Intelligent Information Retrieval and Presentation with Multimedia Databases

Floris Wiesman
IKAT, Universiteit Maastricht
P.O. Box 616, 6200 MD
Maastricht, The Netherlands
wiesman@cs.unimaas.nl

Stefano Bocconi
Centrum voor Wiskunde en Informatica
P.O. Box 94079, 1090 GB
Amsterdam, The Netherlands
stefano.bocconi@cwi.nl

Boban Arsenijevic
ULCL, Leiden University
P.O. Box 9515, 2300 RA
Leiden, The Netherlands
b.arsenijevic@let.leidenuniv.nl

ABSTRACT

The paper introduces a knowledge-based multimedia approach to multimedia information retrieval. The approach uses domain knowledge to augment a user's query, performs automatic ontology mapping to search different multimedia databases, and combines the results in a multimedia presentation. The texts in the presentation are generated from the domain knowledge. Thus, the user can view a coherent multimedia *presentation* that contains the answer to his or her query. The paper describes an architecture for realizing the approach. The individual parts of the architecture have been implemented, but are not yet integrated in one system.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Selection process; H.3.5 [**Online Information Services**]: Web-based services; H.5.1 [**Multimedia Information Systems**]: Miscellaneous

General Terms

Design, Algorithms, Theory

Keywords

semantic web technologies, multimedia presentations, ontology mapping, natural language generation

1. FROM MULTIMEDIA SEARCH TO MULTIMEDIA PRESENTATIONS

Multimedia presentations utilize a combination of several media, which results in shared load of the different perceptional channels [7] and reduction of cognitive memory load [6], thus ultimately in conveying effectively information. Text-only information retrieval systems have retrieval modes that show query results in various granularities: in the coarse extreme they show links to documents, ranked with respect to their relevance to the query. In the fine-grained extreme they provide an exact answer to the query (question-answering system). In between are the passage-retrieval systems. In this paper we introduce a retrieval mode that is related to passage retrieval and question answering but which is especially geared to multimedia information.

In brief, our approach uses domain knowledge to augment the query, performs automatic ontology mapping to search different multimedia databases, and combines the results in a multimedia presentation. The texts in the presentation are generated from the domain knowledge and the various ontologies. Thus, the user can view a multimedia *presentation* that contains the answer to his or her query.

The remainder of this paper is organized as follows. In Section 2 we elaborate on our approach and present our architecture. Three parts of the architecture are dealt with in separate sections: presentation generation (Section 3), natural-language generation (Section 4), and ontology mapping (Section 5). Finally Section 6 provides conclusions and directions for future work.

2. THE I²RP ARCHITECTURE

Our approach to multimedia information retrieval can best be described on the basis of our architecture, called the I²RP architecture.[1] It is depicted in Figure 1. From left to right the figure shows the course of query processing to the presentation of the results.

The user starts with formulating a query. The *query processor* augments the query with domain knowledge by retrieving also elements that are semantically related to the query term. The domain knowledge is stored in a semantic network, which is served by the *ontology agent*. This agent also has access to various multimedia databases. For each database the agent automatically creates a mapping such that items in the databases are linked to concepts in the semantic network. Thus, retrieving information from the databases becomes an inferencing task. The result is a subgraph of the semantic network that contains the answer to

[1] I²RP is the project acronym, which stands for *Intelligent Information Retrieval and Presentation in Public Historical Multimedia Databases.*

DIR 2003, 4th Dutch-Belgian Information Retrieval Workshop.

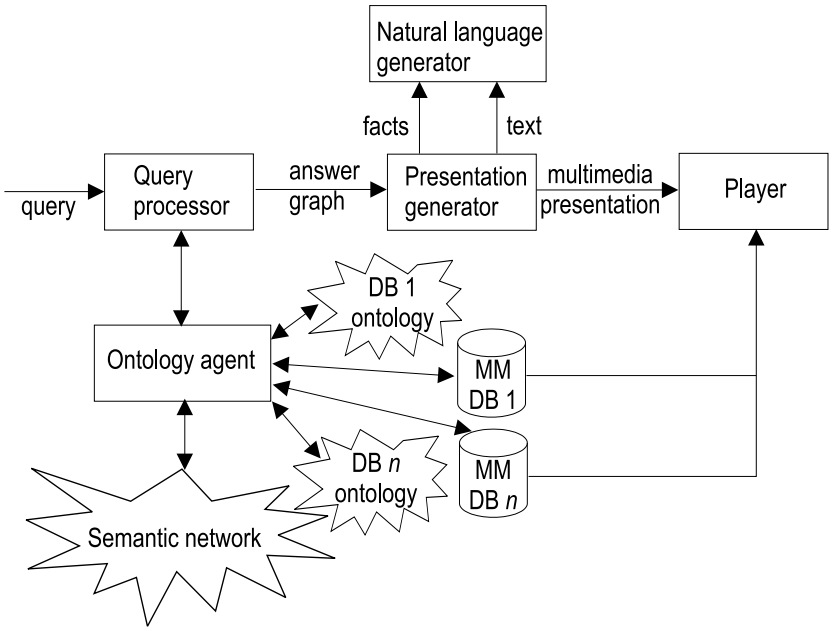

Figure 1: The I^{2}RP architecture.

the query; we call this subgraph the *answer graph*.

The next task is to generate a presentation from the answer graph. This is done by the *presentation generator*. A presentation may comprise written text, sound, pictures, and movies. The presentation generator decides how to combine the information from the answer graph into one presentation that conveys the information to the user as well as possible. Complete texts may be taken from the databases, but the texts may also be generated from the answer graph on the fly. This is the task from the *natural language generator*. It receives a selection of facts from the answer graph and using knowledge of syntax and semantics it generates natural language texts that are incorporated in the presentation.

Finally, the presentation can be viewed by the user with a multimedia player. Since the presentation does not contain actual multimedia items but only links to them, the player accesses the databases while playing the presentation.

Although our project is limited to museums as test domain, the architecture has a wider scope. This explains the generic character of the natural language generator and the ontology agent.

The different parts of the architecture are developed by different groups. The query processor and presentation generator come from CWI, the natural language generator is a product of Leiden University, and the ontology agent is from the Universiteit Maastricht. They are further discussed below.

3. QUERY PROCESSING AND PRESENTATION GENERATION

The I^{2}RP system generates multimedia presentations about a user-specified subject using a semantically annotated knowledge source. Users start off specifying via a web interface the topics they are interested in. The system then accesses the knowledge source to retrieve relevant information items (the query processor in Figure 1) and structures them in a presentation (the presentation generator in Figure 1). Figure 2 shows a multimedia player screen with an example presentation. Each knowledge source available is described by an ontology, here called domain ontology. The ontology agent guarantees that all information sources use the same domain ontology.

Retrieving information based on a domain ontology makes it possible to retrieve items which might not contain information about the main topic of the query but are semantically related (sometimes indirectly via multiple steps) to it. For example, a presentation about Rembrandt's biography might include a description of his student Jan Lievens even if this information item does not contain any reference to Rembrandt, but is annotated with a semantic relation 'studentOf'. Inferencing on the semantic relations can also help to discover relevant items; for example, if A is spouseOf B and B is sonOf C, then A and C are also relatives.

This is not the only way the domain ontology can serve the purpose of information retrieval: if elements are retrieved because of their semantic relations with the topics of the presentation and with each other, these semantic relations should be preserved when presenting the results to the users, translating the semantic relations in spatio-temporal relations (related items are presented in spacial or temporal proximity). A ranked list will very likely not preserve semantic relations among the retrieved items. Again using the example of Rembrandt, in a list a relevant information item (e.g., text or image) about Rembrandt can be ranked as first, while a less relevant information item about Rembrandt's son can be ranked much lower (or excluded from the list). It would be better to have an ordered presentation

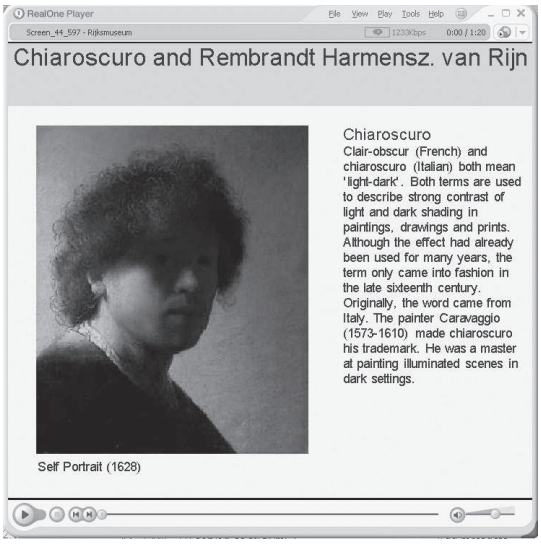

Figure 2: A Multimedia Presentation. The format is SMIL and the player is RealOne.

about Rembrandt's life with the two information items next to each other (assuming the focus is on Rembrandt's private life and not on his career).

A domain ontology can thus be used to recreate a coherent context (i.e., the presentation) for the information items, where coherent context means that the structure of the presentation has a semantic motivation. Our approach to provide a coherent context is to use narrative theory [4]. The idea is that organizing a presentation to tell a story requires the story and the presentation to be coherent, that is, to communicate a message to the user in a familiar and logical way.

The presentation's narrative in the presentation generator is created by defining what genre the presentation should belong to, for example, a biography or a curriculum vitae, and then determining who the main actors are in the story. For example, in a biography of an artist, the system knows as roles the main character and his/her family members, teachers, collaborators, and students. Successively the presentation generator asks the query processor to find information items related to these roles in the knowledge base. If the query processor finds them, the presentation generator includes them in the presentation and it structures them according to their role (e.g., all family members are grouped in the private life section).

The core functioning mechanism is the selection of the roles to include in the presentation. The selection is rule based: rules define the conditions for an information item to get a role in the presentation. If a rule is satisfied, the information item is selected and assigned a role. For example, a rule constructing a private life narrative unit could define that any element X which has an isMarried relation to any element Y with role 'main character' is assigned the role of Spouse. Other rules can then be applied on the newly created role or on other roles in the presentation.

At the current stage the rules are straightforward and check for particular semantic relations among information items, but the plan is to extend them to take into consideration relations among more items (e.g., composing more rules with boolean operators). Such rules could also assess the relevance of the information items based on their relations with other information items, while at the current stage elements either match or do not match a rule (the rules we use are described more in depth in [3]).

An important aspect in generating multimedia presentations is that each building unit (information item) of the generated narrative can be of different modality (text, picture, audio, etc.) and the dependencies and referentiality that exist between the modalities influence the meaning of the presentation.

The transformation from a pre-media structure (which in our approach is the narrative structure) to a media-dependent structure (which we call presentation structure) is made in the presentation generator and is based on rules. These rules determine the choice of a particular modality (or combination of modalities) by mapping features describing types of information to features describing the inherent structure of each modality. Thus each type of information is presented with the modality that best conveys its meaning.

When the presentation generator has decided upon the structure of the presentation, it provides the Natural Language Generator (described in the next section) with facts for the presentation (e.g., date of birth, place of birth) and the natural-language generator generates the texts to be included in the presentation. The final content is encoded in SMIL (Synchronized Multimedia Integration Language [12]) and served to the user.

4. NATURAL LANGUAGE GENERATION

Natural-language generation (NLG) is only related to the textual level of multimedia information retrieval, but it improves this level in several different aspects. The NLG-related subproject of I^2RP is named Spreekbuis and its focus is on developing an algorithm for semantically based NLG. Therefore the stress of the research is on the field of computational semantics and the semantics-syntax interface, especially on how the semantic form can provide the relevant information for syntactic realization.

Our project develops a NLG system that starts from the level of semantics and aims to transform a selected meaning into a natural-language sentence. A semantic representation, as complete as possible (containing the participating concepts, the event-structure information, the temporal organization, quantification, and the relevant discourse functions), is to be transformed into a form supplied with syntactic functions and lexical material, which is further used as a base for realization of a sentence.

One of the most important and most difficult tasks for the interface between semantics and syntax is to preserve the proper mapping of the semantic relations in the form provided for syntactic realization, so that realized sentences fully reflect the meaning from the semantic representation. For example, the meaning of 'Pieter Lastman gave classes

to Rembrandt' should not be realized as 'Rembrandt gave classes to Pieter Lastman', although they have the same conceptual participants. Working on a semantic network with an adequate notation, such generation would ideally be able to realize through natural language any piece of information selected from the database, without much irrelevant content. Instead of the usual way of providing the results to the user, where often large pieces of text containing the requested information are found and displayed, the results can be condensed into sentences that answer the the user's query.

Combined with a parser that outputs the same type of semantic representation that the generator uses as its input, the generator is able to work on both structured and unstructured databases. In a question-answering system, it could be implemented in the following way: questions asked in natural language are parsed, and fed to the query processor; after it returns the candidate passages of text (excerpted from the unstructured database in the way it is already done within the I^2RP architecture), the text is parsed and its parts matched to the parse of the question. The matching semantic contents are then used to generate sentences. Matching semantic parses of the question and the retrieved information could use the help of an inferencing system to achieve a wider range of matching possibilities.

Currently Spreekbuis encompasses two generators: the Performance Grammar Generator [5] and Delilah [2]. Delilah is a very robust parser for Dutch; it parses sentences to and generates them from a semantic form. The major research task at the moment is to preserve the full semantics of the input in syntactic realization. Our plan is first to develop the algorithm and if possible also the software that will provide a more information-preserving interface between the semantic form and its realization in natural language, particularly with respect to the argument structure and adjunct-argument distinctions. In other words, we want to determine within a semantic form which participating concepts are to be realized as arguments and which as adjuncts. For instance, we do not want to get a sentence like 'Rembrandt inhabited Leiden being a student by Jacob van Swanenburch' for 'Rembrandt studied in Leiden under Jacob van Swanenburch'.

5. ONTOLOGY MAPPINGS

The ontology agent provides uniform access to the domain representation (a semantic network) and the various multimedia databases. In the ideal case we start with the semantic network and then automatically map the databases to the semantic network. Thus any multimedia record is accessible from the semantic network.

Currently, we cannot make a mapping from a database to the semantic network automatically; therefore, this mapping is established manually. Once a mapping to one database exists, we can establish mappings to other databases automatically. The procedure described below uses one agent for each database/ontology for clarity; in the I^2RP architecture one agent does all the work.

Figure 3 illustrates some forms of *semantic heterogeneity* that must be solved to establish a mapping: different concept names are used for the same data and data may be structured differently. To obtain a mapping we must be able to split and merge data fields. For instance, the concept 'date' in ontology 1 containing the data '1661–1662' must be split into '1661' and '1662' in order to map 'date' in ontology 1 to 'start' and 'end' in ontology 2. The inverse mapping requires merging 'start' and 'end'. None of the approaches proposed in the literature, e.g. [9, 8, 11], offers an adequate solution.

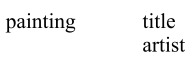

Figure 3: Ontologies 1 and 2.

Suppose that agent 1 wishes to know the artist's name and the material of some paintings. Agent 1 knows that the information is (probably) available in a database managed by agent 2. Therefore, agent 1 contacts agent 2. In order for agent 1 to put forward its request, the agents first have to establish whether both use the same ontology or whether they use an ontology of which the other agent knows how to map it on its ontology. If the agents use different ontologies and if no mapping is known, the agents should try to establish a mapping. The way the agents establish a mapping is inspired by language games [10].

To illustrate the idea behind language games for ontology mapping, suppose that two agents wish to communicate about the concept 'painting'. Moreover, the agents use different conceptualizations of the concept 'painting' (as depicted in Figure 3) and some paintings are known by both agents.

A concept such as a 'painting' may consist of a hierarchy of sub-concepts. For the *primitive* concepts in this hierarchy, an instance specifies the actual data values. For example, an instance could be a painting titled 'Self portrait as St. Paul', painted by Rembrandt Harmensz. van Rijn with oil on canvas. By finding an instance of the concept 'painting' known by both agents, the agents determine *joint attention*. The joint attention will be the basis of the language game. To establish the joint attention, agent 1 produces an utterance containing a unique representation of a concept and instance of the concept. Agent 2, upon receiving the utterance, investigates whether it has a concept of which an instance matches to a certain degree with the communicated instance. To do so, agent 2 measures the proportion of words that two instances have in common. The instance with the highest proportion of corresponding words together with the communicated instance constitute a joint attention – provided that the correspondence is high enough.

After establishing the set of joint attentions, agents 2 tries to establish a mapping between the primitive concepts that make up the concept. To do so, agent 2 needs an utterance from agent 1 and itself. An utterance for an instance is sim-

ply formed by a list of all words of the instance. Hence, the structure of the ontology plays no role. Next, agent 2 tries to establish associations between the different primitive concepts. Agent 2 generates associations between the primitive concepts of the two utterances on the basis of the proportion of corresponding words in pairs of primitive concepts, one from each utterance. Possible associations are:

field $x \leftarrow$ field y.
field $x \leftarrow$ field y, split(s), first.
field $x \leftarrow$ field y, split(s), last.
field $x \leftarrow$ field y, field z, merge (t).

Here, the operator field denotes the selection of a primitive concept where x, y, and z represent the primitive concepts to be selected. The operator split divides a data field into two sub-fields using the separator s to determine the point of division. We consider the following separators: ' ', ',', ';', and TC (a type change, i.e., a change from letters to digits or vice versa). After splitting a data field the operators first and last can be used to select either the first or the last subfield. The operator merge takes two data fields and merges them into one data field adding the separator t in between. As separators can be added: '', ' ', ',' and ';'. The following illustrates a mapping from agent 2 to agent 1.

field painting.date ← field painting.period.start, painting.period.end, merge('-').

Agent 2 searches through a space of possible associations guided by the proportion of words that instances of concepts have in common. Each new utterance from agent 1 enables agent 2 to update the strength of the associations. After having received a number of utterances, agent 2 may accept certain associations as being correct. Agent 2 has established a complete mapping from agent 1 to itself when it has a unique association for each primitive concept in its ontology.

6. CONCLUSIONS

This paper presented a new approach to information retrieval from multimedia databases. The main features of the approach are knowledge-based query augmentation, automatic mapping between the ontologies used, and combination of retrieval results in a single multimedia presentation. Texts in the presentation are generated by a natural-language component.

The various parts of our I^{2}RP architecture are realized as prototypes. What remains to be done is to combine them in one system. Further future work will concentrate on the query processor. Its searching abilities are currently limited. The natural language processing of queries could breech the gap between ontology-based queries and keyword-oriented queries. Finally, more sophisticated rules for the presentation generator will be investigated.

An important development that will contribute to the success of the I^{2}RP approach is the Semantic Web [1]. Originally devised as a means to improve information retrieval from the Web, semantic markup can also play a part in the presentation of information when it is combined with the I^{2}RP semantic network.

7. ACKNOWLEDGEMENTS

This research was carried out under the NWO ToKeN2000/I^{2}RP project (grant no. 634.000.002).

8. ADDITIONAL AUTHORS

Additional authors: Yulia Bachvarova (CWI, email: yulia.bachvarova@cwi.nl), Nico Roos (IKAT, Universiteit Maastricht, email: roos@cs.unimaas.nl) and Lambert Schomaker (AI, Rijksuniversiteit Groningen, email: schomaker@ai.rug.nl).

9. REFERENCES

[1] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5):35–43, 2001.

[2] C. Cremers. Dislocation, clustering and disharmony. In C. Retoreé and E. Stabler, editors, *Resource Logics and Minimalist Grammar (ESSLLI 99)*, 1999.

[3] J. Geurts, S. Bocconi, J. van Ossenbruggen, and L. Hardman. Towards Ontology-driven Discourse: From Semantic Graphs to Multimedia Presentations. In *Second International Semantic Web Conference (ISWC2003)*, Sanibel Island, Florida, USA, October 20-23, 2003. To be published.

[4] J. Greimas. *Structural Semantics: An Attempt at a Method*. Lincoln: University of Nebraska Press, 1983.

[5] G. Kempen and K. Harbusch. Dutch and German verb clusters in Performance Grammar. In P. Seuren and G. Kempen, editors, *Verb clusters in Dutch and German*. Benjamins, Amsterdam, The Netherlands, 2001.

[6] R. E. Mayer and R. Moreno. A Split-Attention Effect in Multimedia Learning: Evidence for Dual Processing Systems in Working Memory. *Educational Psychology*, 90(2):312–321, 1998.

[7] S. Y. Mousavi, R. Low, and J. Sweller. Reducing Cognitive Load by Mixing Auditory and Visual Presentation Modes. *Educational Psychology*, 87(2):319–334, 1995.

[8] M. P. Papazoglou, N. Russell, and D. Edmond. A translation protocol achieving consensus of semantics between cooperating heterogeneous database systems. In *Conference on Cooperative Information Systems*, pages 78–89, 1996.

[9] H. S. Pinto. Some issues on ontology integration. In *IJCAI-99 workshop on Ontologies and Problem-Solving Methods (KRR5)*, 1999.

[10] L. Steels and P. Vogt. Grounding adaptive language games in robotic agents. In C. Husbands and I. Harvey, editors, *Proc. of the Fourth European Conference on Artificial Life*. MIT Press, 1997.

[11] R. M. van Eijk, F. S. de Boer, and W. van der Hoek. On dynamically generated translators in agent communication. *International Journal of Intelligent Systems*, 16:587–607, 2001.

[12] W3C. Synchronized Multimedia Integration Language (SMIL 2.0) Specification. W3C Recommendation, August 7, 2001. Edited by Aaron Cohen.