

A semantic query approach to personalized e-Catalogs service system

Donglin Chen¹, Xiaofei Li², Yueling Liang³ and Jun Zhang⁴

Wuhan University of Technology, Institute of E-business, ¹chendl@whut.edu.cn, ²lixiaofeidf@sina.com, ³liangyl1987@126.com, ⁴sky_jun1016@163.com

Received 6 September 2009; received in revised form 17 August 2010; accepted 30 September 2010

Abstract

With the emergence of the e-Catalog, there has been an increasingly wide application of commodities query in distributed environment in the field of e-commerce. But e-Catalog is often autonomous and heterogeneous, effectively integrating and querying them is a delicate and time-consuming task. Electronic catalog contains rich semantics associated with products, and serves as a challenging domain for ontology application. Ontology is concerned with the nature and relations of being. It can play a crucial role in e-commerce as a formalization of e-Catalog. User personalized catalog ontology aims at capturing the users' interests in a working domain, which forms the basis of providing personalized e-Catalog services.

In this paper, we present an ontological model of e-Catalogs, and design a semantic personalized e-Catalog service system (SPECSS), which achieves match user personalized catalog ontology (UPCO) and domain e-Catalog ontology (DECO) based on ontology integrated and focus on four key technologies: user personalized catalog ontology generation, domain and local e-Catalog construction, semantic match between them and e-Catalog semantic query system based on heterogeneous catalog database.

Key words: e-Catalog, personalized ontology, semantic match, semantic query, e-Catalog service

1 Introduction

Electronic catalog is the aggregation of the static WWW pages including product information which is an essential component in e-commerce scenarios. It contains information of product trade, such as pricing, features, and terms between partners. Clearly defined product information is a necessary foundation for collaborative business processes. Furthermore, semantically enriched product information may enhance the quality and effectiveness of business transactions. As a multifunctional applied system, it serves for advertisement, marketing, selling and client support, and at the same time it is a retail channel.

The Web has experienced continuous growth since its creation. As the number of Internet users and the number of accessible Web pages grow, it is becoming more and more difficult for users to find documents among e-Catalogs that are relevant to their particular needs. Users can search with a search engine which allows users to enter keywords to retrieve e-Catalogs that contain these keywords. The navigation strategy and search have their own problems. Indeed, approximately one half of all retrieved documents have been reported to be irrelevant [1]. The main reasons for obtaining poor search results are that (1) many words have multiple meanings [2]; (2) key words are not enough to express the rich concepts and the natural semantics of customers' queries; (3) the property query lacks of semantic support, and is difficult to search for knowledge, and has other problems of mechanisms. Therefore, we cannot query the united queries based on different classification rules; (4) related merchandises can not be returned, such as querying "Memory" and "board". For people, the concepts are not isolated; there are kinds of interrelations between the concepts [3]. What is needed is a solution that will *personalize* the e-Catalog selection and be presented to each user.

The need for domain ontology development and management is becoming more and more important to knowledge-driven e-Catalog services. That is, ontology can play a critical role here which can be considered as a unified conceptual graph that represents concepts and relationships among the concepts existing in an application domain. A semantically rich user model and an efficient way of processing semantics are the keys to provide personalized e-Catalog services [4]. In view of the existing limitations, we develop a personalized ontology based on user model, called user personalized catalog ontology, which has the same level of semantics as domain ontology.

The rest of this paper is structured as follows: Section 2 describes related work. In Sections 3, we put forward our modeling methodology for generating user personalized catalog ontology and domain e-Catalog ontology. In Section 4, we introduce the crucial algorithm of semantic match for our SPECSS. And Section 5 explains the theory of semantic personalized e-Catalog service system (SPECSS). Then in Section 6, we present the implementation of the system and its evaluation. Conclusion and future work are drawn in Section 7.

2 Related Work

As to semantic query issue of e-Catalog throughout e-commerce scenarios, much work has been done in the stages of semantic query based on e-Catalog, including e-Catalog standardization and integration, personalized catalog service research and e-Catalog ontology design.

2.1 E-Catalog Classification Standards

Construction e-Catalog is based on product classification standards. The Chinese Standardization Research Institute implements general product classification standard based on XML [5]. Some of these XML standards are compared by Li [6], who discusses seven different product description standards used in e-commerce scenarios, their complexity and potential integration problems, but only on the basis of very simple metrics. International product classification standards—UNSPSC, eCl@ss, RosettaNet, EGAS, NCS, etc—have established a comprehensive classification level [7]. [8] Presents a practical model for eCl@ss and touches upon the practical issues for classification schemes. These classification standards are different with each other in structure and content, and lack semantic service ability as well.

1. There are many related concepts of e-Catalog, such as Product Catalogue, Electronic Product Catalog, Products and Services Catalog, and Product Classification;
2. Lack of effective integration between the standards, and customers must have certain professional knowledge about international catalog standards in order to search the most relevant results;
3. The product classification was assumed to be the simple code-based hierarchical model such as UNSPSC or eCl@ss.

2.2 E-Catalog Ontology Design

Researches in recent years show that applying ontology to e-commerce scenarios would bring benefits such as solving the interoperability problems between different e-commerce systems [8]-[11]. Especially, e-Catalog, which is a key component of e-commerce systems, seems to be the most adequate domain within e-commerce scenarios where ontology can realize the expression of e-Catalog on a semantic level. It is possible for e-business systems to offer diverse interoperable services by sharing well-defined e-Catalog model containing rich semantics. Fensel [12] described in principle how ontologies can support the integration of heterogeneous and distributed information in e-commerce scenarios which is mainly based on product catalogs, and what tasks are needed to be mastered. E-Catalog ontology model is defined as ECO (concepts, relationship, properties, axioms and individuals). At present, there are two methods to build ECO:

2.2.1 Manual definition based on the classification standards

Australian scholar Martin Hepp proposed the method to generate domain catalog ontology in [13], who issued eCl@ss domain catalog ontology based on OWL, and used the parent-child structure of classification standard to obtain is-kind relation between the concepts of catalog ontology. South Korean scholar Hyunja Lee [14] manually enriched the semantics of e-Catalog ontology by increasing product properties and metric units and pointed out that products, classification scheme, properties and UOMs as the key concepts and developed a framework in which various types of relationships existing in product ontology are exploited for the score propagation in [2]. At least five deficiencies exist:

- Have not considered the method to extract e-Catalog ontology from distributed enterprise product catalogs (database);
- Have not studied the syntactic and semantic integration problems of distributed e-Catalog;
- Have not yet researched the integration of international classification standards;
- Manually defined domain e-Catalog ontology, and included product properties, but have not involved extracting individuals;
- Domain e-Catalog ontology service is only for customer query, but not considering personalized e-Catalog service. [15] designed retrieving metadata semantics-based IPIS service system. However, this method is not suitable for dynamic changes of product information for the e-commerce scenarios.

2.2.2 Automatic or semi-automatic generation methods of catalog ontologies

The method extracts local catalog ontology from enterprise product information system (such as ERP), Web site formation. Obrst, Wray and Liu [16] discuss the main challenges of building and aligning ontologies for products and services in B2B e-commerce scenarios. [17] designed DOME, an enterprise e-Catalog ontology management system structure, and put forward building standardized local e-Catalog ontologies from enterprise information systems mapping with UNSPSC classification standard. This method just established a centric e-Catalog structure, but omitted the syntax and semantics integration among distributed e-Catalogs and the integration of international classification standards.

2.3 E-Catalog Ontology Integration

E-Catalog integrations include syntactic integration and semantic integration. We need to integrate international product classification standards, and enterprise e-Catalogs or product classification databases. There are more than 25 kinds of e-Catalog classification standards in the world. [18] provided a catalog management method for the various classification standards. The standard integration XML-based can solve syntactic and structural heterogeneity, for example, CEN/ISSS(European Committee for Standardization/Information Society Standardization System) put forward e-Catalog standards integration [3]. Corcho and Gómez-Pérez [19] also show how multiple standards for classifying products and services can be integrated using ontological mappings, and sketch a prototype implementation based on the WebODE platform. Catalog ontologies based on various international product classification standards are heterogeneous [20], and then the key of e-Catalog semantic integrations becomes catalog ontologies integration. There are three methods for current ontology integration [21] (1) schema integration based on ontology properties; (2) integration of ontology concepts; (3) multi-layer structural integration of ontologies (individuals, properties, concepts), such as our research group put forward semantic similarity products match algorithms based on ontology concepts in combination with properties[22], [23]. E-Catalog integration can use [24] for reference. [25] applied the mapping method of ontology concepts and properties to integrate global data synchronization networkGDSN and EPCglobal catalog ontologies.

2.4 Personalized e-Catalog Services

Within unstructured data a keyword search engine can do a very valuable job like in Google.com or Baidu.com. But this technique does not utilize the semantics available in structured data. Moreover, it has lots of problems with the syntax, despite the semantics of typed e-Catalogs is clear. E.g., in the ranked keyword search XRANK [10] would cause a lot of problems. Also exploiting structured and typed data, parametric search aims to find the right alternatives in case there is no perfect match [11]. Iteratively, the user can soften or skip some search conditions. The most problematic deficit in this technology is that there is no deterministic way, indeed no confirmed way at all to find the best alternative. The user never knows when it is best to terminate the search and with which result. That is, traditional key-based retrieval method can not satisfy massive heterogeneous personalized catalog service, then [26] introduce metasearch engines, but this method is passive service. [27] provided an intelligent catalog recommend method using customer requirements mapping with product categories. [28], [29] researched personalized catalog ontology service. [28] brought forward personalized e-Catalog model based on customer interests and [29] is a personalized catalog service community, WebCatalog. [30] designed enterprise e-Catalog based on customer behavior. The knowledge representation and acquisition of client catalog turns into the key problems. In order to reach an effective method, K-clustering algorithm and e-Catalog segmentation approach are described in [31], and [32] described the customer segmentation method based on brand and product, price. In [33], the author researched personalized catalog service with one-to-one market by association rules and CART. In recent years, personalized ontologies (also known as private ontology, such as [27]) are introduced into e-Catalog service, Peter Haase put forward personalized ontology learning theory based on user access and interest coordination [34]. In distributed system, there are sharing concepts of domain ontologies and personalized knowledge ontologies [35]. Therefore, it has important theoretical and practical significance to apply personalized ontologies to personalized e-Catalog service.

As to this requirement, in this paper, SPECSS focuses on personalized catalog service and e-Catalog ontology construction in order to provide semantic e-Catalog query. Therefore we build user personalized catalog ontology based on consumers' behavior matched with domain e-Catalog ontology through semantic match model. The match result set is used for semantic query system.

3 Methods of Building E-Catalog Ontology

This study proposes a foundation for personalized e-Catalogs service by constructing two catalog ontology. One of them is user personalized catalog ontology which provides users' personal information and preferences and the other one is domain e-Catalog ontology that provides us standard e-Catalog knowledgebase.

3.1 User Personalized Catalog Ontology

In order to satisfy customer's personalized requirement, we should master more information of the customers. And it is not enough that we only construct domain e-Catalog ontology from semantic dictionary and international classification standards (such as eCl@ss, UNSPSC). Sometimes customers also cannot describe their own thought, to understand their potential mind, we need a user e-Catalog ontology. Based on consumer behavior, we propose a personalized approach to build user personalized catalog ontology (UPCO).

- First, build user ontology backbone (UOB) based on users' personal information and preferences;
- Second, extract user catalog information from user purchase history, user searching keywords, user browsing catalog, user feedback information and experts recommendation by catalog information extracting module;
- Third, web resource semantic processing module is convenient for providing personalized services, according to user catalog ontology information, such as classifying web resource and formatting process.

We can export personalized catalog information from personalized user ontology, which can be sorted, as well as classify users' interest. The process is described in figure 1.

In the project SPECSS, we establish UPCO, which is general in domain application. When we need to apply it to specific areas, we can build the corresponding catalog ontologies in the fields according to the general user personalized catalog ontology framework, and combine users' personalized request with e-Catalog semantic queries, which can upgrade users' interest from the level keyword-based to a knowledge-based level, and link with keywords by semantics, because the UPCO describes users' interest in customers' view, and is closer to the users' real thoughts. SPECSS organizes a group of keywords expressing users' interest through UPCO, when users put up semantic query, it is no longer a simple keywords match, but considering users' personal preference and information, and tightly integrates the users and products, so that the system can improve the semantic query precision rate and recall rate, as well as be conducive to sort query results.

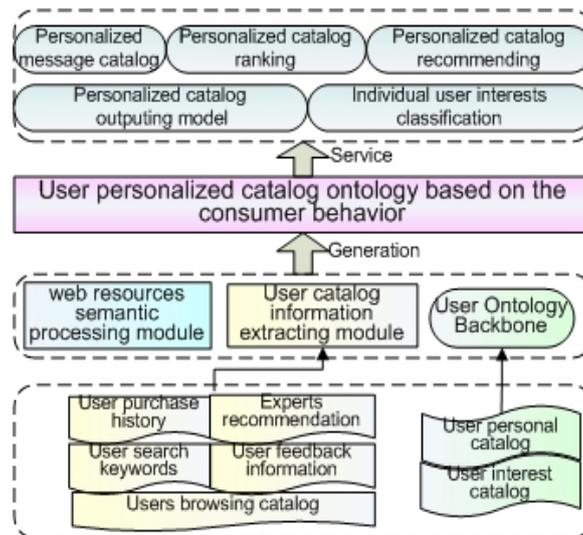


Figure 1: Generating User Personalized Catalog Ontology

Figure 2 shows a user personalized catalog ontology framework, in which we describe user information, user preference and product concepts, properties and individuals that users are interested in, including product area, brand and quality authentication. Users associate with the product by property hasPreference, and we set aside a weight interface in property "hasPreference", indicating the fact users' different observation extent about different properties of a product which is shown in Figure 3.

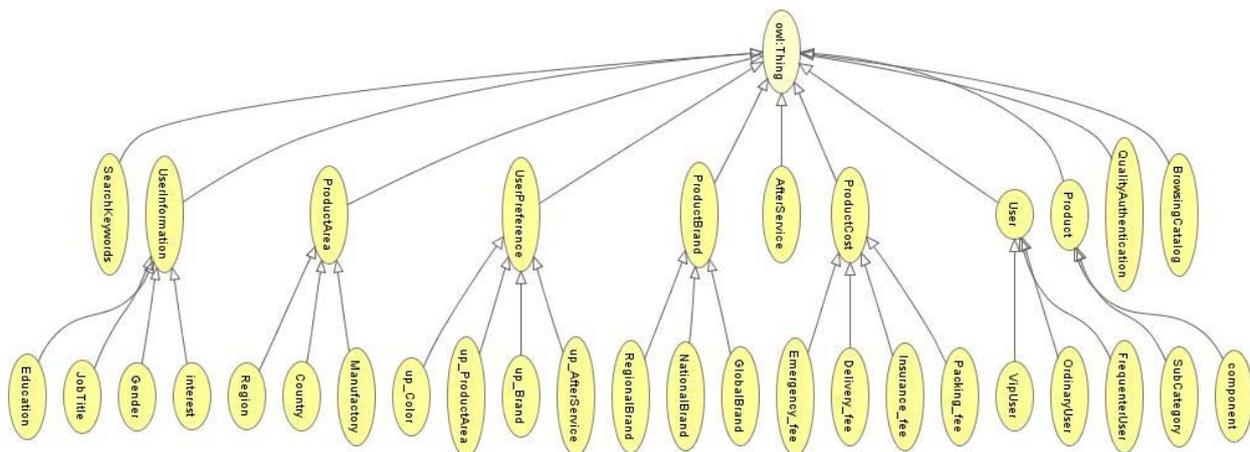


Figure 2: The framework of user Personalized Catalog Ontology

3.2 Domain e-Catalog Ontology

- Generation domain e-Catalog ontology (DECO): Generation domain e-Catalog ontology is divided into four steps: ① Extraction of the core concepts and properties for domain e-Catalog ontologies, according to the UNSPSC and eCI@ss standards, wordNet standards and semantic catalog dictionary. ② Construction of a DECO model. ③ Definition DECO and storing them into catalog warehouse through user-defined DECO subsystem. ④ Acquisition standardized DECO by e-Catalog ontology pruning subsystem, combining wordnet and semantic catalog dictionary (see figure 4).
- Generation local e-Catalog ontology : Local e-Catalog ontologies are built based on heterogeneous distributed database, first, we analyze catalog database mode, and extract its ER model and convert the ER model to initial local e-Catalog ontology after defining the mapping rules. Then, we need standardize initial local e-Catalog ontology by standardization module, as in figure 5.
- Integration initial domain e-Catalog ontology and local e-Catalog ontology to acquire domain e-Catalog ontology DECO

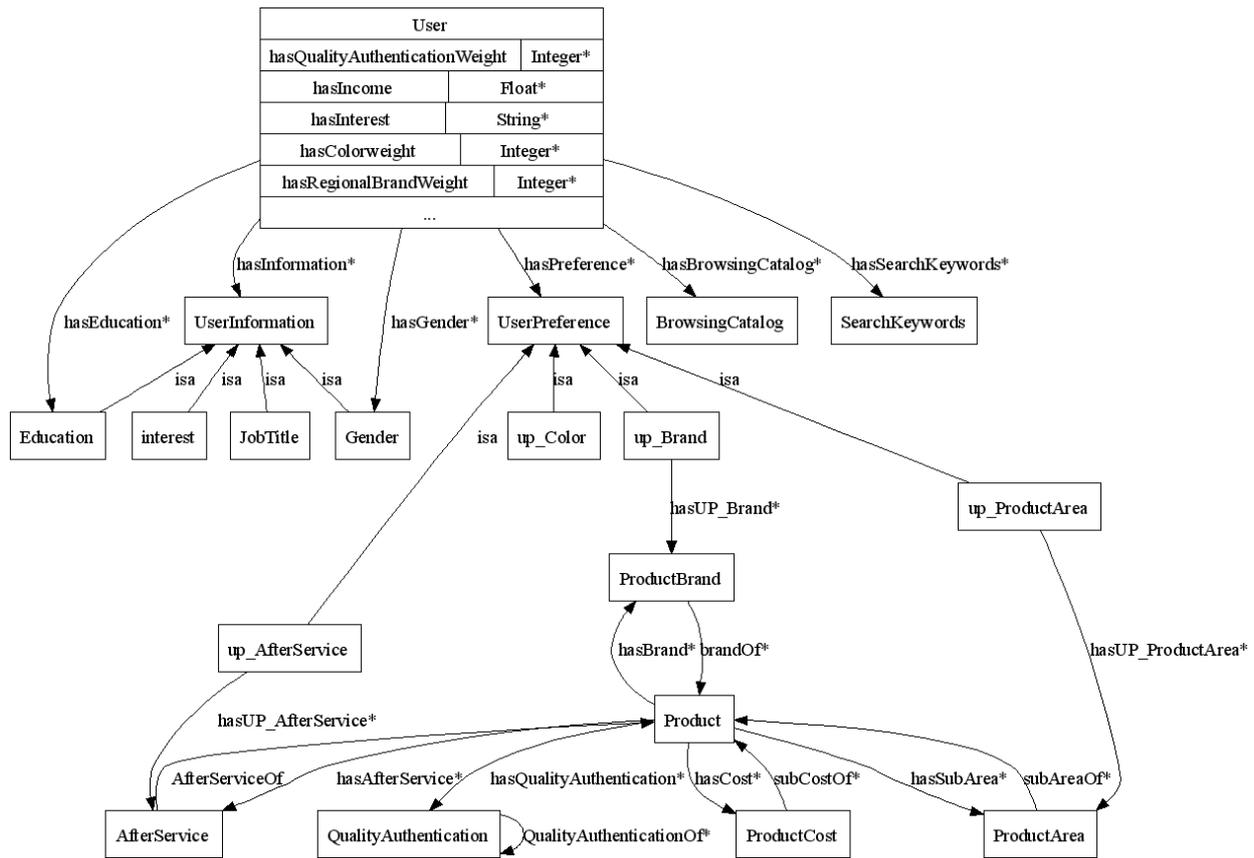


Figure 3: The Relationship of user Personalized Catalog Ontology

We have researched on the technology of database schema-based semi-automatic generation domain catalog and Chinese e-Catalogs Semantic Dictionary in [36], here it is only a framework and in [37], our group has built the mapping rules from the relational database schema to local ontology.

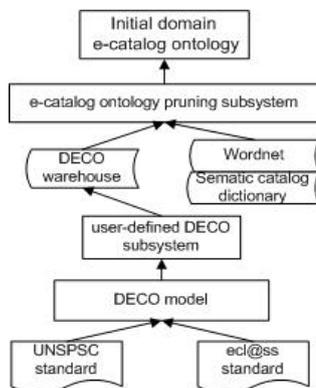


Figure 4: The principle of DECO Generation

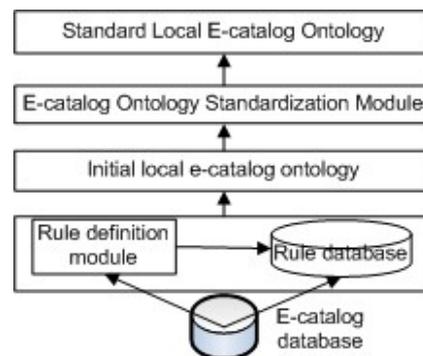


Figure 5: The Principle of LECO Generation

4 Semantic Match Based on Ontology

One critical step of semantic match is that calculation semantic match degree between the terms of ontology concepts. There have been many methods to calculate conceptual semantic match in e-commerce scenarios [38]. Common calculation methods and models are: (1) Identifier-based method [39], which uses word-building to find the semantic match degree between the concepts, and primarily reflects the linguistic similarity of the two concepts; (2) Synonym dictionary-based method [40], which organizes all concepts to a tree hierarchy structure according to synonym dictionary where there is only one path between any two nodes and this path length is taken as a measure of semantic distance of the two concepts; (3) Feature Match-based model [41], which calculates semantic match of concepts by the collection of properties; and (4) Semantic relationship-based model [42], also known as the semantic

distance-based model, which calculates semantic match of concepts based on hierarchy information and is mainly used in the same ontology.

In this paper, we need to calculate the semantic match of UPCO and DECO, and adopt identifier-based method to calculate the semantic match of names of concepts, and property feature-based clustering analysis to calculate the semantic match of concept properties, and property value-based method to calculate the semantic match of concept individuals, and at the end, we get the final ontology semantic match result sets by integrating the above results. In the following, we give an overview of the approach. First, semantic match module calculates concept-based semantic match degree, if the concept value is no less than the set threshold directly, it would become candidate 1. Second, the module will calculate property-based semantic match degree, and then we get candidate 2 or it does not exist, the same as candidate 3 which comes from individual-based semantic match. The next step, we need to integrate the candidates to gain the match result sets. Figure 6 shows this process.

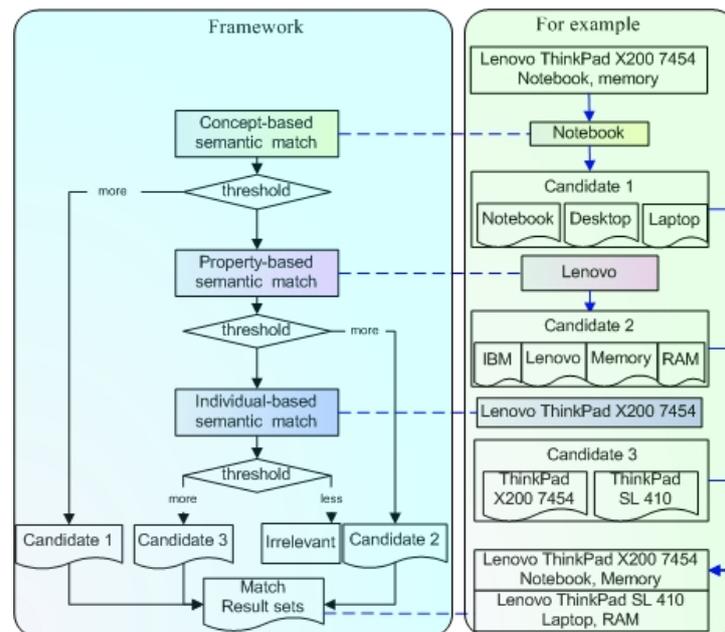


Figure 6: The Overview of Ontology-based Semantic Match Approach

4.1 Concept-based Semantic Match

In this paper, we adopt synonym set-based approach to calculate the semantic match degree of the concepts in the UPCO and DECO, which would make use of the Semantic Catalog Dictionary. If two ontology concepts have the same or similar characters, they usually have the same or similar means. However, if the naming rules are inconsistent with each other in the DECO and UPCO, the characters of the same semantic concepts may be completely different, then we might get zero match degree. Therefore, this paper puts forward a method that semantically expands the ontology concept into a concept aggregation according to the synonym set of the concept in the definition of ontologies (Same As relationship) and Wordnet before identifier-based calculation the semantic match of names of concepts. That is, we calculate semantic match degree of the synonym set concepts C_1 and C_2 , except for calculating semantic match degree of them.

The algorithm is as follows: Calculate the semantic match degree of each element c_i of the synonym set of C_1 and each element c_j of the synonym set of C_2 the identifiers-based, and then take the maximum value as the semantic match degree of C_1 and C_2 , that is $sm(C_1, C_2) = \max(sm(c_i, c_j))$.

4.2 Property-based Semantic Match

Property-based semantic match method respectively calculates the semantic match degree of datatype and objective properties ($smd(P_1, P_2)$ and $sno(P_1, P_2)$) and then sets weight for the semantic match degree of the two kinds of properties, and at last, integrates them to gain the semantic match based on property. This work has been developed in our earlier work [21] and are motivated and defined as following. The motivation is to take into account the degree of specificity of the properties, based on the fundamental idea that a property being used very frequently is generally less specific than a property assigned to only a few e-Catalogs and the weight given by the users in the UPCO. That is, we modify weight setting method, while first of all, we need process normalization properties and adopt k-medoids algorithm, as follows:

Function k-medoids (Dataps, K)
 // Dataps: set of datatype properties
 // K: maximum number of clusters

The datatype properties of concept C_1 as $P_1(P_{11}, P_{12}, \dots, P_{1i}, \dots, P_{1m})$;
 The datatype properties of concept C_2 as $P_2(P_{21}, P_{22}, \dots, P_{2i}, \dots, P_{2n})$;
 The weight of the properties as $W_1(W_{11}, W_{12}, \dots, W_{1i}, \dots, W_{1m})$ and $W_2(W_{21}, W_{22}, \dots, W_{2i}, \dots, W_{2n})$;
 The datatype property set $P(P_1, P_2, \dots, P_m, P_{m+1}, P_{m+2}, \dots, P_{m+n})$;
 // $P(W_1P_1, W_2P_2, \dots, W_mP_m, W_{m+1}P_{m+1}, W_{m+2}P_{m+2}, \dots, W_{m+n}P_{m+n}) = P(W_{11}P_{11}, \dots, W_{1m}P_{1m}, W_{21}P_{21}, \dots, W_{2n}P_{2n})$
 Arbitrarily selecting K datatype properties as $O(O_1, O_2, \dots, O_i, \dots, O_k)$;

Calculate $d_w(O, P) = \sum_{j=1}^{m+n} w_j \delta(p_{ij}, p_{ij}), l, i = 1, 2, \dots, m+n$;

// The cost function:

$$E(U, P) = \sum_{l=1}^k \sum_{i=1}^{m+n} u_{li}^\alpha d_w(O_l, P_i), u_{li} \in [0,1], \sum_{i=1}^k u_{li} = 1$$

For($i=1$ to $m+n$)
 Calculate $d_{i1}, d_{i2}, \dots, d_{ik}$;
 If $d_{i1} = \min(d_{i1}, d_{i2}, \dots, d_{ik})$
 $(p_i) \rightarrow O_1$;
 Else if $d_{ij} = \min(d_{i1}, d_{i2}, \dots, d_{ik})$
 $(p_i) \rightarrow O_j$;

// Update the cluster center:
 $\sum_{i, p_j=O_j} u_{li}^\alpha$ achieves the maximum in each property j;
 Output: $P(P_1, P_2, \dots, P_i, \dots, P_k)$

The extended approach categorizes the properties of concept C_1 and C_2 in its type, then the properties can be divided into several property sets according to Function k-medoids and the pre-setted weight. Then the weight (W_d) equals

$$W_d = \frac{1}{\text{the numble of datatype properties cluster}} \text{ and}$$

$$W_o = \frac{1}{\text{the numble of objective properties cluster}}$$

$$smp(C_1, C_2) = W_d * smd(P_1, P_2) + W_o * smo(P_1, P_2)$$

4.3 Individual-based Semantic Match

To query user preferred product, we should get the product similar with user preferences, namely calculating the instance similarity between DECO individual and UPCO individual. We calculate the semantic match of the individuals by the property value-based method.

Calculate the semantic match method based on linguistics, when we calculate semantic match degree of the property values.

$$smi(C_1, C_2) = \frac{ed(C_1, C_2)}{(|C_1| + |C_2|) / 2}$$

Explanation:

- $|C_1|$ is the length of the string C_1 , $|C_2|$ is the length of the string C_2 , $ed(A, C_2)$ is the same number of characters in C_1 and C_2 .
- String C_1 and C_2 are input parameters, in the process, which are the properties values of two products, such as GlobalBrand lenov and Lenov_China in Figure 7.
- calculate the individual semantic match of the two products through comparing several groups property semantic match degree.

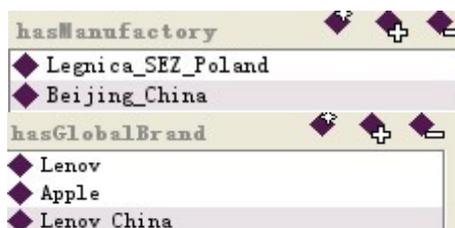


Figure 7: The Property Values of Individual

And the semantic match degree of C_1 and C_2 :

$$SM(C_1, C_2) = w_1 sm(C_1, C_2) + w_2 smp(C_1, C_2) + w_3 smi(C_1, C_2)$$

5 Designing Personalized e-Catalog Service System

To sum up, implementation distributed e-Catalog semantic query, first of all, personalized catalog ontologies are customized e-Catalog ontologies according to consumers; secondly, we need to build domain e-Catalog ontologies (DECO) and establish local e-Catalog ontologies (LECO) extracted from heterogeneous databases; thirdly, we match the two kinds of ontologies by match algorithm through semantic match module which generates match resultsets. The semantic match resultsets are repositories which implement query reasoning and expanding module in SPECSS, as well as the basic of semantic tagging to information resources.

The structure of the SPECSS is divided into four parts distributing e-Catalog-based:

- User personalized catalog ontology: a customized e-Catalog ontology, extracted from web resources, user ontology backbone and user catalog information;
- Domain e-Catalog ontology: the foundation of e-Catalog semantic query reflects the semantic mapping relationship between e-Catalog databases and domain e-Catalog.
- Semantic match: if we say domain e-Catalog ontology is woods, UPCOs are leaves, and therefore, we must match the "leaf" within "woods", to gain the shared concepts, properties and individuals.
- E-Catalog semantic query engine: receives query conditions that users input in querying interface, and translates them into ontology describe which semantic query engine can understand. Then, returns query result sets by means of reasoning and expanding module and interacts with the distributed database.

The basic theory of distributed semantic query based on e-Catalog ontology is: users input key words, phrases, sentences or paragraphs (users' queries, U_q) in user querying interface; query generator module of SPECSS translates U_q to ontology describe; query reasoning and expanding module is responsible for reasoning and expanding the describe using the semantic match result sets, then outputs semantic queries (S_q) in forms of Sparql [43]; further more, query disassembling module disassembles S_q into the local e-Catalog ontologies and finally extract data from distributed e-Catalog database, at this time, re-writing S_q s to SQLs based on different e-Catalog databases is necessary, then querying results will be created. Query combining and filtering module combines the distributed results and filters repetitive and invalid results; semantic packing module will repack the querying results in the form of SQL to generate the final result set which is in the form of ontology and recommend them to users. Figure 8 depicts the semantic process.

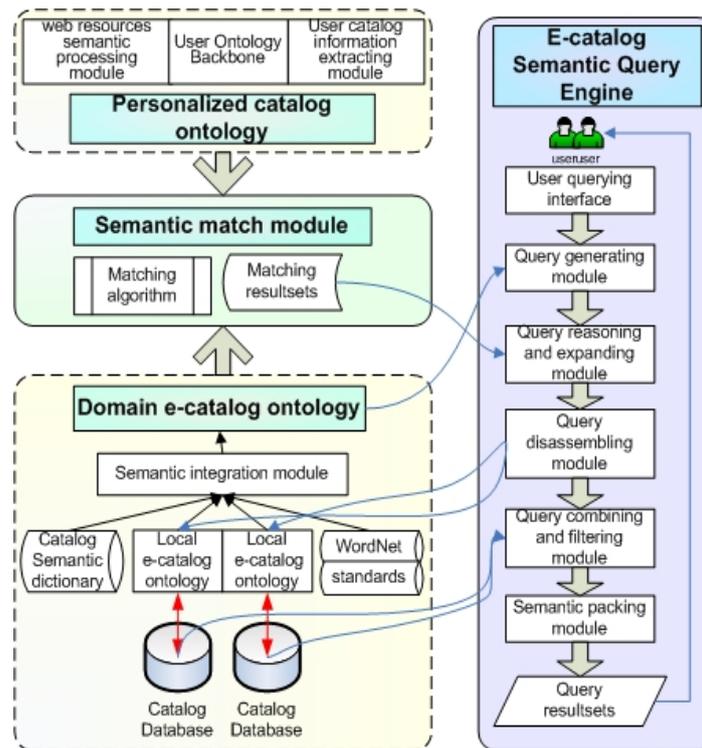


Figure 8: Diagram of e-Catalog Semantic Query Based on Distributed Database

- U_q : the keywords, phrases, sentences or paragraphs that users input in the querying interface;
 $U_q = \{a \mid a \in Sw \parallel a \in Ph \parallel a \in S \parallel a \in P\}$, where Sw: searching keywords, Ph: phrases, S: sentences, P: paragraphs;
- S_q : the generating query sentences according to U_q in the form of Sparql; $S_q = \text{select ?a from DECO}$.

5.1 Query generating module

This portion is responsible for receiving users' queries from querying interface, and converting U_q to a unified, identifiable Sparql. Example 1 : "Show me a list of laptops, made by IBM, with at least 1GB of memory, 80GB HDD and more than 1 year warranty"

First SPECSS establishes DECO and LECO based on the above approach, and combines with UPCO, to generate descriptions of ontology according to WSML language [44] as follows:

```
namespace { _ "http://example.org",
            tasks _ "http://example.org/ontologies/tasks/" }
goal GetBrandInformation
annotations
    dc:description hasValue "Describes the desire of getting brand information of IBM notebook product"
endAnnotations
ontology ComputerProducts { _ "http://example.org/ontologies/products/ComputerProducts" }
    concept Notebook
        hasManufacture ofType manufactory
        hasBrand ofType brand
        hasMemory ofType xsd#integer
        hasHD ofType xsd#integer
        hasWarranty ofType xsd#decimal
importsOntology User { _ "http://example.org/ontologies/user/" }
    concept user
        hasPreference ofType userPreference
        hsaInformation ofType userInformation
```

The above description uses the turtle syntax for representing RDF [45], [46].

@prefix tasks:	<	http://example.org/ontologies/tasks/#>.
@prefix np:	<	http://example.org/ontologies/products/ComputerProducts/Notebook#>.
@prefix u:	<	http://example.org/ontologies/user/#>.
?brand	np:p	IBM
?memory	np:p	memory
?warranty	np:p	warranty
?HD	np:p	HD
?type	np:	type

A RDF graph can be represented as a “tree” with three columns, ?subject, ?predicate and ?object. Each row corresponds to one triple. Similarly, the result of a SPARQL SELECT query is a tree of RDF nodes. These trees are RDF relations. Query generating module consists of operators on RDF relations [47]. Some of them are well-known from regular relational algebra, others are slightly modified to reproduce SPARQL semantics.

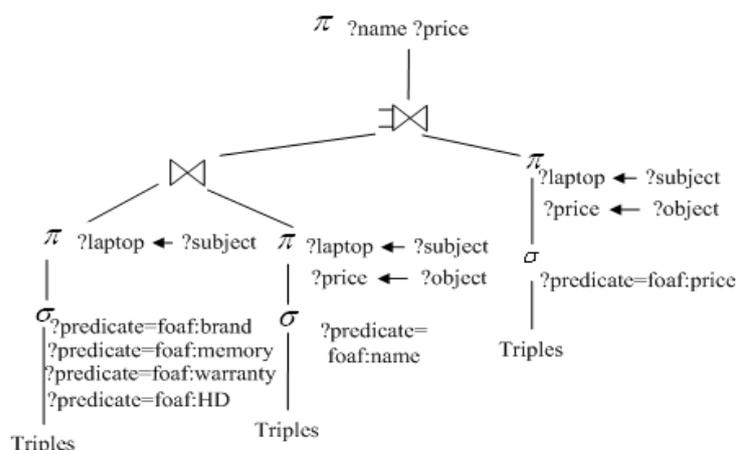


Figure 9: A SPARQL Query and its Translation into a Relational Operator Tree

5.2 Query reasoning and expanding module

Compared with traditional query, the largest characteristics of semantic query are that it introduces expansion reasoning functions into the users’ queries in the querying course. It mainly reasons and expands the users’ queries to gain Sq which are associated with Uq.

When we recommend semantic information, we first query the visited records in order to quickly find users interested goods which have strong correlation with what users want.

For example 1, we know that there may be hundreds of thousands of kinds of IBM notebook satisfying the conditions, and if we return all the results to them, it greatly reduces the system efficiency and user satisfaction. SPECSS extracts user preferences and basic information personalize the user ontology-based to expand the user queries. Such as the user may be a university student, and he may prefer the metal shell and the appearance of blue waves. We focused on the income of university students, and limit ?price below 6000RMB, ?color prioritizing blue, and ?shell for the metal material.

5.3 Query breaking module

In order to reduce the complexity of the search, we also need further break down Sq into several atomic queries, and then deal with each of the atomic queries in LECOs, which is the work of query breaking module. Of course, how to determine which LECO should be appropriate and how to fast position these LECOs are the main problems that this module needs to solve. We consider we can make use of OS finding file algorithms and adding name space to mapping table to solve these problems.

5.4 Query re-writing module

Query re-writing module: Reasoning and expanding are extended to deal with queries in the middle of a process, but the final results are from the relational database. Therefore, re-writing query semantic to query terms from relational database is a necessary step. Rewriting e-Catalog semantic query bases on the principle of building LECOs which is

based on catalog database, reverses building LECOs process, restores the database in the form of tables, then makes a query with SQL.

5.5 Semantic packing module

Semantic packing module: The process puts up query results of relational database in semantic ontologies, then submits to combining and filtering results module, in order to get the final results to return to the users.

Here, SPECSS will return search results based on the size of the importance, along with the semantic annotations for the convenience of users' choosing. Alternatively, we can amalgamate a number of ontologies pointing the same individual as integrated ontologies, which can be utilized to annotate resultsets. Since the ontology contains a number of similar ontologies, so its comments reflect the similar viewpoints of the majority people, which can help users understand.

5.6 Combining and filtering results module

We need to further compound result ontologies from last module to user-oriented semantic result ontologies. This module is the solution of combining ontology and plays a role as filtering out irrelevant ontologies.

Semantic result ontologies are composed of two or more LECOs and their related ontologies. There is overlap among the properties, methods, or the relationships, that is, the semantic relationship degree $SRD(O_1, O_2) > 0$. Regard related ontologies as basic ontologies, LECOs as expanding ontologies.

Algorithm of e-Catalog- combining and filtering:

Constructing semantic results SR, where AO is basic ontology, local e-Catalog ontologies $O_1, O_2 \dots, O_m$ are expanding ontologies, SRD is r.

```
Input: LECO  $O_1, O_2 \dots, O_m$ , basic ontology AO;  
Output: semantic results SR;  
Begin  
  for( $O_1, O_2 \dots, O_m$ ) {  
    copy AO, $O_i$  ( $i=1,2\dots m$ ) to the SR;  
    finding domain ontology DO mapping  $O_i$ , according to the semantic mapping table; }  
  for(sub-ontology s in DO){  
    if( $R_D^W(O_i,s) \geq m$  && s isn't in DO)  
      copy the components of s to SR;}  
  return SR;  
End
```

The basic operation is calculating SRD and copying sub-ontologies. Assuming the average number of sub-ontologies in DO is n, the number of LECO is m, we can get the algorithm complexity is $O(mn)$.

6 Implement and Experiment

SPECSS is implemented in Java and Jena2 API. Jena2 [48] is a Java framework to build the semantic web application program, provide the programming environment for the languages like RDF, OWL, SPARQL, etc., and include rule-based inference engine. Therefore, we use it to process ontologies, and carry out SPARQL query.

The system was evaluated by having five users implement the system to create personal ontologies. Each user was asked to provide his/her personal information and preference, especially give the different weight on different properties, as shown in Table 1. Our experiment was carried out on five PCs with Windows XP, Pentium(R) D CPU 2.80 GHz, and 1G RAM, in order to simulate heterogeneous environment. The user was given a query interface to input his/her query parameters and view each one of their concepts and every concept from the DECO that had been matched to the personalize catalog concept. Also the user was able to decide which concept or property was not needed when reasoned and expanded the query. In the experiment, we take computer domain as an example. The user was asked to compare the semantic query result and that from the keyword-based search engines and decide if SPECSS was the better. Therefore, we manually create the domain e-Catalog ontolog (DECO) and user personalized catalog ontology (UPCO) and calculate semantic match degree in the system as in Figure 10 and 11.

Table 1: The Statistics of Experimental User Personalized Catalog Ontology

name	gender	age	academy degree	job title	interest	favorite computer brand	Concerned Properties	searching keywords
Xiaofei Li	female	23	master	e-commerce	playing ping pang and badminton; surfing	Lenov; Apple;HP; Dell	Price 5; Brand 4; Price/ Quality 4; warranty 3; Pay Method 3; off 3; Delivery Method 2; condition active(new/used/ refurbish)1	intelligent;product and service; personalized service;Pink laptop, price is less than \$1000
Yueling Liang	male	23	master	e-commerce	music; movie; sport	HP; ThinkPAD	Price 5; Brand 5; Price/Quality 3; warranty 3; condition active (new/used/ refurbish) 2	product and service; personalized catalog ontology; laptop;brand ;comment
Jun Zhang	male	24	master	e-commerce	movie; information technology; sports	ThinkPad; Apple;HP	Price/Quality 5; Price 4; Brand 4; warranty 3; Delivery Method 3; condition active (new/used/ refurbish) 2	laptop;brand ;comment; new type and price between \$800 and \$1200
Shangying Xu	female	24	master	e-commerce	music; reading; movie; playing ping pang and badminton	IBM;Dell	Price 5; Brand 5; Price/Quality 5; warranty 4; condition active (new/used/ refurbish) 2	price; intelligence; product and service; personalized service; performance price ratio
Peilu Zhang	female	24	master	e-commerce	music; movie; sports; shopping	Apple	Price 5; Brand 4; Quality 4; appearance 3; warranty 3; off 2; after sale service 2; condition active (new/used/ refurbish) 1	brand;color; performance price ratio;product comments; relative services; price is less than 5000 yuan

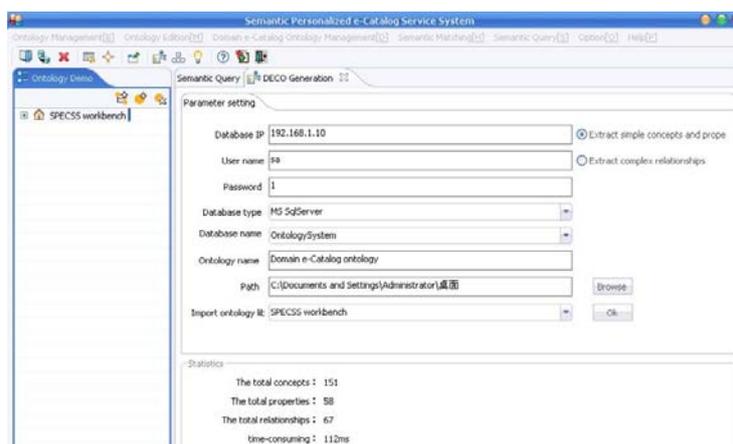


Figure 10: The Interface of DECO Generation

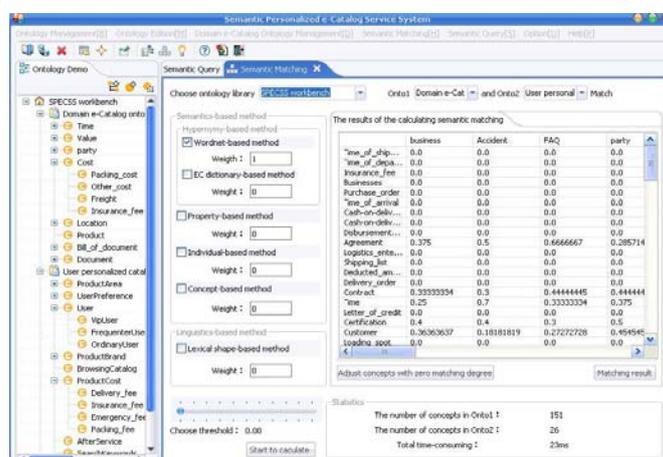


Figure 11: The Interface of Semantic match in SPECSS

In SPECSS, we first import domain e-Catalog ontology, then user can manage DECO and edit UPCO through simple catalog ontology edition model which is shown in the right of SPECSS. For example, user can choose source ontology from DECO, and customize the ontological concepts, properties, domain and range of the properties, or delete them.

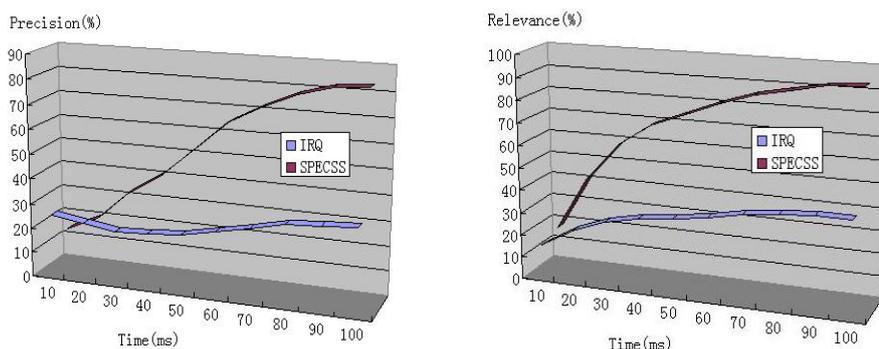


Figure 12: Precision and Relevance Comparison Between SPECSS and IRQ

We evaluated the system with two measures, precision and relevance, as shown in Figure 12. Precision measures the number of relevant pages that were seen vs. the total number of pages that were seen. Relevance measures the number of relevant pages seen plus the number irrelevant pages not seen vs. the total number queried.

7 Conclusion and Future Work

In this paper, we have provided complementary contributions to related work on applying ontology into e-Catalog and traditional query. We focus on the theory of e-Catalog semantic query and personalized catalog service, which can express the preference and potential intention of users while they search products, including UPCO and DECO

Construction, and semantic match algorithm. In addition, we propose a system named SPECSS for customers and introduce each layer and module of it, especially Query Generation Module. In the end, we implement SPECSS system and have five user take experiment in order to evaluate its performance.

In future work, we will focus on: (1) automatically learn e-Catalog ontological concepts, properties and relationship from web to build UPCO; (2) add business properties besides general properties to DECO; (3) construct the Reasoning and Expanding Module of SPECSS, to set rules onto DECO; and (4) extend SPECSS system to web information search, not only heterogeneous catalog databases.

Acknowledgments

This paper is supported by the key project of 863-Plan 2009AA043508, National Natural Science Fund 70972094, Humanities and Social Sciences 09YJC870025 and National Natural Science Fund 70572079.

References

- [1] E. Agirre and G. Rigau, Word sense disambiguation using conceptual density, in Proceedings of 16th International Conference on Computational Linguistics. Copenhagen, Denmark. Association for Computational Linguistics, 1996, pp. 16-22.
- [2] E. Agirre and G. Rigau, A proposal for word sense disambiguation using conceptual distance, in Proceedings of International Conference on Recent Advances in Natural Language Processing, Tzigov Chark, Bulgaria, 1995, pp. 258-264.
- [3] A. Amiri, Customer-oriented catalog segmentation: Effective solution approaches, Decision Support Systems, vol. 42, no. 3, pp. 1860-1871, 2006.
- [4] D. Beckett and T. Berners-Lee. (2008, January) Turtle - terse RDF triple language. W3C Team Submission. [Online]. Available: <http://www.w3.org/TeamSubmission/turtle/>.
- [5] J. de Bruijn, D. Fensel, and M. Kerrigan, Modeling Semantic Web Services, Heidelberg: Springer-Verlag, 2008, pp. 30-52.
- [6] E. Casasola, ProFusion personal assistant: An agent for personalized Information filtering on the WWW, M. S. thesis, The University of Kansas, Kansas, KCK, U.S.A., 1998.
- [7] S. Doring and W. KieBling, Evaluation and optimization of the catalog search process of e-procurement platforms. Electronic Commerce Research and Applications, vol. 5, no.1, pp. 44-56, 2006.
- [8] I. Chen, J. Ho, and C. Yang, On hierarchical web catalog integration with conceptual relationships in thesaurus, in Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, 2006, pp. 635-636.
- [9] K. Masanobu, D. Kobayashi, D. Xiaoyong, and I. Naohiro, Evaluating word similarity in a semantic network, Informatics, 2000, vol. 24, no. 1, pp. 192-202.
- [10] D. Fensel, Y. Ding, B. Omelayenko, E. Schulten, G. Botquin, M. Brown, and A. Flet, Product data integration in B2B e-Commerce, IEEE Intelligent Systems, vol. 16, no. 4, pp. 54-59, 2001.
- [11] O. Corcho, A. Gómez-Pérez, Solving integration problems of e-Commerce standards and initiatives through ontological mappings, in Proceedings of the 17th International Joint Conference on Artificial Intelligence, Seattle, 2001.
- [12] R. Cyganiak, A relational algebra for SPARQL. Digital Media Systems Laboratory HP Laboratories Bristol. HPL-2005-170, September 28, 2005.
- [13] Z. Cui, D. Jones, and P. O'Brien, Semantic B2B Integration: Issues in Ontology-based Approaches, SIGMOD Record, vol. 31, no. 11, 2002.
- [14] C. Donglin, L. Liying, D. Feng, W. Rui, A Study on Semantic Query System and Key Technologies of Distributed E-Catalogues, in Proceedings of the 4th International Conference on Wireless Communications, Network and Mobile Computing, DaLian, China, 2008, pp. 1-5.
- [15] D. Fensel, Y. Ding, B. Omelayenko, E. Schulten, G. Botquin, M. Brown, and A. Flet, Product data integration in B2B e-Commerce, IEEE Intelligent Systems, vol. 16, no. 4, pp. 54-59, 2001.
- [16] S. Gauch, J. Chaffee, and A. Pretschner, Ontology-based personalized search and browsing, Web Intelligence and Agent Systems, vol. 1, no. 3-4, pp. 219-234, 2003.
- [17] S. Gauch, M. Speretta, A. Chandramouli, and A. Micarelli, The Adaptive Web: Methods and Strategies of Web Personalization, Berlin: Springer-Verlag, 2007, pp. 54-89.
- [18] N. Guihua, G. Xuan, C. Donglin, Build standard domain ontology based on relational database, in Proceedings of the International Conference of Chinese Logistics and Transportation Professional, Wuhan, China, 2008, pp. 531-536.
- [19] P. Haase and A. Hotho, Collaborative and Usage-driven Evolution of Personal Ontologies, in Proceedings 2nd European Semantic Web Conference, Heraklion, Crete, 2005, pp. 486-499.
- [20] M. Hepp, Products and services ontologies: A methodology for deriving owl ontologies from industrial classification standards, International Journal on Semantic Web & Information Systems, vol. 2, no. 11, pp. 72-99, 2006.

- [21] M. Hepp, J. Leukel, and V. Schmitz, A quantitative analysis of product classification standards: content, coverage, and maintenance of eCI@ss, UNSPSC, eOTD, and the RosettaNet Technical Dictionary, Knowledge and Information Systems, vol. 13, no. 1, pp. 77-112, 2007.
- [22] W. Kim, D. W. Choi, and S. Park, Product information meta-search framework for electronic commerce through ontology mapping, Lecture Notes in Computer Science, The Semantic Web: Research and Applications, vol. 3532, Tucson, AZ: Springer, 2005, pp. 408-422.
- [23] D. Kim, S.-G. Lee, J. Chun, S. Park, and J. Oh, Catalog management in e-Commerce systems, in Proceedings of Computer Science and Technology, Cancun, 2003.
- [24] D. Kim, J. Kim, and S. Lee, Catalog integration for electronic commerce through category-hierarchy merging technique, in Proceedings of the 12th International Workshop on Research Issues on Data Engineering, San Jose, USA. IEEE Society, 2002, pp. 28-33.
- [25] R. Krovetz and W. B. Croft, Lexical ambiguity and information retrieval, ACM Transactions on Information Systems, vol. 10, no. 2, pp. 115-141, 1992.
- [26] L. Kwon and C. O. Kim, Recommendation of e-commerce sites by matching category-based buyer query and product e-Catalogs, Computers in Industry, vol. 59, no. 4, pp. 380-394, 2008.
- [27] J. Lee and T. Lee, Massive catalog index based search for e-Catalog matching, in Proceedings of the 9th IEEE International Conference on e-Commerce Technology. Tokyo. IEEE Computer Society, 2007, pp. 341-348
- [28] H. Lee, J. Shim, S. Lee, and S. Lee, Modeling considerations for product ontology, in Lecture Notes in Computer Science, Advances in Conceptual Modeling: Theory and Practice, vol. 4231, Tucson, AZ: Springer, 2006, pp. 291-300.
- [29] J. Leukel, V. Schmitz, and F. Dorloff, A modeling approach for product classification systems, in Proceedings of 13th International Conference on the Database and Expert Systems Applications. Aix-en-Provence, 2002, pp. 868-874.
- [30] H. Li, XML and industrial standards for electronic commerce, Knowledge and Information Systems, vol. 2, no. 4, pp. 487-497, 2000.
- [31] S. Liao, C. Chen, C. Hsieh, and S. Hsiao, Mining information users' knowledge for one-to-one marketing on information appliance, Expert Systems with Applications, vol. 36, no. 3, pp. 4967-4979, 2009.
- [32] L. Lim and M. Wang, Managing e-Commerce catalogs in a DBMS with native XML support, in Proceedings of the IEEE International Conference on e-Business Engineering, Beijing, 2005, pp. 564-571.
- [33] C. Lin and C. Hong, Using customer knowledge in designing electronic catalog, Expert Systems with Applications, vol. 34, no. 1, pp. 119-127, 2008.
- [34] D. Liu, Y. Lin, and C. Chen, Deployment of personalized e-Catalogues: An agent-based framework integrated with XML metadata and user models, Journal of Network and Computer Applications, vol. 24, no. 3, pp. 201-228, 2001.
- [35] K. Masanobu, D. Kobayashi, D. Xiaoyong, and I. Naohiro, Evaluating word similarity in a semantic network, Informatics, 2000, vol. 24, no. 1, pp. 192-202.
- [36] G. Nie, X. Zuo, and D. Chen, A modified concept similarity calculation methods in ontology mapping, Computer Application, vol. 28, no. 6, pp. 1563-1565, 2008.
- [37] L. Obrst, R. E. Wray, H. Liu, Ontological engineering for B2B e-Commerce, in Proceedings of the International Conference on Formal Ontology in Information Systems, Ogunquit, 2001, pp. 117-126.
- [38] H. Paik and B. Benatallah, Personalised organisation of dynamic e-Catalogs, in Web Services, e-Business, and the Semantic Web (C. Bussler, R. Hull, S. McIlraith, M. E. Orłowska, B. Pernici and J. Yang, Eds.). Heidelberg, Berlin: Springer Verlag, 2002, pp. 139-152.
- [39] E. Prud'hommeaux and A. Seaborne. (2005, July) SPARQL Query Language for RDF. W3C Working Draft. [Online]. Available: <http://www.w3.org/TR/2005/WD-rdf-sparql-query-20050721/>.
- [40] R. Rada, H. Mili, E. Bicknell, and M. Blettner, Development and application of a metric on semantic nets, IEEE Transaction on System, Man and Cybernetics, vol. 19, no. 1, pp. 17-30, 1989.
- [41] H. Sun-Young and K. Eun-Gyung, A study on the improvement of query processing performance of OWL data based on Jena, in Proceedings of the International Conference on Convergence and Hybrid Information Technology, Daejeon, 2008, pp. 678-681.
- [42] A. Tversky, Feature of similarity, Psychological Review, vol. 84, no. 4, pp. 327-352, 1977.
- [43] L. Yu and L. Sun, Products information standardization technology research in electronic commerce, Chinese manufacturing's informationization, vol. 33, no. 2, pp. 106-108, 2004.
- [44] Y. Yu and J. Yan, Semantic mapping in product design and catalog integration, in Proceedings IFIP International Conference on Network and Parallel Computing, 2008, pp. 493-498.
- [45] J. Zhaxue, G. Huang, F. Tang, and Y. Ye, Ontology based e-Catalog match for integration of GDSN and EPC global Network, in Proceedings of the IEEE International Conference on e-Business Engineering, Beijing, China. IEEE Computer Society, 2005, pp. 212-215.