Capturing Semantics from Search Phrases: Incremental User Personification and Ontology-Driven Query Transformation

Vadim Ermolayev, Natalya Keberle, Sergey Plaksin, Vladimir Vladimirov

Dept. of Mathematical Modeling and IT, Zaporozhye State Univ.,
Zhukovskogo 66, 69063, Zaporozhye, Ukraine
{eva, kenga, psl, vvlad}@zsu.zp.ua

Abstract: Reported is the methodology of the semantic transformation of an initial user's search query in the form of key words or key phrases to the resulting query composed of the relevant concepts of the domain ontology. Transformation methodology is based on incremental user profiling. The mapping of a user's keywords to the concepts of the domain ontology is built according to the presented transformation rules. These rules are based on the usage of the rich set of the semantic relationships comprising subsumption, synonymy, instantiation and meronymy provided as the DAML+OIL ontology. ACM research papers domain is chosen for the methodology evaluation. Transformation algorithm is implemented in the research prototype as the combined capability of the query transformation agent and the ontology agent of the intelligent multi-agent information retrieval mediator[1].

1. Introduction

The volumes and the diversity of the resources provided by the World Wide Web are already immense today and continue to grow rapidly. The audience retrieving data from the Web is spreading extensively as well. The Internet initially designed as the tool for merely research purposes is now becoming the World-wide information providing infrastructure. The variety of its applications ranges from information retrieval for scientific purposes through e-business and corporate knowledge management to entertainment. In these settings the traditional, keyword based, means for information retrieval provided by existing search engines become increasingly ineffective. The reasons are: information overload and the mismatches between the personal user's understanding of his/her keyword(s) and the semantics of syntactically relevant search results.

Ontologies are known (e.g., [Gr93]) as explicit shared formal specifications of real world conceptualizations. They are used, for instance, as a kind of a bridge to drive the matchmaking between a user's personal conception of a percept and the semantic annotations of the resources under retrieval. Given the appropriate and the widely

[1] The mediator is under development in frame of the RACING project (http://www.zsu.zp.ua/racing/) funded by Ukrainian Ministry of Education and Science, Grant No 0102У005339.

recognized ontology is available for the purpose, the question is how to use it most efficiently and effectively. The sub-problems for this mediation problem are:

- How to capture user's personal conception from his input, being the list of arbitrary keywords or phrases? How to map this captured meaning to the ontology concepts?
- How to ensure that the annotations of the resources within the domain are coherent with the mediator domain ontology?

The paper does not address the solution of the problem of resource annotation and of the mapping of these annotations to the mediator ontology. This is why the domain of ACM publications was chosen for the methodology evaluation. The annotations coherent to ACM Classification Ontology (ACMTopic)[2] are available from the abstracts of the papers published by ACM (e.g., the full texts available on the Internet, ACM Digital Library paper abstracts[3]).

A straightforward solution of the first sub-problem is to pick-up the concepts of a recognized domain ontology to compose the search phrase. The pitfall here is that a user, generally, may not be familiar with the necessary ontology. Moreover, in case a huge ontology (like, e.g., a common sense linguistic ontology Sensus [KL94] with about 70 000 concepts) is used for the purpose it may require quite a considerable effort just to get a general impression of it. Another way is to capture a user's conceptualization from his/her key phrases and to map it to the ontology concepts. The challenge here is – how exactly to capture the meaning from a key word? A key word is treated only as a syntactical atom by search engines – only lexical similarity is measured and exploited at most.

Normally a human when communicating to another human formulates his/her queries in natural language sentences and expects the recipient to pose qualifying questions on what is meant in the query. It is also expected that the captured semantics is incrementally preserved and effectively used for future conversations. The paper presents the similar approach to semi-automatic incremental capturing of the semantics of key phrases a user submits to a search engine. A user submits a search phrase which is than transformed by mapping the key words or the key phrases from this query one by one to the concepts of the domain ontology – enhanced ACM Topic ontology for the test case. These mappings are incrementally collected within the user profile. The user profile is further on used as the reference ontology to find personalized mappings for new queries.

The reminder of the paper is structured as follows. Section 2 surveys the related work. Section 3 describes user profiles. Section 4 reports on the semantic relationships used to describe key word – ontology concept mappings. Section 5 provides the query transformation rules based on the semantics of the relationships between keywords and concepts. Section 6 presents the query transformation algorithm. Section 7 reports on the prototype implementation. Section 8 provides the concluding remarks and the prospects for the future work.

---

[2] ACMTopic DAML+OIL ontology is available at http://daml.umbc.edu/ontologies/classification.daml.
[3] ACM requires that each paper provides the list of Index Terms within the abstract. Index terms are that of ACMTopic taxonomy.

2. Related Work

As it was outlined before the challenge of the semantic mediation of information retrieval brings up several problems: Where to get the proper domain ontology? How to provide coherent resource descriptions? How to map a search query to the domain ontology concepts? The following approaches are widely used attempting to provide the solutions:

1) Elaboration of semantically rich resource annotations formalized by means of description languages/ontologies. Providing XML(RDF)-based query languages for retrieving data from the annotated resources.
2) Categorization of the resources for (semi-)automatic extraction of the semantics and further (semi-)automatic creation of the domain taxonomies.
3) User profiling and personification providing the references of the user preferred semantics of his/her search phrases.

Unfortunately no one of them proves to become a Silver Bullet, at least so far. Moreover, these approaches are often combined to achieve better performance and recall/precision of information retrieval.

The efforts of W3C[4] RDF and Semantic Web communities result in the development of RDF-based languages family: RDF-S[5], DAML+OIL[6], OWL[7]. The examples of the query languages for retrieving information from the resources annotated by RDF-based languages are XQL [RLS98], XML-QL [De98], LOREL [Ab97]. Characteristic examples of the prototype systems which exploit the routine of semantically rich resource annotation for further retrieval are SHOE [HH00] and OntoSeek [GMV99]. SHOE provides the description language for the annotation of HTML resources with semantic information. SHOE search tool allows a user to specify a context for the query by manual ontology browsing. The context is further on used to help the user to build a query by example. OntoSeek is the system for content-based search in product catalogs and yellow pages based on lexical conceptual graphs (LCG). OntoSeek requires that the resources should be annotated by their LCGs. A user is prompted to submit his/her query as LCG as well. The search algorithm is based on the matchmaking of the query conceptual graph (CG) to the part of the ontology CG. OntoSeek exploits the third-party linguistic ontology (Sensus). An OntoSeek user is free to choose the arbitrary concepts for his annotation. The constraint however is that this freedom is bounded to the existing concepts of the ontology, though Sensus is rather a big thing.

Categorization systems are primarily designed for browsing, filtering and search in diverse information sources. (Semi-)automatic categorization (creating classifiers, categories, taxonomies) stands actually for mining classification semantics from huge collections of semi-structured documents. Known are several systems providing automated categorization. Category taxonomies are constructed before (Readware[8],

---

[4] World Wide Web Consotium: http://www.w3c.org/
[5] http://www.w3.org/TR/rdf-schema/ – last checked on March 31, 2003
[6] http://www.w3.org/TR/daml+oil-reference/ – last checked on March 31, 2003
[7] http://www.w3.org/TR/owl-ref/ – last checked on March 31, 2003
[8] http://www.readware.com – last checked on March 31, 2003

TEXIS[9]), in the process of (Readware, Sun Microsystems Conceptual Indexing [Wo00]) or after the registration (or the upload) of the documents to the system (LexiQuest[10]).

User personalization is widely used in Web portals and recommender[11] systems. Most personalization strategies are based on the use of some kind of a user profile. Two questions arise: 1-st – How to build a profile? and 2-nd – Which semantic data to capture within a profile?

Two categories of the approaches to profile creation should be mentioned: manual (often based on a sort of semantic annotation, like in OntoSeek, QuickStep [MRS01], Foxtrot [MRS02]) and automated (based on "watching over the user's shoulder" technique capturing user's preferences from his browsing behavior). For example, personal web-based agents like Letizia [Li95], Syskill & Webert [PMB96] and Personal Webwatcher [Ml96] track the users' browsing and provide inputs for the user profile formulation. Profiles are constructed from positive and negative examples of interest, obtained from explicit feedback or heuristics analyzing users' browsing behavior.

Most commonly, the information captured within a profile contains: the list of weighted key words, some structured information (e.g., bookmark structure). A user profile is essentially a reference ontology in which each concept is supplied with its weight indicating the perceived user interest in this concept [GCP03]. The weight is usually denoted as the distance between different senses of the concept.

3. User Profiles – Incremental Personification

In frame of the reported work user profiles are the part of the RACING mediator [Er02] knowledge base. A user profile is the reference ontology which allows collecting and keeping the knowledge on what a key word or a key phrase means for the particular user. This personal knowledge is represented as a semantic relationship between the given key word/phrase and the concept of the mediator ontology as follows:

$$(K_i <sr> C_i\ Sim), \quad (1)$$

where:

K_i – is the key word/phrase;

C_i – is the ontology concept, which the user considers to be relevant to given K_i;

<*sr*> – is the semantic relationship (Section 4) which holds between K_i and C_i

Each key word/phrase K_i may have *several* related concepts C_i resulting in several records in the user profile as far as the user may have multiply meanings of the key word for different settings. The presence of several C_i-s reflects, e.g., a user's interest in several subject domains.

It is hardly possible to precisely describe the exact meaning of a key word by the means of this very simple structure and the restricted set of semantic relationships. The function

[9] http://www.thunderstone.com – last checked on March 31, 2003
[10] http://www.lexiquest.com – last checked on March 31, 2003
[11] The Proc. of the 2001 ACM SIGIR Workshop on Recommender Systems:
http://cs.oregonstate.edu/~herlock/rsw2001/workshop_notes.html – last checked on March 31, 2003

of the last element of the profile record ($Sim \in [0,1]$) is to rate the accuracy of the given semantic description. *Sim* is the heuristic measure assigned by the user. It reflects his/her personal opinion on how close is his/her meaning of the key word or the key phrase to the description provided by the profile record. *Sim*=0 stands for the complete mismatch, whereas *Sim*=1 stands for the complete coherence. Let, for example, the personal user's meaning of the key word '**agent**' be '*the person who stands at his/her door and wants to sell something*'. Then, if the profile record says ('**agent**' is-a '**intelligent software system**'), the similarity value *Sim* might be close to 0. Otherwise, if the user thinks that an '**agent**' is '*a program that sells goods via the Internet*', the similarity value *Sim* should be closer to 1. Similarity values in user profiles allow to flexibly alter the precision of the search by setting the threshold factor. Only the concepts from the user profile records which similarity value is greater than the threshold factor will be used in the search phrase transformation (Section 6).

User profiles receive new records when users pose queries containing the key words or the key phrases which are yet unknown to the mediator. It of course takes some effort from the user (Step 3 of the transformation algorithm, Section 6) to introduce the new key word to the system. However, the user can't do better than to spend this effort on this very key word at this very time. Otherwise, the precision of information retrieval will be much worse (traditional key word based search) and the effort spent for filtering out the irrelevant responses may appear to be more substantial than the effort spent for the new key word introduction.

4. Semantic Relationships

Most classifications, used in information retrieval systems are taxonomies (e.g., Thunderstone TEXIS, Sun Conceptual Indexing [Wo00]) and semantic networks (LexiQuest, Readware). Some of them make use of linguistic ontologies like WordNet [Mi95] or Sensus. The advantage of some of the latter ones (e.g., OntoSeek, Readware) is that the exploited semantic relationships between the ontology concepts are richer than "subtype/supertype" taxonomy relationships.

In the reported work semantic relationships are used to specify semantic dependencies between the different ontology concepts and between the key words and the ontology concepts in the user profiles. The set of the "legal" semantic relationships is restricted to subsumption, meronymy ("part-whole"), instantiation and synonymy (refer to [St93] for the example of the complete classification). Subsumption, synonymy and instantiation relationships are the basic constructs of DAML+OIL, whereas meronymy relationships are specified in RACING-meronymy ontology[12] as the extension of DAML+OIL.

The semantics of the mentioned relationships is as follows:

- ***is-a*(*X, Y*)** is the binary predicate over the concepts *X, Y*, which stands for a subsumption relationship between concepts *X* and *Y* (e.g., *X* – a car and *Y* – a vehicle)

[12] http://eva.zsu.zp.ua/eva_personal/ontologies/racing-meronymy.daml

- ***μ* (*X, Y*)** is the binary predicate over the concepts *X, Y*, which stands for a meronymy relationship between concepts *X* and *Y* (e.g., *X* – flour and *Y*– a pie)

- ***instance-of* (*x, X*)** is the binary predicate over the concept *X* and one of its instances *x* (e.g., *X* – a city and *x* – Kharkiv)

- ***synonym*(*X,Y*)** is the binary predicate over the concepts *X* and *Y*, which stands for a synonymy between these concepts (e.g., *X* – a motor and *Y* – an engine)

Following [St93], a subsumption relationship is understood as "A is-a B" or "A is-a-kind-of B", where A stands for the subtype and B is the supertype. Meronymy relationships are the relationships of inclusion type, which differ from class inclusion in the sense that they occur between something (a whole) and its parts. Generally, known are seven kinds of meronymy relationships [St93]: component-object ("component-of"), member-collection ("member-of"), portion-mass ("part-of"), phase-activity ("part-of"), feature-event ("part-of"), place-area ("is-in"), stuff-object ("made-of"). Three of them have the same verb phrase. It is thus necessary to analyze the whole relationship expression to clarify the difference (refer to [St93], [St88] for more details). Luckily, these distinctions do not affect the transformation rules.

5. The Rules for User Query Transformation

It is known (e.g., from [Ch00]) that the semantic mappings may be translations or transformations. A translation is an ideal solution because it preserves the semantics of the concepts. However, as it was mentioned in Section 3, it is hard to preserve the semantics of the mapping of an arbitrary key word or a key phrase to the ontology concept(s) because of the lack of the expressivity of the descriptive tools under disposal. The mappings exploited in frame of the reported research are therefore the transformations. These transformations are constructed in a way to ensure that the *recall* (e.g., [GCP03]) of the resulting query (RQ) in terms of the ontology concepts is at least the same than the *recall* of the initial user's query (IQ) in terms of key words or key phrases. This implies the following ***IQ preservation principle***: the resulting search phrase should have the same or the broader meaning if compared to the initial search phrase.

So far, for the simplicity reasons, it is assumed that an IQ is submitted by a user in the form of the blank separated list of the arbitrary key words and/or quoted key phrases in English ($\{K_1 ... K_n\}$) accompanied with one of the two possible logical connectors AND or OR. This implies that a query may be either disjunctive (K_1 OR K_2 OR ...OR K_n) or conjunctive (K_1 AND K_2 AND ... AND K_n).

An IQ→RQ transformation is performed with the help of the set of the mapping rules for the aggregation of semantic relationships. These rules are defined as follows. Let *X, Y, Z* be the arbitrary ontology concepts. In case they are related only with "*is-a*" relationship, the following rules will hold:

$$\begin{array}{lll} is\text{-}a(X, Y) \text{ AND } is\text{-}a(Y, Z) \Rightarrow & is\text{-}a(X, Z) & \text{(transitivity)} \\ is\text{-}a(X, Y) \text{ AND } is\text{-}a(X, Z) \Rightarrow & is\text{-}a(X, (Y \text{ AND } Z)) & \text{(additivity)} \\ is\text{-}a(X, Y) \text{ AND } is\text{-}a(Y, X) \Rightarrow & X \equiv Y & \text{(antisymmetry)} \end{array} \quad (2)$$

In case X, Y, Z are related with meronymy relationship the following rules will hold (see [Sm98]):

$$\begin{array}{lll} \mu(X, Y) \text{ AND } \mu(Y, Z) \Rightarrow \mu(X, Z) & & \text{(transitivity)} \\ \mu(X, Y) \text{ AND } \mu(Y, X) \Rightarrow X \equiv Y & & \text{(antisymmetry)} \end{array} \quad (3)$$

In case X, Y, Z are related with both subsumption and meronymy relationships the following meronymy inheritance rules will hold:

$$\begin{array}{lll} \textit{is-a}(X, Y) \text{ AND } \mu(Y, Z) \Rightarrow \mu(X, Z) & & \\ \textit{is-a}(X, Y) \text{ AND } \mu(Z, Y) \Rightarrow \mu(Z, X) & & \text{([Ev88])} \end{array} \quad (4)$$

In case the semantic relationship(s) is(are) set up between a key word(s) and the relevant ontology concept(s) in the user profile the IQ→RQ transformation rules hold (Table 1). The rationale for these rules is similar to that of (2)-(4).

IQ→RQ transformation rules. Table 1.

Key word(s)	**Semantic relationship between key word(s) and concepts in a user profile**	**Concept(s)**
K_i	*is-a* (K_i, Y)	Y
K_i	*is-a* (K_i, Y) AND *is-a* (K_i, Z)	Y AND Z
K_i	μ (K_i, Y)	Y
K_i	μ (K_i, Y) AND μ (K_i, Z)	Y OR Z
K_i	μ (K_i, Y) AND *is-a* (K_i, Z)	Y AND Z
K_i, K_j, $i \neq j$	μ (K_i, Y) AND μ (K_j, Y)	Y

Application of the transformation rules to an IQ results in the intermediate query which comprises only the ontology concepts. It may happen that the semantic relationships hold between the involved ontology concepts. The rules for semantic aggregation of these ontology concepts should therefore be introduced. Two groups of aggregation rules for conjunctive and disjunctive queries are given in Table 2.

IQ→RQ transformation routine is therefore as follows:

- Transform IQ key words/phrases one by one with te help of the rules of Table 1
- Perform intermediate query concepts aggregation according to the rules of Table 2

Aggregation rules for two types of queries. Table 2.

Logical sentence	**Concepts involved**	**Resulting logical sentence**
Aggregation rules for conjunctive queries		
A AND A_1	A, A_1: *is-a* (A_1, A)	A_1
A_1 AND A_2	A_1, A_2	A_1 AND A_2
A_1 AND a_2	A_1, *instance-of* (a_2, A_2)	A_1 AND A_2=a_2
A_1 AND (A_2 OR A_3)	A_1, A_2, A_3	A_1 AND (A_2 OR A_3)
A_1 AND A_2	A_1, A_2: *synonym*(A_1, A_2)	A_1 OR A_2
A_1 AND A_2	A_1, A_2: μ(A_1, A_2)	A_1 AND A_2
Aggregation rules for disjunctive queries		
A OR A_1	A, A_1: *is-a*(A_1, A)	A
A_1 OR A_2	A_1, A_2	A_1 OR A_2
A_1 OR a_2	A_1, *instance-of* (a_2 ,A_2)	A_1 OR A_2=a_2
A_1 OR (A_2 AND A_3)	A_1, A_2, A_3	A_1 OR (A_2 AND A_3)

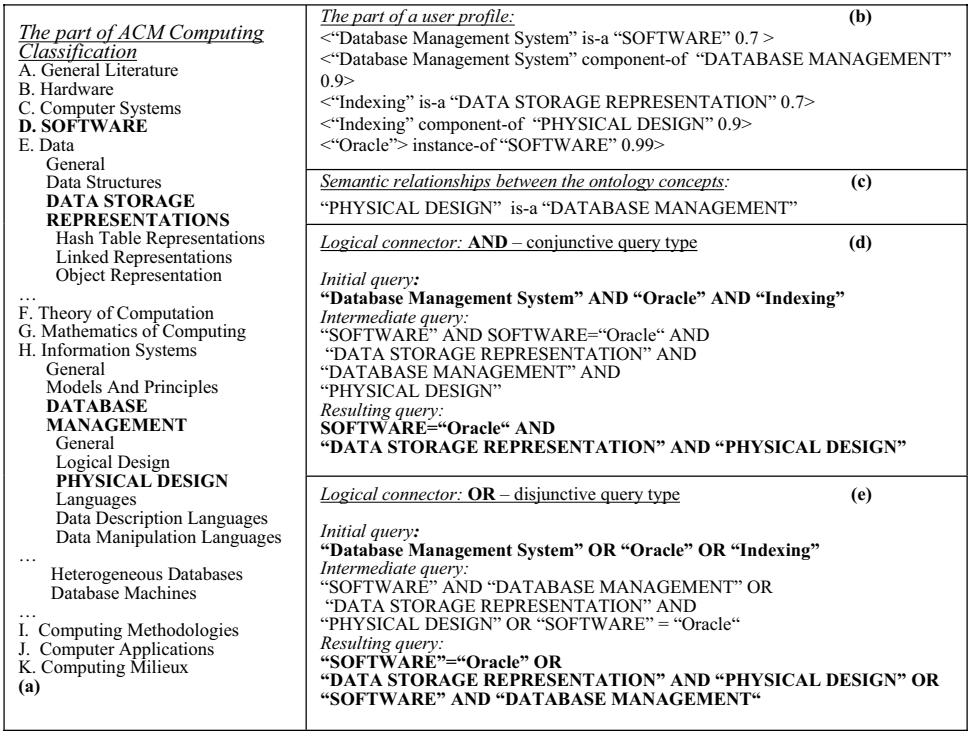

Figure 1. Example of IQ→RQ transformation in ACM research papers domain:
(a) the fragment of ACM Topic taxonomy; (b) the part of a user profile;
(c) the relationship between ontology concepts;
(d) RQ – conjunctive type; (e) RQ – disjunctive type.

The transformation routine is illustrated by the example from ACM research papers domain on Fig. 1.

6. The Algorithm for Initial User Query Transformation (IQ→RQ)

$K_1 ... K_n$ are further on referred to as IQ atoms. The task of the Transformation Algorithm (TA) is:

- To build the Query Plan (QP) as the set of mappings of K_i to the concepts of the ontology C_j:

$$(K_i <sr> C_j \; Sim \; URI) \tag{5}$$

- To ask the user to approve the proposed QP by checking appropriate K_i mappings of QP
- To apply Transformation Rules (TR) to the Concepts of QP and thus to finally compose the RQ

K_i, <*sr*>, C_j and *Sim* in (5) have the same meaning as in (1). *URI* is the Universal Resource Identifier of C_j of the mediator domain ontology.

TA is presented on Fig. 2 and comprises the following steps for each K_i of the IQ.

Step 1: User profile match

The goal of this step is to find out if the semantic mapping(s) for K_i have been created before. The user profile is inquired for the records containing K_i as the key word. All matching records are added to the QP.

Step 2: Direct Ontology Match

The search for direct match with ontology elements is performed in case TA fails to find user profile matches for K_i. The ontology is inquired for the concepts, the synonyms of the concepts, the instances of the concepts which are syntactically equivalent to K_i. A record for QP is created for each match. <*sr*> in this record belongs to the following subset of semantic relationships:

- *"equivalent-of"* in case a matching concept name has been found
- *"synonym-of"* in case a matching synonym has been found
- *"instance-of"* in case a matching concept instance has been found

Similarity value *Sim* is set to 1 as far as the direct match has been found. C_j is set to the concept name.

Step 3: Manual Ontology Match

In case both ST1 and ST2 have failed to find relevant matches the only chance left is to perform manual ontology browsing and to ask the user to pick up the matching ontology concepts to the QP according to his personal understanding of K_i meaning. This activity will further on replenish the user profile and thus provide more knowledge on the user personification to the system. While picking up matching ontology concepts C_i the user also chooses the appropriate semantic relationship to hold between K_i and C_i. The semantic relationship is chosen from the available set provided by DALM+OIL and RACING Meronymy Ontology. The user also assigns the similarity value $Sim \in [0,1]$ according to his personal opinion on the differences between the meaning of K_i he/she has in mind and the assigned semantic relationship of K_i to the chosen ontology concept C_i.

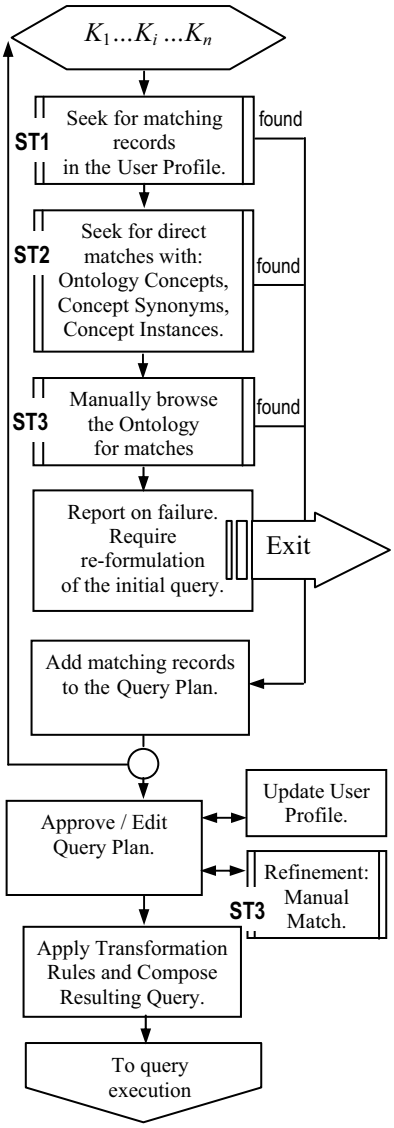

Figure 2. IQ→RQ Transformation algorithm

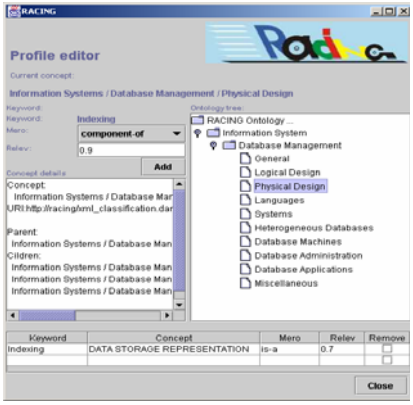

Figure 3. RACING User Profile Editor the interface for ST3 of IQ→RQ algorithm.

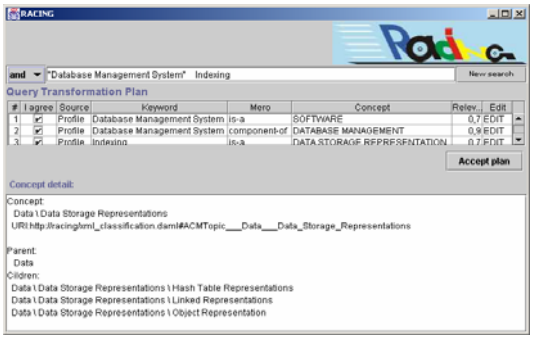

Figure 4. The interface for QP refinement and approval. Presented is the QP for the example on Fig. 1(d).

After steps 1 to 3 are performed and still no match has been found TA fails and requires initial query re-formulation. The failure means that:

- Either the ontology is not complete and does not contain the concepts reflecting the user's field of interest
- Or the user was lazy enough and did not properly perform ST3, preferring to re-formulate the query and to rely on the personification knowledge already recorded to his/her profile

The fact that the failures are not rare might signal on the necessity to refine the ontology (not discussed in the paper). Rare failures are quite a normal situation and arise for example if the user has finally decided that the query formulation was not correct.

QP approval/refinement phase takes place in case all IQ atoms have found their candidate mappings within the QP. The purpose of the phase is to ask the user to check the selection of the most relevant mappings from the QP and, possibly, to refine some of the mappings by applying ST3. The selection of the QP mappings is further on used for the final query transformation as described in Section 5.

7. Prototype Implementation

Agent-based software prototype has been implemented to evaluate the transformation methodology. The prototype multi-agent system comprises two FIPA-compliant[13] agents: User Query Transformation agent (QTA) and RACING mediator Ontology Agent (OA). QTA is the agent which has direct contact to the user and performs the query transformation. The user interfaces of QTA are shown on Fig. 3 and 4. One of the major functions of the OA is mediator knowledge base management. OA supplies QTA with the contents of the user profile, RACING Meronymy ontology and the required portions of the domain mediator ontology (ACM Topic in our current implementation).

[13] Implementation platform is FIPA-OS: http://fipa-os.sourceforge.net/

8. Conclusions and Future Work

Reported results are the partial implementation of the ongoing RACING project. The main goal of the project is to design and deploy the agent-based rational intelligent mediator for information retrieval. Presented IQ→RQ transformation algorithm is the capability of QTA of the RACING mediator. IQ→RQ transformation process is actually based on the contents of a user profile, domain ontology provided by the OA in responce to QTA requests. This capability is essential for the further implementation of query processing processes.

The reported approach combines ontology-driven incremental user personification and the mapping of the IQ atoms to the concepts of the domain mediator ontology. The mapping of a user's keywords to the concepts of the domain ontology is built according to the presented transformation rules. These rules are based on the usage of the rich set of the semantic relationships comprising subsumption, synonymy, instantiation and meronymy which extends standard DAML+OIL. Though the current implementation uses the specific (ACM Topic) taxonomy as the domain ontology, it is evident that the proposed methodology is ontology invariant. Any other widely recognized ontology[14] may be incorporated into the mediator knowledge base due to the import facility of the OA. Moreover, the incremental profiling technique may provide valuable feedbacks for the enrichment, revision or harmonization of the domain ontology. The refined ontology may be than exported by the OA and made publicly available.

One of the important planned directions of the future work is the full scale evaluation of the transformation methodology by means of the series of experiments measuring recall and precision figures as the dependencies of the satiation of user profiles for different users. Another direction is the study of the influence of the similarity threshold factors in user profiles on the precision of the resulting queries.

References

[Ab97] Abiteboul, S. et al.: The lorel query language for semistructured data. In J. of Digital Libraries, Vol. 1:1, 1997, pp. 68-88.

[Ch00] Chalupsky, H. Ontomorph: A translation system for symbolic knowledge. In (Cohn, A.; Giunchiglia, F.; Selman, B. Eds): Principles of Knowledge Representation and Reasoning. Proc. 7th Int. Conf. (KR'2000), San Francisco, CA, pp. 471-482.

[De98] Deutsch, A. et al.: XML-QL: A query language for XML. http://www.w3.org/TR/NOTE-xml-ql.

[Er02] Ermolayev, V. et al.: Towards Agent-Based Rational Service Composition – RACING Approach. Tech. Report. Dept. of Mathematical Modeling and IT, Zaporozhye State Univ., Jun., 2002, 32 pp.

[Ev88] Evens, M.: Introduction. In (Evens, M. Ed.): Relational Models of the Lexicon: Representing Knowledge in Semantic Networks. Cambridge University Press, NY, 1988, pp.1-37.

[14] Coded in DAML+OIL

[GCP03] Gauch, S.; Chaffee, J.; Pretschner, A.: Ontology-Based User Profiles for Search and Browsing. To appear in J. User Modeling and User-Adapted Interaction: The Journal of Personalization Research, Special Issue on User Modeling for Web and Hypermedia Information Retrieval, 2003.

[Gr93] Gruber, T. R.: A Translation Approach to Portable Ontology Specifications, Knowledge Acquisition, Vol. 5, 1993, pp. 199-220.

[GMV99] Guarino, N.; Masolo, C.; Vetere, G.: OntoSeek: Content-Based Access to the Web. IEEE Intelligent Systems, Vol. 14:3, May 1999, pp. 70-80.

[HH00] Heflin, J.; Hendler, J.: Searching the Web with SHOE. In Artificial Intelligence for Web Search. Papers from the AAAI Workshop. WS-00-01, AAAI Press, 2000, pp. 35-40.

[KL94] Knight, K.; Luk, S.: Building a Large Knowledge Base for Machine Translation. In: Proc. AAAI-94, AAAI Press, Menlo-Park, Caliofornia, 1994, pp. 773-778.

[Li95] Lieberman, H.: Letizia: An Agent That Assists Web Browsing. In: Proc. 1995 Int. Joint Conf. on Artificial Intelligence, Montreal, Canada, August 1995.

[Mi95] Miller, G.A.: WORDNET: A Lexical Database for English. In: Comm. of the ACM Vol. 2:11, 1995, pp. 39-41.

[Ml96] Mladenic, D.: Personal WebWatcher: Implementation and Design, Technical Report IJS-DP-7472, Department of Intelligent Systems, J.Stefan Institute, Slovenia, 1998.

[MRS01] Middleton, S.E.; De Roure, D.C.; Shadbolt, N.R.: Capturing Knowledge of User Preferences: ontologies in recommender systems. In: Proc. 1st Int. Conf. on Knowledge Capture (K-CAP 2001), Victoria, BC, Canada, Oct. 2001.

[MRS02] Middleton, S.E.; De Roure, D.C.; Shadbolt, N.R.: Foxtrot Recommender System: User profiling, Ontologies and the World Wide Web, Poster, Proc. 11th Int. World Wide Web Conference (WWW'2002), Hawaii, USA, May 2002.

[PMB96] Pazzani, M.; Muramatsu J.; Billsus, D.: Syskill & Webert: Identifying interesting web sites. In: Proc. National Conf. on Artificial Intelligence, Portland, Oregon, 1996.

[RLS98] Robie, J.; Lapp, J.; Schach, D.: XML Query Language (XQL), In: Proc. the Query Language Workshop (QL) of WWW Conf., Dec. 1998.

[Sm98] Smith, B.: Basic Concepts of Formal Ontology. In (Guarino, N. Ed.): Proc. of the 1st Int. Conf. on Formal Ontologies in Information Systems (FOIS'98), Trento, Italy 1998, pp.19-28.

[St88] Storey, V. C.: View Creation: An expert system for database design. Ph.D. Dissertation, Faculty of Commerce and Business Administration, Univ. of British Columbia, Vancouver, Canada. Washington, DC: ICIT Press, 1998.

[St93] Storey, V. C.: Understanding semantic relationships. The Very Large Data Bases (VLDB) Journal, Vol. 2:4, 1993, pp. 455-488.

[Wo00] Woods, W. A. et al.: Linguistic Knowledge can Improve Information Retrieval. In (Treichel, J.; Holzer, M. Eds.): Sun Microsystems Laboratories. The First Ten Years: 1991-2001 Perspectives Series 2001-5, Oct. 2001.