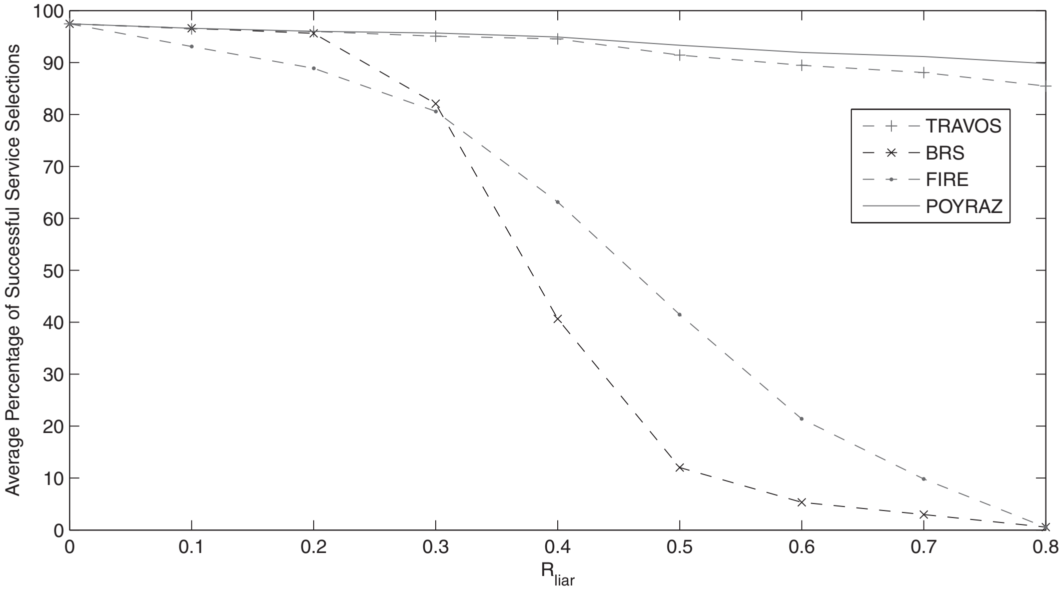

FIGURE 6. Average percentage of successful service selections for varying values of R_{liar}. There is no subjectivity or variation on context during the experiments ($R_{\text{sub}} = 0.0$ and $P_{\text{CD}} = 0.0$).

of liars, eventually performances of both approaches approach 0.0 as the ratio of liars approaches 0.8.

Figure 6 shows that FIRE is more successful than BRS for $R_{\text{liar}} > 0.3$. This result is interesting, because unlike FIRE, BRS has a mechanism for filtering out deceptive information. In order to understand the reasons behind this observation, in Figure 7 we

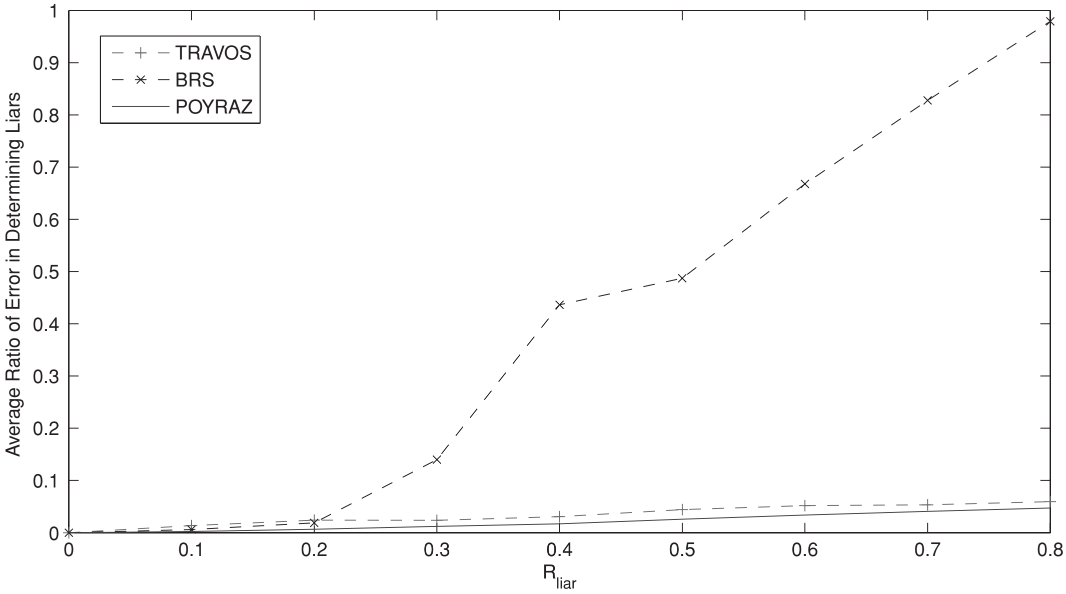

FIGURE 7. Average ratio of error in determining liars for varying values of R_{liar}. There is no subjectivity or variation on context during the experiments ($R_{\text{sub}} = 0.0$ and $P_{\text{CD}} = 0.0$).

plot the average error in determining liars for BRS, TRAVOS, and POYRAZ. As shown in Figure 7, when the ratio of liars becomes greater than 0.2, BRS's error in determining liars dramatically increases. This means that BRS starts misclassifying liars as honest and honest consumers as liars when the ratio of liars increases. This is an expected result because BRS is designed for environments where a significant majority of the consumers are honest. The high amount of error in determining liars implies the usage of more ratings from liars and fewer ratings from honest consumers. Therefore, in the case of BRS, filtering ratings may lead to less successful service selection compared to FIRE. Unlike BRS, TRAVOS, and POYRAZ have very low ratios of error. POYRAZ and TRAVOS fail to determine liars in at most 5% and 6% of the cases, respectively, while BRS's error in determining liars approaches 100%. In this part of our evaluations, we show that FIRE and BRS fail in service selection when there are a significant number of liars in the environment. Next, we repeat our experiments for the case where consumers have different tastes for similar service demands.

2. *Deceptive and subjective environments without variation on context:* In this setting, consumers do not change their service demands ($P_{\text{CD}} = 0.0$) as in the previous setting. However, this time, half of the consumers having similar service demands have conflicting satisfaction criteria ($R_{\text{sub}} = 0.5$). Figure 8 demonstrates the results of an experiment where there are no liars among the consumers ($R_{\text{liar}} = 0.0$). The figure shows that performances of rating-based approaches are significantly lower than the performance of POYRAZ. This is the effect of subjectivity on the rating-based service selection, because in the case where there is no subjectivity ($P_{\text{CD}} = 0.0$, $R_{\text{liar}} = 0.0$ and $R_{\text{sub}} = 0.0$), performances of the four service selection approaches are the same as shown in Figure 6. Vulnerability of rating-based approaches to subjectivity is expected, because rating-based approaches assume that there is no subjectivity among the consumers (Jøsang et al. 2007). That is, they assume that every consumer gives good ratings to "good" providers and bad ratings

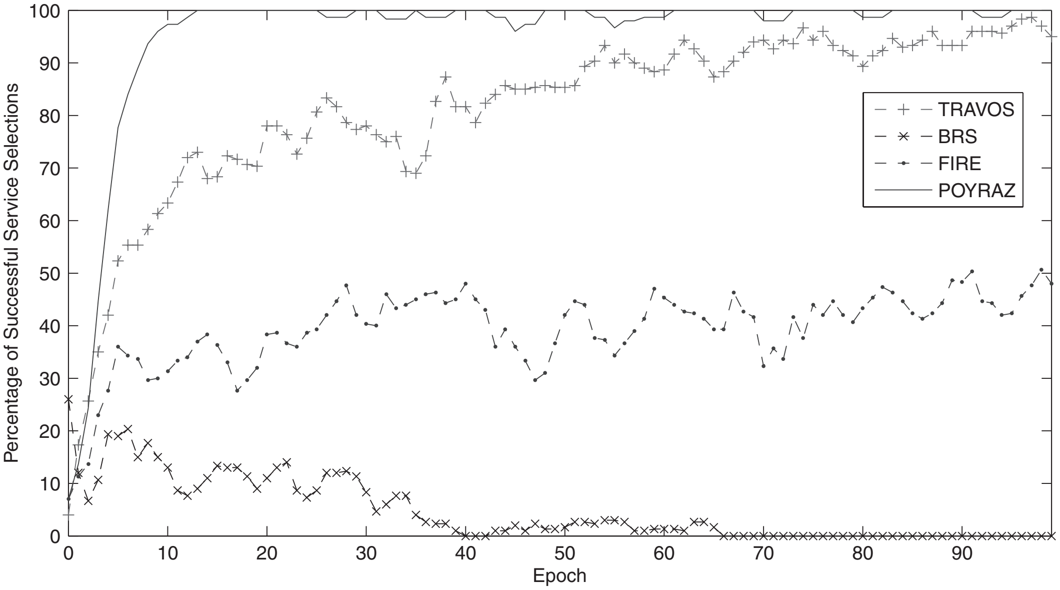

FIGURE 8. Percentage of successful service selections. All of the consumers are honest ($R_{\text{liar}} = 0.0$). Half of the consumers having similar demands have different tastes ($R_{\text{sub}} = 0.5$). There is no variation on context during the experiment ($P_{\text{CD}} = 0.0$).

to "bad" providers. However, in the case of subjectivity ($R_{\text{sub}} = 0.5$), the definition of "good" and "bad" depends on each consumer and may change significantly from consumer to consumer as in real life.

In this setting, the performance of TRAVOS is much better than the performances of BRS and FIRE. The main reason for this performance difference is that TRAVOS can successfully identify advisors whose ratings conflict with personal observations. In other words, TRAVOS labels advisors with conflicting taste as liars and it removes their ratings during service selection. In this way, it enables consumers to use ratings from others with similar taste. Although TRAVOS is not proposed to handle subjectivity, its mechanism of filtering out unfair ratings works well for removing subjectivity during service selection. This is because both subjectivity and deception ultimately result in consumers disseminating conflicting ratings for the same providers. Note that when subjectivity is high as in our setting, BRS has the worst performance.

Half of the consumers are liars in Figure 5 and have different tastes in Figure 8. These figures show that the effect of subjectivity is more severe than that of the deception for TRAVOS and BRS. The main reason for this observation is the fact that it is harder to determine consumers (advisors) with different taste than the liars. That is, ratings of a honest consumer and a liar for the same providers always conflict. However, if two consumers are both honest but their satisfaction criteria are different as in the case of subjectivity, their ratings conflict only for the providers that satisfy one of those consumers. On the other hand, ratings of those consumers are consistently negative for the other providers (ones which do not satisfy any of those consumers). For example, in our experiments, two consumers with different tastes give conflicting ratings only for two of the providers while their ratings are consistently negative for the rest. Therefore, determining consumers with different taste is more difficult than determining liars.

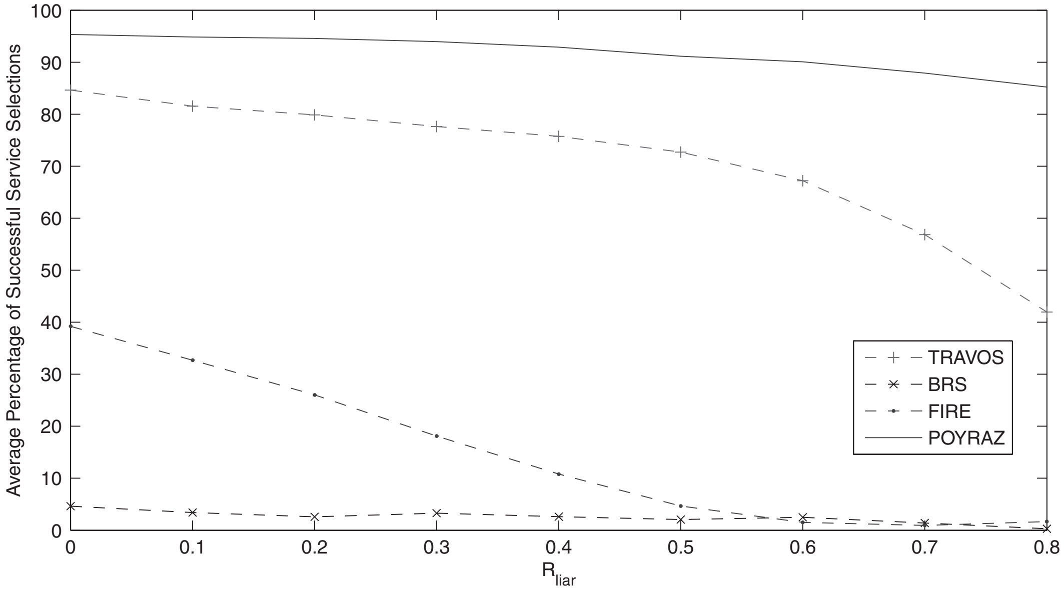

FIGURE 9. Average percentage of successful service selections for varying values of R_{liar}. Half of the consumers having similar demands have different tastes ($R_{\text{sub}} = 0.5$). There is no variation on context during the experiments ($P_{\text{CD}} = 0.0$).

In many real-life settings, deceptive and subjective information exist together. In order to see the combined effect of subjectivity and deception during service selection, we change the ratio of liars when there exists subjectivity in our experiments. We show our results in Figure 9. Our experiments show that for POYRAZ, increasing the ratio of liars from 0 to 0.8 results in a small decrease in the percentage of successful service selections; the decrease is only from 96% to 86%. If we compare the performance of TRAVOS, BRS, and FIRE, we see that TRAVOS has the best performance in terms of the success in service selection. However, TRAVOS is more sensitive to deception and subjectivity than POYRAZ; its performance decreases from 85% to 42% in Figure 9 when the ratio of liars is increased from 0 to 0.8. Therefore, we can confidently state that POYRAZ is much more robust to deception and subjectivity than TRAVOS, BRS, and FIRE.

3. *Reliable environments with variation on context and no subjectivity:* Unlike the previous settings, in this setting, consumers change their service demands with probability P_{CD} after receiving a service. Moreover, all of the consumers are honest ($R_{\text{liar}} = 0.0$) and their satisfaction criteria are similar if their service demands are also similar ($R_{\text{sub}} = 0.0$). Figure 10 summarizes the average percentage of successful service selections for different approaches when the value of P_{CD} is varied from 0.0 to 1.0. Figure 10 indicates that the performance of the rating-based approaches decreases sharply when $P_{\text{CD}} > 0$, whereas the performance of the proposed approach is near to 100%. This sharp performance decrease is intuitive, because ratings of a consumer reflect the aggregation of its past transactions with the providers. Assume that a provider *BookHeaven* is an expert on selling books, but not competent in selling music CDs. Assume that *Bob* recently made five transactions for five items from *BookHeaven*: two books and three CDs. Because *BookHeaven* is an expert on book selling, the transactions related to the books were successful, but the transactions related to the CDs were not. In this case,

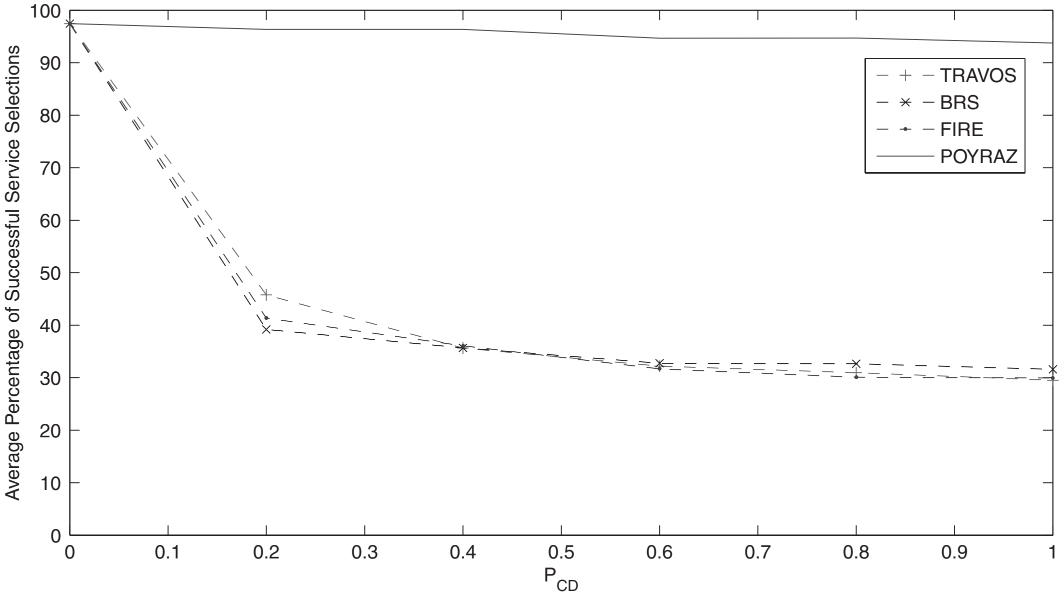

FIGURE 10. Average percentage of successful service selections when the context is allowed to vary during service selection. There is no subjectivity ($R_{\text{sub}} = 0.0$) and all of the consumers are honest ($R_{\text{liar}} = 0.0$).

the overall rating of *Bob* for *BookHeaven* is bad, because number of unsuccessful transactions is higher than that of the successful transactions. If another consumer wants to buy a book, the rating of *Bob* for *BookHeaven* will be misleading. In other words, as consumers change their demands, their ratings about the providers become more misleading, depending on the variation in the expertise of the providers. However, POYRAZ differentiates between the experiences belonging to different contexts. It can easily recognize that *BookHeaven* can provide a satisfactory service if a book is demanded, but it cannot produce a satisfactory service if a music CD is asked for. Hence, as seen in the Figure 10, POYRAZ almost always leads to satisfactory service decisions. Its percentage of successful service selections does not go below 94% while the performances of the rating-based approaches decrease to 30%.

4.4.2. Comparing Alternative Models of Trust for POYRAZ. POYRAZ has two components, an experience-based service selection component and a trust-based deceptive information filtering component. In this part of our experiments, we compare the trust model used in POYRAZ with alternatives from the literature. For this purpose, we integrate the deceptive information filtering approaches of TRAVOS and BRS into the experience-based service selection component of POYRAZ and compare these integrated approaches with POYRAZ. In this way, we demonstrate that the deceptive information filtering component of POYRAZ outperforms the alternatives of BRS and TRAVOS.

The experience-based service selection approach in Section 2 does not detect and filter deceptive experiences. Therefore, this approach is highly vulnerable to deception. Figure 11 shows the performance of the experience-based service selection approach (denoted as *Exp* in the figure) when there are liars in the environment. As the ratio of liars in the environment increases, the percentage of successful service selections considerably decreases and becomes 30% when 80% of the consumers are liars. This means that experience-based service selection fails significantly when there are liars in the environment.

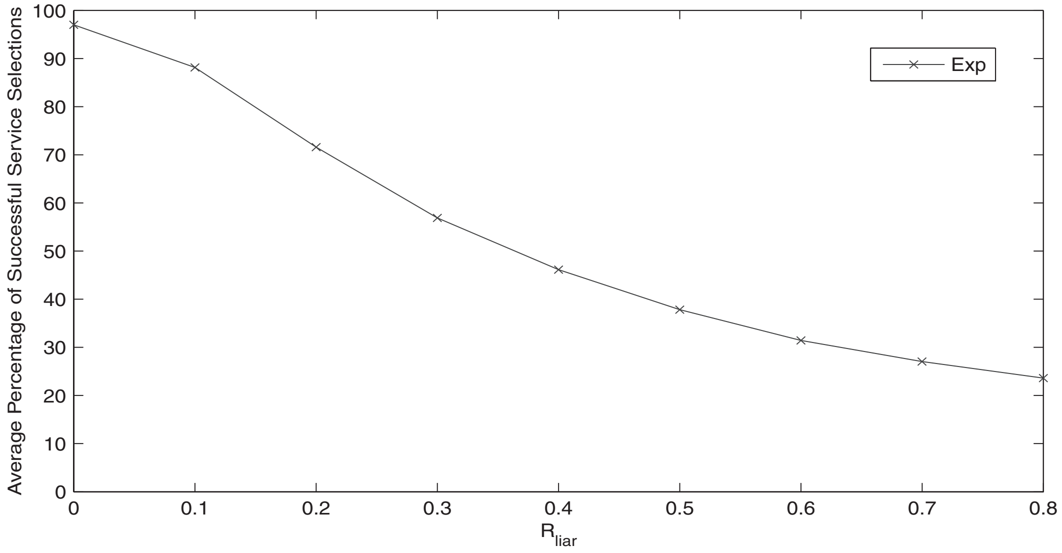

FIGURE 11. Average performance of the experience-based service selection for different ratios of liars ($R_{\text{sub}} = 0.5$ and $P_{\text{CD}} = 0.2$).

In order to perform service selections under deception, we need to filter deceptive information. Otherwise, most of the service decisions may lead to dissatisfaction of the consumers. In order to compare alternative models of trust for POYRAZ, we construct two integrated service selection approaches called Exp^{BRS} and Exp^{TRAVOS} . Exp^{BRS} and Exp^{TRAVOS} are exactly the same as POYRAZ, except they use different deceptive information filtering methods, that of BRS and of TRAVOS, respectively.

Figure 12 shows the performance of the experience-based service selection when different deceptive information filtering methods are used. POYRAZ has the best performance in our experiments. This means that the information filtering approach used in POYRAZ is better than those used in BRS and TRAVOS. The performance of the experience-based service selection decreases dramatically when the information filtering method of BRS is used to filter deceptive experiences. This is expected because this filtering method assumes that a significant majority of consumers are honest (Whitby et al. 2005). If this is not the case, error in determining liars dramatically increases as explained before, so Exp^{BRS} fails significantly.

The performance of Exp^{TRAVOS} does not go below 82%. This means that the performance of the experience-based service selection is enhanced significantly when the information filtering method from TRAVOS is integrated. On the other hand, for each ratio of liars, POYRAZ outperforms Exp^{TRAVOS} and its performance does not go below 87%. Hence, the proposed deceptive information filtering approach in Section 3 is better than the other deceptive information filtering approaches.

Note that Figure 12 shows the average percentage of successful service selections during the simulations, and it does not show how the service selection performance changes over time during simulations. In order to show how well POYRAZ does with respect to Exp^{BRS} and Exp^{TRAVOS} more clearly, we demonstrate average service selection performance over time for different ratios of liars in Figures 13 and 14. For simplicity, only the first 50 epochs of the simulations are shown in these figures.

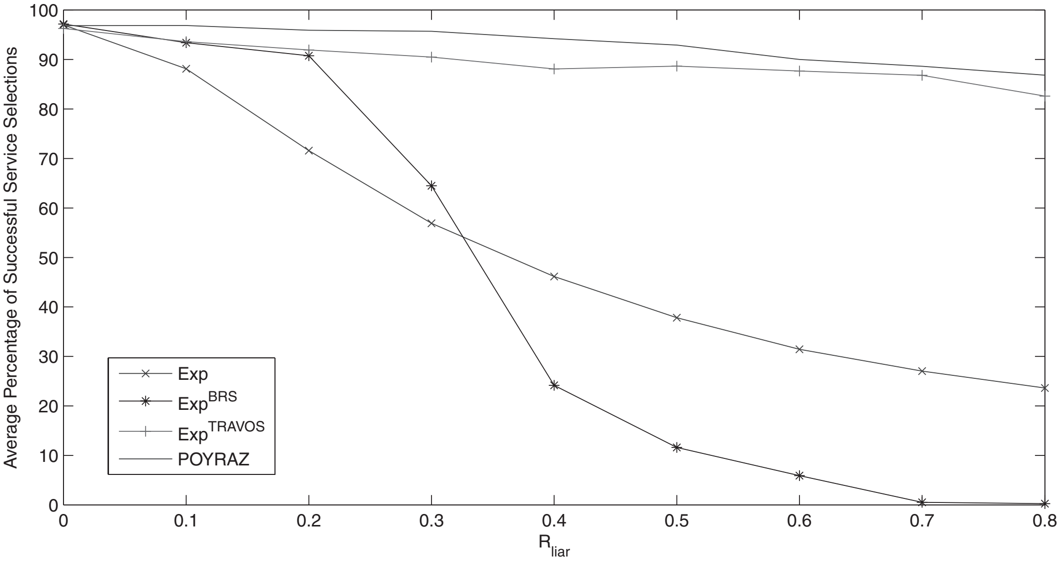

FIGURE 12. Average percentage of successful service selections for different ratios of liars ($R_{\text{sub}} = 0.5$ and $P_{\text{CD}} = 0.2$).

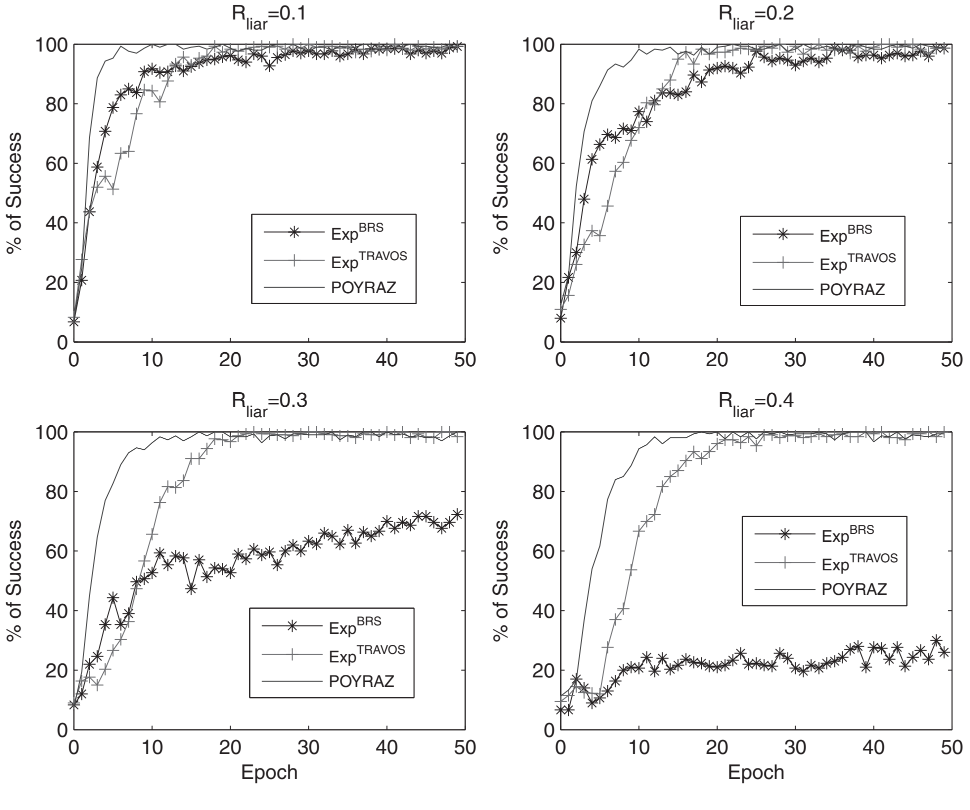

FIGURE 13. Average service selection performance over time for different ratios of liars ($0.1 \leq R_{\text{liar}} \leq 0.4$, $R_{\text{sub}} = 0.5$ and $P_{\text{CD}} = 0.2$).

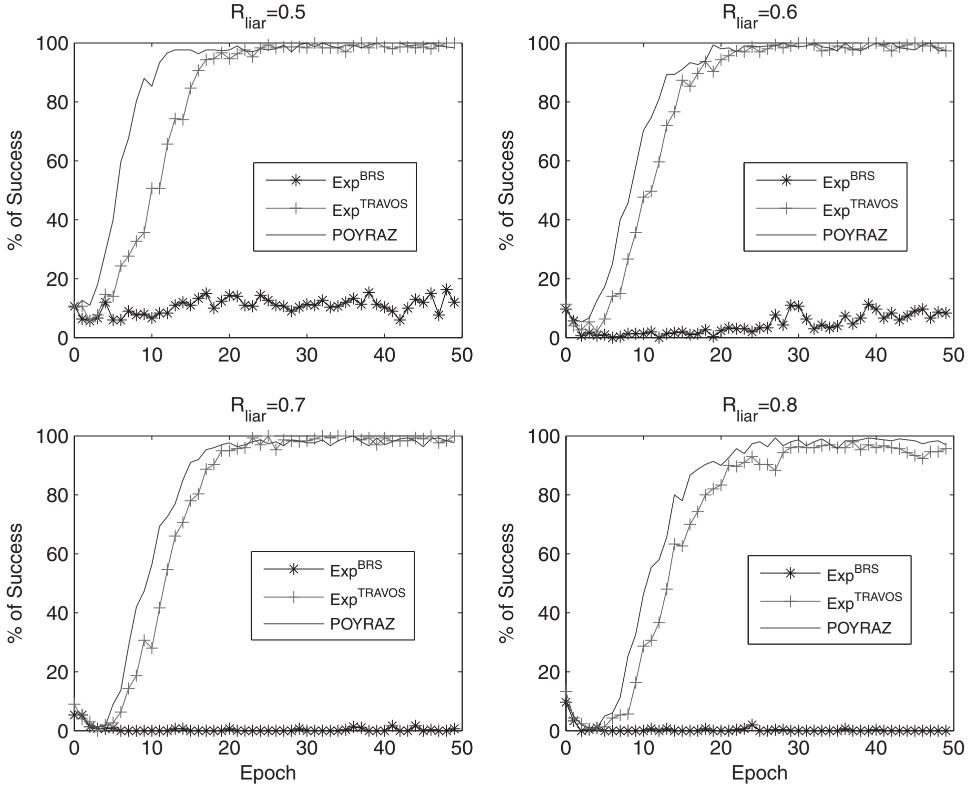

FIGURE 14. Average service selection performance over time for different ratios of liars ($0.5 \leq R_{\text{liar}} \leq 0.8$, $R_{\text{sub}} = 0.5$ and $P_{\text{CD}} = 0.2$).

Figures 13 and 14 show that when the ratio of liars is low ($R_{\text{liar}} < 0.3$), Exp^{BRS} is much better than Exp^{TRAVOS} in the beginning of the simulations, because it can determine deceptive experiences using the shared experiences instead of personal observations, which do not exist in the beginning but accumulate over time. On the other hand, Exp^{TRAVOS} uses only personal observations, so it cannot determine liars until it gets sufficient personal observations over time. Once Exp^{TRAVOS} has enough number of personal observations, it can successfully determine unreliable consumers and outperforms Exp^{BRS}. For higher ratios of liars ($R_{\text{liar}} \geq 0.3$), the performance of Exp^{BRS} is very low.

The performance of POYRAZ is better than that of Exp^{BRS} and Exp^{TRAVOS}, because it uses both personal and shared information to determine deceptive experiences. When the personal observations are not enough, POYRAZ combines its personal observations with the public information from others. Therefore, it can achieve a good performance even in the beginning of the simulations. For higher ratios of liars, public information misleads POYRAZ, but POYRAZ starts using its observations soon afterwards, so it is not affected significantly by the misleading public information.

When the ratio of liars is high ($R_{\text{liar}} > 0.5$), the performances of POYRAZ and Exp^{TRAVOS} are close, but POYRAZ still outperforms Exp^{TRAVOS}. This performance difference can be explained by the fact that TRAVOS does not use all of its personal observations related to an advisor while evaluating the trustworthiness of the advisor. However, POYRAZ uses a larger number of personal observations while evaluating trustworthiness. As a result, it determines liars and reaches its maximum service selection performance earlier than Exp^{TRAVOS}.

5. DISCUSSION

In this section, we first summarize our experimental results and then discuss our work with references to the literature. Last, we provide an overview of the significance of our contributions.

5.1. Summary of Results

As explained in the Introduction, this research aims to provide an integrated approach for context-aware service selection in deceptive environments that is experimentally confirmed to be valuable. Our primary experimental results are presented in Section 4.4.1 and can be summarized as follows.

We compare our approach with three rating-based service selection approaches from the literature; FIRE, BRS, and TRAVOS. In environments where there is no deception, variation on context and subjectivity, these rating-based approaches have the same performance as POYRAZ. All of the service selection approaches can successfully determine the most satisfactory service providers in this case. Unfortunately, this setting is far from being realistic in many real-life scenarios. In the case where there are liars among the consumers, the performances of BRS and FIRE dramatically decrease. The decrease is sharp when the ratio of liars increases. However, TRAVOS and POYRAZ almost always make satisfactory service selections, even if most of the consumers in the society are liars.

When the consumers are allowed to have different tastes for the same service demands, rating-based approaches suffer from the subjectivity. Although, subjectivity dramatically affects the performances of BRS and FIRE, the performance of TRAVOS decreases only 10%. TRAVOS achieves relatively better performance than BRS and FIRE by determining and eliminating ratings from the consumers having different tastes. Unlike the rating-based approaches, POYRAZ is not considerably affected by the subjectivity. This is intuitive because, unlike rating-based approaches, POYRAZ does not depend on the subjective *opinions* of other consumers during service selection. If there is not only subjectivity but also liars among the consumers, the performance of TRAVOS decreases dramatically while POYRAZ is only affected slightly in this setting.

Consumers may vary the context of their service demands regularly as in many real-life settings. Even if consumers have the same taste and are always honest, variation on context may result in serious decreases in the performances of service selection approaches. Our experiments show that rating-based approaches are very sensitive to the variation on context. If consumers frequently change their service demands, their ratings become more confusing than before. As a result, TRAVOS, BRS, and FIRE fail in selecting the satisfactory service providers. On the other hand, POYRAZ can differentiate between different experiences depending on their contexts. Hence, our approach is not affected by the frequent changes in the context of service demands and almost always selects satisfactory service providers.

POYRAZ is a novel combination of experience-based service selection and trust-based deceptive information filtering. In order to show that our choice of method for information filtering is better than its alternatives, we also empirically compare our filtering method with other deceptive information filtering methods from the literature, in Section 4.4.2 Our experiments show that our deceptive information filtering method determines liars more accurately and improves the performance of the experience-based service selection more significantly.

These results have an important implication: In many situations, an agent cannot and will not know the details of the environment, such as the percentage of liars or how often others change their context. It would be extremely useful to be able to rely on the service selection method without considering such environmental details. Our results above clearly show that POYRAZ can be used in any environment where service selection is needed without considering the characteristics of the environment explicitly.

5.2. Related Work

Current service provider selection strategies accept ratings as first-class citizens, but do not allow more expressive representations like we have here. Whereas rating-based approaches (Jøsang et al. 2007) assume that the ratings are given and taken in similar contexts (e.g., in response to similar service demand), we can make the context explicit. This allows agents to evaluate others' experiences based on their needs. Thus, the use of contextual information and experiences improve the satisfaction rate of the consumers as we show in this paper.

The infinite relational trust model of Rettinger, Nickles, and Tresp (2007) takes into account contextual information as well, when modeling trust between interaction agents, but only focuses on learning initial trust for unknown agents. Their model makes use of only direct interactions between two agents, whereas we allow experiences to be shared among agents. Moreover, unlike their model, we describe contextual information in our approach using an ontology in a flexible manner.

Sen and Sajja (2002) develop a reputation-based trust model that is used for selecting processor agents for processor tasks. Each processor agent can vary its performance over time. Agents are looking for processor agents to send their tasks by using only evidence from others. Sen and Sajja propose a probabilistic algorithm to guarantee finding a trustworthy processor. In our framework, service demands among agents are not equivalent; and a provider that is trustworthy for one consumer need not be so for a different consumer. Hence, each consumer may have to select a different provider for its needs.

Yolum and Singh study properties of referral networks for service selection, where referrals are used among service consumers to locate service providers (Yolum and Singh 2005). Current applications of referral networks rely on exchanging ratings. They suffer from circulation of subjective information. However, it would be interesting to combine referral networks with the ontology representation that we propose so that agents can exploit the power of ontologies for knowledge representation as well as referrals for accurate routing.

Maximilien and Singh develop a quality of service (QoS) ontology to represent the quality levels of service agents and the preferences of the consumers (Maximilien and Singh 2004). Their representation of QoS attributes is richer (using availability, capacity, and so on); however, their ontology does not represent commitments and thus business contracts as a part of the ontology. Further, their system does not allow reasoning by agents individually as we have developed here.

Our work is distinguished from the literature as follows. First, our approach enables context-aware service selections by enabling consumers to record their past experiences semantically using an ontology, instead of plain subjective ratings. Our representation of past experiences handles subjectivity and enables consumer-oriented service selections. Second, unlike most of the service selection approaches, our approach explicitly reasons about the reliability of information resources during service selection. While the challenges of selecting providers and detecting deception are considered together in models such as (Zhang and Cohen 2006), this is done in the context where it is ratings of providers that are shared between consumers, so that approaches for addressing subjective differences and context-dependent needs are not developed.

5.3. Overview of Contributions

Previous approaches to service selection are mainly based on capturing and exchanging ratings about service offerings. Ratings only reflect the subjective opinions of the consumers about the providers and do not contain any semantic information about the episodes that lead to these ratings. Şensoy and Yolum previously proposed to enable consumers to semantically and objectively share their past experiences with the providers, instead of their subjective ratings (Şensoy and Yolum 2007). This approach enables consumers to evaluate others' experiences using their own criteria and context. In this paper, we show that this approach is vulnerable to deception and we propose to integrate a trust mechanism into this particular approach for context-aware service selection. That is, we explicitly deal with the problem of deception during context-aware service selection in this paper. This is very important and useful, because deceptive information significantly misleads consumers and results in unsatisfactory service selections.

In order to deal with deception, it is necessary to apply a filtering technique that will detect which experiences are bogus and will filter them so that they will not be considered in the service selection process. Several filtering techniques are available in the literature. Our experiments show that it is possible to extend available filtering approaches to be used with experience-based service selection. However, existing filtering mechanisms are designed to be applied in environments where consumers are exchanging ratings, rather than experiences. Hence, they are not targeted to exploit the benefits of experiences. To do this, we propose a new method for the detection and filtering of deceptive information. Our proposed filtering method is applied during service selection. We empirically show that this filtering method is better than its well-known alternatives from the literature. Unlike other deceptive information filtering methods in the literature, the proposed method handles subjectivity effectively and calculates trustworthiness of information sources in a context dependent way using the semantically described experiences of the consumers. In summary, the main contribution of this paper is an integrated approach that is capable of context-aware and consumer-oriented semantic service selection under deception. This approach is important and useful, because, unlike the other approaches in the literature, it handles subjectivity, variation on context, and deception together.

Additionally, for the first time in this paper, well-known service selection approaches from the literature are compared in terms of their sensitivity to context change. Similarly, context-aware service selection is compared with its rating-based counterparts from the literature in detail. Hence, this paper presents a comprehensive comparison of existing service selection approaches under different settings.

6. CONCLUSION

As the number of service providers increases dramatically on the Web, it gets harder to select an appropriate provider for a particular service demand. Traditional approaches to service selection are usually based on the exchange of ratings among consumers in a multiagent system. The main challenge here is the fact that consumers' tastes and expectations may vary considerably for the same service. Therefore, ratings may be significantly misleading if the raters and the consumers using their ratings do not share similar tastes. Moreover, the problem of service selection becomes more challenging when some of the consumers disseminate deceptive information about the providers.

In this paper, we develop a service selection framework that is not only consumer-oriented and context-aware, but is also robust to deceptive information disseminated by malicious consumers. In our proposed approach, service consumers semantically describe their past experiences with service providers, so that any consumer can interpret the experiences of others using its own satisfaction criteria and context. If these experiences are not indicated truthfully, they may be misinterpreted by consumers, as they make decisions about which service providers to select.

In order to cope with deception, we adapt an approach for modeling the trustworthiness of agents in a multiagent system—one that allows for a weighted combination of personal credit and public credit of the consumers that share their experiences, in order to determine whether the consumer is a liar. In order to be sensitive to the different satisfaction criteria that consumers have, we evaluate the experiences that are shared by consumers according to the taste function (F_{taste}) of each consumer, as part of the reasoning about deception.

We then integrate this method for detecting deception into our service selection framework, in order to filter out deceptive information. The result is an overall method for service provider selection that offers definite improvements over other methods that do not adequately account for subjectivity, context-awareness, and untruthfully shared experiences together. We experimentally show that better service providers can be chosen using our approach, even if consumers have different tastes, they change context of their service demands over time or a significant portion of them are liars. In summary, we offer a valuable new framework that supports context-aware service selection and the handling of deception simultaneously, of use for any Web-based application where consumers are required to reason carefully about the available providers.

In this work, we have assumed that the consumer agents record and exchange their experiences with the providers willingly. However, in open settings, there can be times when the agents do not prefer to cooperate with the other agents. This could stem from two facts: (1) The users of the agents may not want to record their experiences as needed by the system and (2) the users may not be willing to exchange this information. Hence, incentives must be created for users to record and exchange their experiences. Incentive creation is an interesting problem that has received attention in the literature (Zhang and Cohen 2007). Such techniques can complement our work to create incentives for exchanging experiences.

We have evaluated the performance of our approach in detecting the possible liars being fairly consistent in lying. For future work, in our evaluations, it would be worthwhile to explore the case where some liars lie only in a specific context while being honest in other contexts. It would also be worthwhile to consider other types of liars from the literature, such as Exaggerated Positive and Exaggerated Negative defined in Yu and Singh (2003). The performance of detecting these types of liars would then be evaluated and compared against competing approaches.

ACKNOWLEDGMENTS

Authors Şensoy and Yolum thank to Boğaziçi University Research Fund and The Scientific and Technological Research Council of Turkey (TUBITAK) for their support under grants BAP07A102 and 105E073, respectively. Authors Zhang and Cohen acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

DELLAROCAS, C. 2000. Immunizing online reputation reporting systems against unfair ratings and discriminatory behavior. *In* Proceedings of the Second ACM conference on Electronic commerce, Minneapolis, MN, pp. 150–157.

DUDA, R. O., P. E. HART, and D. G. STORK. 2001. Pattern Classification. John Wiley and Sons, Chichester, UK.

FEIGENBAUM, L., I. HERMAN, T. HONGSERMEIER, E. NEUMANN, and S. STEPHENS. 2007. The semantic Web in action. Scientific American, **297**(6):90–97.

HIRTLE, D., H. BOLEY, B. GROSOF, M. KIFER, M. SINTEK, S. TABET, and G. WAGNER. 2006. Schema specification of ruleml 0.91.

HORROCKS, I., P. F. PATEL-SCHNEIDER, H. BOLEY, S. TABET, B. GROSOF, and M. DEAN. 2004. SWRL: A Semantic Web Rule Language combining OWL and RuleML. W3C Member Submission. Available at: http://www.w3.org/Submission/SWRL.

HUYNH, T. D., N. R. JENNINGS, and N. SHADBOLT. 2004. FIRE: An integrated trust and reputation model for open multi-agent systems. *In* Proceedings of 16th European Conference on Artificial Intelligence, Valencia, Spain, pp. 18–22.

JØSANG, A., and R. ISMAIL. 2002. The beta reputation system. *In* Proceedings of the Fifteenth Bled Electronic Commerce Conference e-Reality: Constructing the e-Economy, Bled, Slovenia, pp. 48–64.

JØSANG, A., R. ISMAIL, and C. BOYD. 2007. A survey of trust and reputation systems for online service provision. Decision Support Systems, **43**(2):618–644.

LIANG, Z., and W. SHI. 2008. Analysis of ratings on trust inference in open environments. Performance Evaluation, **65**(2):99–128.

MAXIMILIEN, M., and M. P. SINGH. 2004. A framework and ontology for dynamic Web services selection. IEEE Internet Computing, **8**(5):84–93.

MONTGOMERY, D. C. 2001. Design and analysis of experiments. John Wiley and Sons, Chichester, UK.

MUI, L., M. MOHTASHEMI, and A. HALBERSTADT. 2002. A computational model of trust and reputation. *In* Proceedings of the 35th Hawaii International Conference on System Science (HICSS), Hawai'i, pp. 2431–2439.

PAN, J. Z. 2007. A flexible ontology reasoning architecture for the semantic Web. IEEE Transactions on Knowledge and Data Engineering, **19**(2):246–260.

PAOLUCCI, M., and K. SYCARA. 2004. Semantic Web services: Current status and future directions. *In* Proceedings of the IEEE International Conference on Web Services (ICWs'04), **32**. IEEE Computer Society, Washington, DC.

RETTINGER, A., M. NICKLES, and V. TRESP. 2007. Learning Initial Trust among Interacting Agents. Lecture Notes in Computer Science, **4676**:313–327.

RUSSELL, S., and P. NORVIG. 2002. Artificial Intelligence: A Modern Approach (2nd ed.). Prentice Hall, Englewood Cliffs, NJ.

SABATER, J., and C. SIERRA. 2001. Regret: Reputation in gregarious societies. *In* Proceedings of the Fifth International Conference on Autonomous Agents. ACM, New York, pp. 194–195.

SEN, S., and N. SAJJA. 2002. Robustness of reputation-based trust: Boolean case. *In* Proceedings of the First International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS), Bologna, Italy, pp. 288–293.

ŞENSOY, M., and P. YOLUM. 2007. Ontology-based service representation and selection. IEEE Transactions on Knowledge and Data Engineering, **19**(8):1102–1115.

SINGH, M. P. 1999. An ontology for commitments in multiagent systems: Toward a unification of normative concepts. Artificial Intelligence and Law, **7**:97–113.

SIRIN, E., B. PARSIA, B. C. GRAU, A. KALYANPUR, and Y. KATZ. 2007. Pellet: A practical OWL-DL reasoner. Journal of Web Semantics, **5**(2):51–53.

TEACY, W., J. PATEL, N. JENNINGS, and M. LUCK. 2006. TRAVOS: Trust and reputation in the context of inaccurate information sources. Autonomous Agents and Multi-Agent Systems, **12**(2):183–198.

WHITBY, A., A. JØSANG, and J. INDULSKA. 2005. Filtering out unfair ratings in bayesian reputation systems. The ICFAIN Journal of Management Research, **4**(2):48–64.

YOLUM, P., and M. P. SINGH. 2005. Engineering self-organizing referral networks for trustworthy service selection. IEEE Transactions on Systems, Man, and Cybernetics, **A35**(3):396–407.

YU, B., and M. SINGH. 2003. Detecting deception in reputation management. *In* Proceedings of Second International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS), Melbourne, Australia, pp. 73–80.

ZACHARIA, G., A. MOUKAS, and P. MAES. 1999. Collaborative reputation mechanisms in electronic marketplaces. *In* Proceedings of the 32nd Hawaii International Conference on System Sciences (HICSS-32), Hawai'i, pp. 8026 – 8032.

ZENG, L., B. BENATALLAH, A. H. NGU, M. DUMAS, J. KALAGNANAM, and H. CHANG. 2004. QoS-aware middleware for Web services composition. IEEEd Transactions on Software Engineering, **30**(5):311 – 327.

ZHANG, J., and R. COHEN. 2006. A trust model for sharing ratings of information providers on the semantic Web. *In* Proceedings of the first Canadian Semantic Web Working Symposium, Quebec, pp. 45–61.

ZHANG, J., and R. COHEN. 2007. Design of a mechanism for promoting honesty in e-marketplaces. *In* Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07), Vancouver, BC, pp. 1495 – 1500.

APPENDIX A. OVERVIEW OF THE P2P PROTOCOL FOR DISCOVERING CONSUMERS WITH SIMILAR SERVICE DEMANDS AND COLLECTING RELATED EXPERIENCES

Service consumers record their experiences with service providers using the *experience ontology*. Those experiences are shared among the service consumers and used for the modeling of service providers. Specifically, when a service consumer has a service demand, it uses experiences related to similar demands to estimate which of the providers are more likely to produce a satisfactory service for the service demand. If the consumer does not have enough experiences related to similar service demands, it communicates with other service consumers with similar demands to collect such experiences.

Note that similarity is a subjective concept and may change for each consumer. Therefore, each consumer should express its own similarity metric while searching for the consumers or the experiences related to similar demands. To allow a consumer to express its description of a similar demand, a *SimilarDemand* concept is included in the experience ontology. This concept is a subclass of the *Demand* concept. A service consumer can express what a similar demand is with respect to its similarity criteria using Semantic Web Rule Language (SWRL) (Horrocks et al. 2004). A simple rule for similarity is shown in Figure A1. In this rule, the

```
<ruleml:imp>
 <ruleml:_head>
    <swrlx:classAtom>
       <owlx:Class owlx:name="#SimilarDemand"/><ruleml:var> DEMAND  </ruleml:var>
    </swrlx:classAtom>
 </ruleml:_head>
 <ruleml:_body>
     <swrlx:DataPropertyValue swrlx:property="#hasDeliveryDuration">
     <ruleml:var>DEMAND</ruleml:var><ruleml:var>DURATION </ruleml:var>
     </swrlx:DataPropertyValue>
        <swrlx:individualPropertyAtom  swrlx:property="&ex;#hasShoppingItem">
        <ruleml:var>DEMAND</ruleml:var><owlx:Individual owlx:name="&ex;#book"/>
     </swrlx:individualPropertyAtom>
     <swrlx:predicateAtom swrlx:predicate="..#ifTrue">
        <owlx:DataValue owlx:datatype="..#string">$1 <= 14 </owlx:DataValue>
        <ruleml:var>DURATION</ruleml:var>
     </swrlx:predicateAtom>
 </ruleml:_body>
</ruleml:imp>
```

FIGURE A1. Example SWRL rule for similar demands.

consumer states that a demand is a similar demand only if it concerns a book and requires a delivery duration less than or equal to 14 days.

SWRL is introduced as a way to integrate rules with OWL-DL ontologies. Unlike other rule languages such as RuleML (Hirtle et al. 2006), SWRL is purposely constrained to make automated reasoning more tractable. Hence, using SWRL rules, consumers can represent logical axioms and reasoning on those axioms can be done in a tractable manner. That is, if a consumer has a particular service demand and a list of others' service demands, then it can apply the SWRL rule representing its similar demand definition to select those demands which are similar to that of its own. If the consumer makes its SWRL rule for similar demands public, other consumers can also use this expression of similarity to reason about whether their past service demands are similar to the demand of the consumer or not. A Description Logic (DL) reasoner with OWL support can be used for the reasoning on similarity.

Accordingly, in order to discover others with similar service demands and collect related experiences from those consumers, the consumer distributes its definition of similar demand through the network of consumers. When consumers receive this SWRL rule, they evaluate their service demands with respect to the distributed similarity metric. Then, they send their personal experiences to the consumer if those experiences are related to similar service demands. Moreover, the consumers examine their knowledge about their acquaintances and send the identities of their acquaintances to the consumer if those acquaintances are known to have service demands similar to the service demand of the consumer. Then, the consumer communicates with those acquaintances further to collect related experiences. Details of this P2P protocol can be found in Şensoy and Yolum (2007).