Combining Semantic and Multimedia Query Routing Techniques for Unified Data Retrieval in a PDMS *

Claudio Gennaro[1], Federica Mandreoli[2,4], Riccardo Martoglia[2], Matteo Mordacchini[1], Wilma Penzo[3,4], and Simona Sassatelli[2]

[1] ISTI - CNR, Pisa, Italy
{firstname.lastname}@isti.cnr.it
[2] DII - University of Modena e Reggio Emilia, Italy
{firstname.lastname}@unimo.it
[3] DEIS - University of Bologna, Italy
{firstname.lastname}@unibo.it
[4] IEIIT - BO/CNR, Bologna, Italy

Abstract. The NeP4B project aims at the development of an advanced technological infrastructure for data sharing in a network of business partners. In this paper we leverage our distinct experiences on semantic and multimedia query routing, and propose an innovative mechanism for an effective and efficient unified data retrieval of both semantic and multimedia data in the context of the NeP4B project.

1 Introduction and Related Work

Information and communication technologies (ICTs) over the Web have become a strategic asset in the global economic context. The Web fosters the vision of an Internet-based global marketplace where automatic cooperation and competition are allowed and enhanced. This is the stimulating scenario of the ongoing Italian Council co-funded NeP4B (Networked Peers for Business) Project whose aim is to develop an advanced technological infrastructure for small and medium enterprises (SMEs) to allow them to search for partners, exchange data and negotiate without limitations and constraints.

According to the recent proposal of Peer Data Management Systems (PDMSs) [5], the project infrastructure is based on independent and interoperable semantic peers who behave as nodes of a virtual peer-to-peer (P2P) network for data and service sharing. In this context, a semantic peer can be a single SME, as well as a mediator representing groups of companies, and consists of a set of data sources (e.g. data repositories, catalogues) placed at the P2P network's disposal through an OWL ontology. Data sources include multimedia objects, such as the descriptions/presentations of the products/services extracted from the companies'

* This work has been partially supported by the Italian National Research Council in the context of the NeP4B Project and of the research initiative "Ricerca a tema libero" and by the IST FP7 European Project S-Cube (Grant Agreement no. 215483)

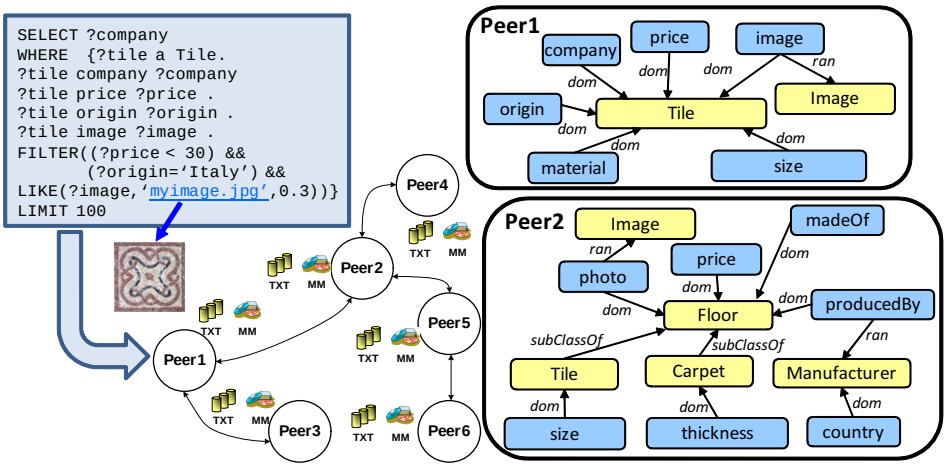


Fig. 1. Reference scenario

Web sites. This information is represented by means of appropriate multimedia attributes in the peers' ontologies (e.g. `image` in Peer1's ontology of Figure 1) that are exploited in the searching process by using a SPARQL-like language properly extended to support similarity predicates. As an example, let us consider the query in Figure 1 which asks Peer1 for "Companies that sell Italian tiles similar to the represented one and that cost less than 30 euros": the `FILTER` function `LIKE` is used to search for images whose similarity with the image provided as argument is greater than 0.3. Moreover, the query also specifies the number of expected results in the `LIMIT` clause.

Each peer is connected to its neighbors through semantic mappings which are exploited for query processing purposes: in order to query a peer, its own ontology is used for query formulation and semantic mappings are used to reformulate the query over its immediate neighbors, then over their immediate neighbors, and so on [5, 7]. For instance, in Figure 1 the concept `origin` translates into `country` when the query is forwarded to Peer2.

In such a distributed scenario, where query answers can come from any peer in the network which is connected through a semantic path of mappings [5], a key challenge is *query routing*, i.e. the capability of selecting a small subset of relevant peers to forward a query to. Flooding-based techniques are indeed not adequate for both efficiency and effectiveness reasons: not only they overload the network (forwarded messages and computational effort required to solve queries), but also overwhelm users with a large number of results, mostly irrelevant.

Query routing in P2P systems has attracted much research interest in the last few years, with the aim of effectively and efficiently querying both multimedia [1] and semantic data [9]. As far as we know, no proposal exists which operates on both these kinds of data in an integrated approach.

As part of the NeP4B project, we leverage our distinct experiences on semantic [6, 8] and multimedia [3, 4] query routing and propose to combine the approaches

we presented in past works in order to design an innovative mechanism for a unified data retrieval in such a context. Two main aspects characterize our scenario. The former one is due to the heterogeneity of the peers' ontologies which may lead to semantic approximations during query reformulation. In this context, we pursue *effectiveness* by selecting, for each query, the peers which are semantically best suited for answering it, i.e. whose answers best fit the query conditions. The latter aspect is related to the execution of multimedia predicates, which is inherently costly (they typically require the application of complex functions to evaluate the similarity of multimedia features). In this setting, we also pursue *efficiency* by limiting the forwarding of a query towards the network's zones where potentially matching instances are more likely to be found, while pruning the others. In this way, we give the user a higher chance of receiving first the answers which better satisfy the query conditions.

To this end, we introduce a query processing model which satisfies the interoperability demands highlighted in the NeP4B project. The proposed model does not compel to a fixed semantics but rather it is founded on a *fuzzy* settlement which proved to be sound for a formal characterization of the approximations originated in the NeP4B network for both semantic [6] and multimedia data [10]. The model leverages the query answering semantics (Sect. 2) to define a query routing approach which operates on both semantic and multimedia data in an integrated way (Sect. 3), and to show how routing strategies (Sect. 4) influence the order of the returned answers. This allows different query processing approaches to be implemented, based on the specific consortium needs and policies. The validity of the proposal is proved by the initial experiments we conducted on different settings (Sect. 5).

2 Query Answering Semantics

The main aim of this section is to propose a semantics for answering queries in the NeP4B network where two kinds of approximations may occur: the one given by the evaluation of multimedia predicates, and the other one due to the reformulation of queries along paths of mappings.

We denote with $\mathcal{P}$ the set of peers in the network. Each peer $p_i \in \mathcal{P}$ stores local data, modelled upon a local OWL ontology O_i. Peers are pairwise connected in a semantic network through semantic mappings between peers' ontologies. For our query routing purposes, we abstract from the specific format that semantic mappings may have. For this reason, we consider a simplified scenario where each peer ontology O_i is represented through a set of ontology classes $\{C_{i_1}, \ldots, C_{i_{m_i}}\}$[5] and semantic mappings are assumed to be directional, pairwise and one-to-one. The approach we propose can be straightforwardly applied to more complex mappings relying on query expressions as proposed, for instance, in [5]. Mappings are extended with scores quantifying the strength of the relationship between the involved concepts. Their fuzzy interpretation is given in the following.

[5] Note that in OWL properties are specified through classes.

Definition 1 (Semantic Mapping). *A* semantic mapping *from a source schema O_i to a target schema O_j, not necessarily distinct, is a fuzzy relation $M(O_i, O_j) \subseteq O_i \times O_j$ where each instance (C, C') has a membership grade denoted as $\mu(C, C') \in [0, 1]$. This fuzzy relation satisfies the following properties: 1) it is a 0-function, i.e., for each $C \in O_i$, it exists exactly one C' in O_j such that $\mu(C, C') \geq 0$; 2) it is reflexive, i.e., given $O_i = O_j$, for each $C \in O_i$ $\mu(C, C) = 1$.*

For instance, Peer1 of Fig. 1 maintains two mapping relations: the mappings towards Peer2 ($M(O_1, O_2)$) and Peer3 ($M(O_1, O_3)$). For instance, $M(O_1, O_2)$ associates Peer1's concept `origin` to Peer2's concept `country` with a score of 0.73, thus expressing that a semantic approximation is detected between the two concepts (for instance, the country might be only a part of an origin).

A query is posed on the ontology of the queried peer. Query conditions are expressed using predicates that can be combined in logical formulas through logical connectives, according to the syntax:

$$
\begin{aligned}
f &::= \langle triple\_pattern \rangle \langle filter\_pattern \rangle \\
\langle triple\_pattern \rangle &::= triple \mid \langle triple\_pattern \rangle \wedge \langle triple\_pattern \rangle \\
\langle filter\_pattern \rangle &::= \varphi \mid \langle filter\_pattern \rangle \wedge \langle filter\_pattern \rangle \mid \\
&\quad \langle filter\_pattern \rangle \vee \langle filter\_pattern \rangle \mid (\langle filter\_pattern \rangle)
\end{aligned}
$$

where *triple* is an RDF triple and a filter φ is a predicate where relational ($=$, $<$, $>$, $<=$, $>=$, $\neq$) and similarity ($\sim_t$) operators operate on RDF terms and values. In particular, note that $\sim_t$ refers to multimedia content and translates the `LIKE` operator where t is the specified similarity threshold.

Each peer receiving a query first retrieves the answers from its own local data then it reformulates the query towards its own neighborhood.

The evaluation of a given query formula f on a local data instance i is given by a score $s(f, i)$ in $[0, 1]$ which says how much i satisfies f. The value of $s(f, i)$ depends on the evaluation on i of the filters $\varphi_1, \ldots, \varphi_n$ that compose the filter-pattern of f, according to a scoring function $sfun_\varphi$, that is: $s(f(\varphi_1, \ldots, \varphi_n), i) = sfun_\varphi(s(\varphi_1, i), \ldots, s(\varphi_n, i))$. Note that filters are predicates of two types: relational and similarity predicates. A relational predicate is a predicate which evaluates to either 1 (true) or to 0 (false). The evaluation of a similarity predicate φ follows instead a non-Boolean semantics and returns a score $s(\varphi, i)$ in $[0, 1]$ which denotes the grade of approximation of the data instance i with respect to φ. It is set to 0 when the similarity of i w.r.t. the predicate value is smaller than the specified threshold t, and to the grade of approximation measured, otherwise. The scoring function $sfun_\varphi$ combines the use of a t-norm ($s_\wedge$) for scoring conjunctions of filter evaluations, and the use of a t-conorm ($s_\vee$) for scoring disjunctions. A t-norm (resp., t-conorm) is a binary function on the unit interval that satisfies the boundary condition (i.e. $s_\wedge(s, 1) = s$, and resp., $s_\vee(s, 0) = s$), as well as the monotonicity, the commutativity, and the associativity properties.[6] The use of t-norms and t-conorms generalizes the query evaluation model with respect to the use of specific

[6] Examples of t-norms are the *min* and the *algebraic product* operators, whereas examples of t-conorms are the *max* and the *algebraic sum* operators.

functions. Therefore, for a given peer p, the query answers retrieved from the evaluation of f on its own local data is given by $Ans(f,p) = \{(i, s(f,i)) \mid s(f,i) > 0\}$, i.e., it is the set of local data instances which satisfy f, possibly with a certain grade of approximation.

Due to the heterogeneity of schemas, any reformulation a peer p_i performs on a given query formula f towards one of its neighbors, say p_j, gives rise to a semantic approximation which depends on the strength of the relationship between each concept in f and the corresponding concept in O_j. Such an approximation is quantified by a scoring function $sfun_c$ which combines the p_j's mapping scores on f's concepts: $s(f,p_j) = sfun_c(\mu(C_1,C_1'),\ldots,\mu(C_n,C_n'))$ where $C_1,\ldots,C_n$ are the concepts in O_i involved in the query formula, and $C_1',\ldots,C_n'$ are the corresponding concepts in O_j according to $M(O_i,O_j)$. $sfun_c$ is a t-norm as all the involved concepts are specified in the *triple_pattern* of f and triples can only be combined through conjunctions.

Starting from the queried peer, the system can access data on any peer in the network which is connected through a semantic path of mappings. When the query is forwarded through a semantic path, it undergoes a multi-step reformulation which involves a chain of semantic approximations. The semantic approximation given by a semantic path $p_1 \rightarrow p_2 \rightarrow \ldots \rightarrow p_m$ (in the following denoted as $P_{p_1\ldots p_m}$), where the submitted query formula f_1 undergoes a chain of reformulations $f_1 \rightarrow f_2 \rightarrow \ldots \rightarrow f_m$, can be obtained by composing the semantic approximation scores associated with all the reformulation steps: $s(f_1, P_{p_1\ldots p_m}) = sfun_r(s(f_1,p_2), s(f_2,p_3),\ldots,s(f_{m-1},p_m))$, where $sfun_r$ is a t-norm which composes the scores of query reformulations along a semantic path of mappings.

Summing up, given a query formula f submitted to a peer p, the set of accessed peers $\mathcal{P}' = \{p_1,\ldots,p_m\}$,[7] and the path $P_{p\ldots p_i}$ used to reformulate f over each peer p_i in $\mathcal{P}'$, the semantics of answering f over $\{p\} \cup \mathcal{P}'$ is the union of the query answers collected in each accessed peer: $Ans(f,p) \cup Ans(f,P_{p\ldots p_1}) \cup \ldots \cup Ans(f,P_{p\ldots p_m})$ where each answer $Ans(f,P_{p\ldots p_i})$ contains the set of the results collected in the accessed peer p_i together with the semantic approximation given by the path $P_{p\ldots p_i}$: $Ans(f,P_{p\ldots p_i}) = (Ans(f,p_i), s(f,P_{p\ldots p_i}))$. As observed before, as far as the starting queried peer is involved, no semantic approximation occurs as no reformulation is required (i.e. $s(f,p) = 1$).

3 Query Routing

In this section we define a query routing approach which operates on both semantic and multimedia data in an integrated way by first introducing the two approaches separately and then by meaningfully combining them.

3.1 Semantic Query Routing

Whenever a peer p_i selects one of its neighbor, say p_j, for query forwarding, the query moves from p_i to the subnetwork rooted at p_j and it might follow any of

[7] Note that $\mathcal{P}'$ not necessarily covers the whole network (i.e., $\mathcal{P}' \subseteq \mathcal{P}$).

the semantic paths originating at p_j. Our main aim in this context is to introduce a ranking approach for query routing which promotes the p_i's neighbors whose subnetworks are the most semantically related to the query.

In order to model the semantic approximation of p_j's subnetwork w.r.t. p_i's schema, the semantic approximations given by the paths in p_j's subnetwork are aggregated into a measure reflecting the relevance of the subnetwork as a whole. To this end, the notion of semantic mapping is generalized as follows. Let p_j^{Δ} denote the set of peers in the subnetwork rooted at p_j, O_j^{Δ} the set of schemas $\{O_{j_k} | p_{j_k} \in p_j^{\Delta}\}$, and $P_{p_i \ldots p_j^{\Delta}}$ the set of paths from p_i to any peer in p_j^{Δ}. The generalized mapping relates each concept C in O_i to a set of concepts C^{Δ} in O_j^{Δ} taken from the mappings in $P_{p_i \ldots p_j^{\Delta}}$, according to an *aggregated score* which expresses the semantic similarity between C and C^{Δ}.

Definition 2 (Generalized Semantic Mapping). *Let p_i and p_j be two peers, not necessarily distinct, and g an aggregation function. A* generalized semantic mapping *between p_i and p_j is a fuzzy relation $M(O_i, O_j^{\Delta})$ where each instance (C, C^{Δ}) is such that: 1) C^{Δ} is the set of concepts $\{C_1, \ldots, C_h\}$ associated with C in $P_{p_i \ldots p_j^{\Delta}}$, and 2) $\mu(C, C^{\Delta}) = g(\mu(C, C_1), \ldots, \mu(C, C_h))$.*

The aggregation function g is a continuous function on fuzzy sets which satisfies the monotonicity, the boundary condition, the symmetry and the idempotence properties. Several choices are possible for g satisfying the above properties, for instance functions such as the min, the max, any generalized mean (e.g. harmonic and arithmetic means), or any ordered weighted averaging (OWA) function (e.g. a weighted sum) [6].

Therefore, each peer p maintains a matrix named *Semantic Routing Index (SRI)*, which contains the membership grades given by the generalized semantic mappings between itself and each of its neighbors $Nb(p)$ and which is used as a routing index. A portion of Peer1's SRI of the reference example is shown below:

SRI_{Peer1}	Tile	origin	company	price	material	size	image
Peer1	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Peer2	0.85	0.70	0.83	0.95	0.83	0.92	1.0
Peer3	0.65	0.85	0.75	0.86	0.95	0.74	1.0

Besides the first row, which represents the knowledge of Peer1's local schema, it contains two entries, one for the upward subnetwork rooted at Peer2, and one for the downward one rooted at Peer3. For instance, from the stored scores, it follows that Peer3's subnetwork better approximates the concept *origin* (score 0.85) than Peer2's one (score 0.70). More details about the management of SRIs can be found in [6].

Thus, when a peer p receives a query formula f, it exploits its SRI scores to determine a ranking $R_{sem}^{p}(f)$ for its neighborhood, so as to identify the directions which best approximate f. More precisely, for each neighbor p_i an overall score is computed by combining, by means of the scoring function $sfun_c$, the scores the SRI row $SRI[p_i]$ associates with the concepts $C_1, \ldots, C_n$ in f: $R_{sem}^{p}(f)[p_i] = sfun_c(\mu(C_1, C_1^{\Delta}), \ldots, \mu(C_n, C_n^{\Delta}))$. Intuitively, the higher is the overall score, the

more likely the peer's subnetwork will provide semantically relevant results to the query.

3.2 Multimedia Query Routing

The execution of multimedia predicates is inherently costly, both from a CPU and from an I/O point of view. Adopting a broadcast-based approach to solve multimedia predicates in the network could thus imply wasting precious resources. Instead, we introduce a routing mechanism for efficient similarity search in PDMS. The approach can be exploited to solve most multimedia predicates as it has been conceived for metric spaces. In a metric space, the similarity between two objects is evaluated according to their pairwise distance: the lower their distance, the higher their similarity. The key idea is to build a distributed index that provides a concise but yet sufficiently detailed description of the multimedia resources available in a given network area. This information is then used at query time to forward a query containing multimedia predicates only towards those directions with the highest number of potential matchings.

To this end, for a given peer p, for each multimedia object X in p's local dataset (e.g., an image), we extract one or more features. We represent each feature F_i of an object X as an element of a metric space and we denote it as X^{Fi}. For each feature F_i, each peer builds different feature indices, in order to allow also for multi-feature queries. Each index exploits m *reference objects* $R_k^{F_i}$, with $k = 1 \dots m$, i.e. objects that are used to determine the position of other objects in the metric space. For ease of presentation, let us consider the case of a single feature. For simplicity, we also drop the symbol F_i from the following formulations, and when there is no possibility of confusion we use the same symbol X for indicating both the multimedia object and its associated metric feature. Formally, if we consider distances from a reference object R_k in the interval $[a, b]$, we select $h + 1$ division points $a = a_0 < a_1 < \ldots < a_h = b$ such that $[a, b]$ is partitioned into h disjoint intervals $[a_i, a_{i+1})$, $i = 0, 1, \ldots, h - 1$. Given a peer p, for each reference object R_k we build the histogram $\mathit{FeatureIdx}(p)_{R_k}$, which measures the number of objects X for which $d(X, R_k) \in [a_i, a_{i+1})\ \forall i$. This index gives us a concise description of all the data owned by a peer and it represents how the peer's objects are distributed with respect to the reference object R_k.

Each peer p also maintains, for its neighborhood $\mathit{Nb}(p)$, a set of Multimedia Routing Indices (MRIs), one for each reference object R_k. Any MRI row $\mathit{MRI}(p, p_i^{\triangle})_{R_k}$, represents the aggregated description of the resources available in the subnetwork $p_i^{\triangle}$ rooted at $p_i \in \mathit{Nb}(p)$ and is built by summing up the index for R_k of the peers in the subnetwork.

$$\mathit{MRI}\left(p,\ p_i^{\triangle}\right)_{R_k} \equiv \mathit{FeatureIdx}(p_i)_{R_k} + \sum_{p_j \in \mathit{Nb}(p_i) - p} \mathit{MRI}\left(p_i, p_j^{\triangle}\right)_{R_k} \tag{1}$$

As an example, consider the MRI of Peer1 represented in the following table, in which we give the number of the total objects in each subnetwork, for each distance interval of the reference object R_1.

$\text{MRI}_{\text{Peer1}}$ / R_1	$[0.0, 0.2)$	$[0.2, 0.4)$	$[0.4, 0.6)$	$[0.6, 0.8)$	$[0.8, 1.0]$
Peer2	8	29	1,300	145	2
Peer3	300	1,512	121	3	0

For our query routing purposes, we focus on similarity-based *Range Queries* over metric objects, defined as follows: given the multimedia object Q and the range r specified as argument of any `LIKE` predicate, the query has to retrieve $\{X \mid d(X, Q) \leq r\}$[8]. For a given range query, the values $d(Q, R_k) - r$ and $d(Q, R_k) + r$ are computed, for each $k = 1, \ldots, m$. The vector representation of each query index $\mathit{QueryIdx}(Q)_{R_k}$ is then built by setting to 1 all the entries that correspond to intervals that are covered (even partially) by the requested range. All the other entries are set to 0. This index has the same form of the histogram $\mathit{FeatureIdx}(p)_{R_k}$ but only contains values in $\{0, 1\}$.

When a peer p receives a query formula f containing a `LIKE` predicate, instead of flooding the network by forwarding f to all its neighbors, p matches the indices of Q with the corresponding routing indices of its neighborhood. The matching phase outputs a score that suggests the degree of relevance of the query with respect to each of the possible forwarding directions. More precisely, p determines a ranking $R^p_{mm}(f)$ for its neighborhood, by taking the minimum of the products (element by element) of the indices $\mathit{QueryIdx}(Q)_{R_k}$ and $\mathit{MRI}(p, p_i^{\triangle})_{R_k}$ for each neighbor $p_i \in \mathit{Nb}(p) = \{p_1, \ldots, p_n\}$ and each reference object R_k, and then evaluating the following ratio:

$$R^p_{mm}(f)[p_i] = \frac{\min_k \left[\mathit{QueryIdx}(Q)_{R_k} \cdot \mathit{MRI}(p,\ p_i^{\triangle})_{R_k}\right]}{\sum_{j=1..n} \min_k \left[\mathit{QueryIdx}(Q)_{R_k} \cdot \mathit{MRI}\left(p,\ p_j^{\triangle}\right)_{R_k}\right]} \tag{2}$$

$R^p_{mm}(f)[p_i]$ gives an intuition of the percentage of potential matching objects underneath the subnetwork rooted at p_i with respect to the total objects retrievable through all the peers in $\mathit{Nb}(p)$.

3.3 Combined Query Routing

Whenever a peer p receives a query, both the semantic and the multimedia routing approaches associate each p's neighbor a score quantifying the semantic relevance and the percentage of potential matching objects in its subnetwork, respectively. This allows p to rank its own neighbors w.r.t. their ability to answer a given query effectively, i.e. minimizing the information loss due to its reformulation along semantic mappings, and efficiently, i.e. minimizing the network load due to the exploration of useless subnetworks.

Thus, since both the semantic and multimedia scores induce a total order, they can be combined by means of a proper aggregation function in order to obtain a global ranking. More precisely, given the two distinct rankings $R^p_{sem}(f)$ and $R^p_{mm}(f)$ computed for the query formula f on peer p we need an *aggregation function* $\oplus$ which, when applied to $R^p_{sem}(f)$ and $R^p_{mm}(f)$, provide a $R^p_{comb}(f)$ reflecting

[8] For ease of presentation, in the following we assume that each query formula f contains at most one `LIKE` predicate.

the overall goodness of the available subnetworks: $R^p_{comb}(f) = \alpha \cdot R^p_{sem}(f) \oplus \beta \cdot R^p_{mm}(f)$, where α and β can be set in order to give more relevance to either the semantic or multimedia aspect.

In [2] it is stated that optimal aggregation algorithms can work only with monotone aggregation function. Typical examples of these functions are the min and mean functions (or the sum, in the case we are not interested in having a combined grade in the interval $[0, 1]$). As an example of how the aggregation process works, let us go back to the sample query in Figure 1 and suppose Peer1 obtains the scores in the following table.

	$\text{SRI}_{\text{Peer1}}$	$\text{MRI}_{\text{Peer1}}$	$min()$
Peer2	0.70	0.53	0.53
Peer3	0.65	0.76	0.65

The rankings computed through SRI and MRI are Peer2-Peer3 and Peer3-Peer2, respectively. If we use the standard fuzzy conjunction min, we compute the following final ranking: Peer3-Peer2. As a result, the most promising subnetwork will be the one rooted at neighbor Peer3.

The obtained ranking reflects the foreseen subnetworks ability in solving the received query both at schema (SRI-based information) and at multimedia (MRI-based information) levels and can thus be properly tailored in order to implement clever routing strategies. This is the subject of the following section.

4 Routing Strategies

Starting from the queried peer, the objective of any query processing mechanism is to answer requests by navigating the network until a stopping condition is reached. A query is posed on the schema of the queried peer and is represented as a tuple $q = (id, f, \tau, nres)$ where: id is a unique identifier for the query; f is the query formula; $nres$ is the stopping condition specifying the desired number of results as argument of the `LIMIT` clause; and τ is an optional relevance threshold. Then, query answers can come from any peer in the PDMS that is connected through a semantic path of mappings.

In this context, the adoption of adequate query routing strategies is a fundamental issue. Indeed, Sec. 2 shows that any peer satisfying the query conditions may add new answers and different paths to the same peer may yield different answers. More precisely, at each reformulation step, a new peer $p_{Next} \in \mathcal{P}$ is selected, among the unvisited ones, for query forwarding. The adopted routing strategy is responsible for choosing p_{Next}, thus determining the set of visited peers $\mathcal{P}'$ and inducing an order ψ in $\mathcal{P}'$: $[p_{\psi(1)}, \ldots, p_{\psi(m)}]$. Further, the visiting order determines the path $P_{p \ldots p_j}$ which is exploited to reach each peer $p_j \in \mathcal{P}'$ and, consequently, the set of returned local answers and the semantic approximation accumulated in reaching p_j, i.e. $Ans(f, P_{p \ldots p_j})$.

Then, in order to increase the chance that users receive first the answers which better satisfy the query conditions, several routing policies can be adopted. More precisely, as we explained in Sec. 3.3, when a peer p receives a query q it exploits

its indices information to compute a ranking $R^p_{comb}(f)$ on its neighbors expressing the goodness of their subnetworks w.r.t. the query formula f.

Afterwards, different query forwarding criteria relying on such ranked list are possible, designed around different performance priorities. In [8] two main families of navigation policies are introduced: The *Depth First* (*DF*) query execution model, which pursues efficiency as its main objective, and the *Global* (*G*) model, which is designed for effectiveness. Both approaches are devised in a distributed manner through a protocol of message exchange, thus trying to minimize the information spanning over the network.

In particular, the DF model is efficiency-oriented since its main goal is to limit the query path. More precisely, with the aim of speeding up the retrieval of some (but probably not the best) results, the DF model only performs a local choice among the neighbors of the current peer, i.e. it exploits the only information provided by $R^p_{comb}(f)$.

Differently from the DF one, in the G model each peer chooses the best peer to forward the query to in a "global" way: It does not limit its choice among the neighbors but it considers all the peers already "discovered" (i.e. for which a navigation path leading to them has been found) during network exploration and that have not been visited yet. More precisely, given the set V of visited peers, it exploits the information provided by $\bigcup_{p \in V} R^p_{comb}(f)$ in order to select, at each forwarding step, the best unvisited peer. Obviously, going back to potential distant peers has a cost in terms of efficiency, but always ensures the highest possible effectiveness, since the most promising discovered peers are always selected.

According to our query answering semantics, $\mathcal{P}'$ is thus defined as the ordered set of visited peers $[p_{\psi(1)}, \ldots, p_{\psi(m)}]$ such that $|\{Ans(f,p) \cup Ans(f,p_{\psi(1)}) \cup \ldots \cup Ans(f,p_{\psi(n)})\}| \geq nres$ and $|\{Ans(f,p) \cup Ans(f,p_{\psi(1)}) \cup \ldots \cup Ans(f,p_{\psi(n-1)})\}| < nres$, where the ordering function ψ is given by the adopted routing strategy.

As to the optional threshold τ, when $\tau > 0$, those subnetworks whose relevance (in terms of combined routing score) is smaller than τ are not considered for query forwarding. The underlying reason is that a forwarding strategy which proceeds by going deeper and deeper toward poorly relevant network areas (i.e. not very semantically related to the query and containing few multimedia matching objects) can exhibit bad performances and, thus, it is better to start backtracking towards other directions. The adoption of a threshold τ may thus positively influence the composition of $\mathcal{P}'$, since "poorer" subnetworks are not considered for query forwarding. On the other hand, a not-null threshold introduces a source of incompleteness in the querying process, as the pruned subnetworks might contain matching objects. Completeness can instead be guaranteed when $\tau = 0$, since subnetworks with a 0 routing score can be safely pruned.

5 Experiments

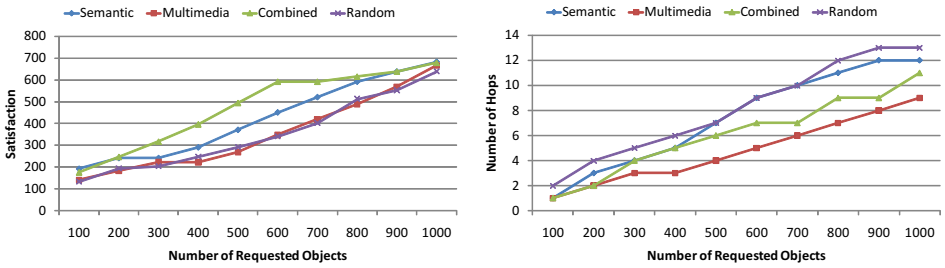
In this section we present an initial set of experiments we performed in order to evaluate our combined query routing approach. Notice that, since we are currently in the initial phase of our testing, the considered scenarios are not particularly complex; in the future we will enrich them with more complicated and larger ones.

For our experiments, we exploited our simulation environments for putting into action the SRI [6, 7] and MRI [3, 4] approaches. Through these environments we modelled scenarios corresponding to networks of semantic peers, each with its own schema, consisting of a small number of concepts, and a repository of multimedia objects. As to the multimedia contents, we use few hundreds of images taken from the Web and characterized by two MPEG-7 standard features: scalable color and edge histogram. We tested our techniques on different alternative network topologies, randomly generated with the BRITE tool (`http://www.cs.bu.edu/brite/`), whose mean size was in the order of few dozens of nodes. In order to evaluate the performance of our techniques we simulated the querying process by instantiating different queries on randomly selected peers and propagating them until their stopping condition on the number of retrieved results is reached: We evaluated the effectiveness improvement by measuring the semantic quality of the results (satisfaction) and, on the other hand, the efficiency improvement by measuring the number of hops performed by the queries. Satisfaction is a specifically introduced quantity which grows proportionally to the goodness of the results returned by each queried peer: Each contribution is computed by combining the semantic mapping scores of the traversed peers (see [6]). The search strategy employed is the depth first search (DF). In our experiments we compare our neighbor selection mechanism based on a combination of SRIs and MRIs (*Comb*) with the two mechanisms which only exploit the SRI (*SRI*) and MRI (*MRI*) values and with a baseline corresponding to a random strategy (*Rand*). The employed aggregation function is the mean. Notice that all the results we present are computed as a mean on several query executions.

Figure 2-a represents the trend of the obtained satisfaction when we gradually vary the stopping condition on the number of retrieved results. As we expected, the *Rand* and the *MRI* strategies show a similar poorly effective behavior since both select the subnetworks to explore without considering their semantic relevance. As we expected, they are thus outperformed by the *SRI* strategy which, on the contrary, is able to discriminate at each step the semantically best direction and, thus, increases the satisfaction in a substantial way. Nevertheless, the *Comb* routing reveals itself as the most effective one: it works by considering in an integrated way semantic and multimedia information and, consequently, tends to cover shorter paths which inherently have a lower approximation (and, thus, a higher satisfaction).

As to the efficiency evaluation, Figure 2-b represents the trend of the hops required for satisfying queries. Also this time, the *Rand* routing exhibits the worst behavior while the *SRI* one, which has no kind of knowledge on multimedia data, often comes closer to it. Though being poorly effective, the *MRI* strategy is instead the most efficient one, since, for each query, it selects the subnetworks with the higher number of (even not semantically good) multimedia matching objects. On the other hand, the lower efficiency of the *Comb* routing is motivated by the fact that it wastes more hops in searching semantically relevant objects.

Summing up, the *Comb* strategy represents the best alternative and proves to be able to increase the chance to retrieve first the answers which better satisfy the query conditions



(a) Obtained satisfaction trends

(b) Required hops trends

Fig. 2. Effectiveness and efficiency evaluation

6 Concluding Remarks

We presented an innovative approach for processing queries effectively and efficiently in a distributed and heterogeneous environment, like the one outlined in the NeP4B project. As far as we know, this is the first research proposal specifically devised to enhance the processing of queries in a network of semantic peers which share both semantic and multimedia data.

References

1. C. Doulkeridis, A. Vlachou, Y. Kotidis, and M. Vazirgiannis. Peer-to-peer similarity search in metric spaces. In *VLDB*, 2007.
2. R. Fagin, A. Lotem, and M. Naor. Optimal Aggregation Algorithms for Middleware. *Journal of Computer and System Sciences*, 66:47–58, 2003.
3. C. Gennaro, M. Mordacchini, S. Orlando, and F. Rabitti. MRoute: A Peer-to-Peer Routing Index for Similarity Search in Metric Spaces. In *Proc. of DBISP2P*, 2007.
4. C. Gennaro, M. Mordacchini, S. Orlando, and F. Rabitti. Processing Complex Similarity Queries in Peer-to-Peer Networks. In *Proc. of SAC*, 2008.
5. A. Halevy, Z. Ives, J. Madhavan, P. Mork, D. Suciu, and I. Tatarinov. The Piazza Peer Data Management System. *IEEE TKDE*, 16(7):787–798, 2004.
6. F. Mandreoli, R. Martoglia, W. Penzo, and S. Sassatelli. SRI: Exploiting Semantic Information for Effective Query Routing in a PDMS. In *Proc. of WIDM*, 2006.
7. F. Mandreoli, R. Martoglia, W. Penzo, and S. Sassatelli. Data-sharing P2P Networks with Semantic Approximation Capabilities. *To appear in IEEE Internet Computing*, 2009.
8. F. Mandreoli, R. Martoglia, W. Penzo, S. Sassatelli, and G. Villani. SRI@work: Efficient and Effective Routing Strategies in a PDMS. In *Proc. of WISE*, 2007.
9. S. Montanelli and S. Castano. Semantically routing queries in peer-based systems: the h-link approach. *Knowledge Eng. Review*, 23(1):51–72, 2008.
10. W. Penzo. Rewriting Rules To Permeate Complex Similarity and Fuzzy Queries within a Relational Database System. *IEEE TKDE*, 17(2):255–270, 2005.