PROGETTO DI UNA UNITÀ DI RICERCA - MODELLO B
Anno 2004 - prot. 2004095494_004

1.1 Tipologia del programma di ricerca

Interuniversitario

Aree scientifico disciplinari

Area 09: Ingegneria industriale e dell'informazione (100%)

1.2 Durata del Programma di Ricerca

24 Mesi

1.3 Coordinatore Scientifico del Programma di Ricerca

BERGAMASCHI ***SONIA*** ***sonia.bergamaschi@unimo.it***

ING-INF/05 - Sistemi di elaborazione delle informazioni

Università degli Studi di MODENA e REGGIO EMILIA

Facoltà di INGEGNERIA

Dipartimento di INGEGNERIA DELL'INFORMAZIONE

1.4 Responsabile Scientifico dell'Unità di Ricerca

MERIALDO ***PAOLO***

Ricercatore Universitario ***27/07/1965*** ***MRLPLA65L27D969W***

ING-INF/05 - Sistemi di elaborazione delle informazioni

Università degli Studi ROMA TRE

Facoltà di INGEGNERIA

Dipartimento di INFORMATICA E AUTOMAZIONE

06-65741221	***06-5573030***	***merialdo@dia.uniroma3.it***
(Prefisso e telefono)	*(Numero fax)*	*(Email)*

1.5 Curriculum scientifico del Responsabile Scientifico dell'Unità di Ricerca

Testo italiano
Paolo Merialdo e' Ricercatore presso la Facoltà di Ingegneria all'Universita' Roma Tre dal 2002. Si e' laureato in in Ingegneria Elettronica presso l'Universita' degli Studi di Genova nel 1990, ed ha conseguito il titolo di dottore di ricerca presso l'Universita' "La Sapienza" di Roma nel 1998, sotto la supervisione del prof. Paolo Atzeni. Ha trascorso un periodo di studio ospite del prof. Alberto Mendelzon presso la University of Toronto.

La sua attivita' di ricerca riguarda prevalentemente lo studio di tecniche e metodologie per la gestione di dati su World Wide Web, e lo studio di tecniche per la Estrazione di informazioni da sorgenti Web.

Ha pubblicato i suoi risultati di ricerca su autorevoli riviste del settore, incluse ACM Transactions on Internet Technolgy, IEEE Transactions on Knowledge and Data Engineering, IEEE Internet Computing, e negli atti delle principali conferenze (VLDB, ACM-SIGMOD, EDBT). E' stato membro dei comitati di programma di varie conferenze internazionali. Dal 1998 ad oggi e' Associate Director di ACM Sigmod Record.

Testo inglese
Paolo Merialdo is Research Associate at Universita' Roma Tre since 2002. He received his Computer Engineering degree from Universita' degli Studi di Genova in 1990. In 1998 he received his PhD, from Universita' di Roma "La Sapienza", under the supervision of prof. Paolo Atzeni. He also spent a period at the University of Toronto, working with prof. Alberto Mendelzon.

His research interests include information extraction and data management techniques for Web data.

He has published his research results in important journals of the field, including ACM Transactions on Internet Technology, IEEE Transaction on Knowledge and Data Engineering, IEEE Internet Computing, and in the refereed proceedings of the major conferences (ACM-SIGMOD, VLDB, EDBT). He has also been program committe member for many international conferences. He is currently serving as Associate Director for ACM SIGMOD-RECORD.

1.6 Pubblicazioni scientifiche più significative del Responsabile Scientifico dell'Unità di Ricerca

1. *ALBERTO MENDELZON; GIANSALVATORE MECCA; PAOLO MERIALDO (2002).* ***Efficient Queries over Web Views*** *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING. (vol. 14(6) pp. 1280-1298)*
2. *PAOLO ATZENI; GIANSALVATORE MECCA; PAOLO MERIALDO (2002).* ***Managing Web-Based Data: Database Models and Transformations*** *IEEE INTERNET COMPUTING. (vol. 6(4) pp. 33-37)*
3. *PAOLO ATZENI; GIANSALVATORE MECCA; PAOLO MERIALDO (2001).* ***Data-Intensive Web Sites: Design and Maintenance*** *WORLD WIDE WEB. (vol. 4(1-2) pp. 21-47)*
4. *VALTER CRESCENZI; GIANSALVATORE MECCA; PAOLO MERIALDO (2001).* ***RoadRunner: Towards Automatic Data Extraction from Large Web Sites*** *27th International Conference on Very Large Data Bases. pp. 109-118*
5. *PAOLO ATZENI; GIANSALVATORE MECCA; PAOLO MERIALDO (1997).* ***Semistructured und Structured Data in the Web: Going Back and Forth*** *SIGMOD RECORD. (vol. 26(4) pp. 16-23)*

1.7 Risorse umane impegnabili nel Programma dell'Unità di Ricerca

1.7.1 Personale universitario dell'Università sede dell'Unità di Ricerca

Personale docente

nº	Cognome	Nome	Dipartimento	Qualifica	Settore Disc.	Mesi Uomo	
						1° anno	2° anno
1.	*MERIALDO*	*Paolo*	*Dip. INFORMATICA E AUTOMAZIONE*	*Ricercatore Universitario*	*ING-INF/05*	*6*	*6*
2.	*ATZENI*	*Paolo*	*Dip. INFORMATICA E AUTOMAZIONE*	*Prof. Ordinario*	*ING-INF/05*	*3*	*3*
3.	*TORLONE*	*Riccardo*	*Dip. INFORMATICA E AUTOMAZIONE*	*Prof. Associato*	*ING-INF/05*	*3*	*3*
4.	*CABIBBO*	*Luca*	*Dip. INFORMATICA E AUTOMAZIONE*	*Prof. Associato*	*ING-INF/05*	*3*	*3*
	TOTALE					**15**	**15**

Altro personale

Nessuno

1.7.2 Personale universitario di altre Università

Personale docente

Nessuno

Altro personale

Nessuno

1.7.3 Titolari di assegni di ricerca

Nessuno

1.7.4 Titolari di borse

Nessuno

1.7.5 Personale a contratto da destinare a questo specifico programma

nº Qualifica	Costo previsto	Mesi Uomo		Note
		1° anno	2° anno	
1. *Borsista*	*22.500*	*5*	*10*	
2. *Borsista*	*18.000*	*4*	*9*	
TOTALE	**40.500**	**9**	**19**	

1.7.6 Personale extrauniversitario indipendente o dipendente da altri Enti

Nessuno

2.1 Titolo specifico del programma svolto dall'Unità di Ricerca

Testo italiano
Estrazione automatica di dati e schemi da sorgenti web data-intensive

Testo inglese
Automatic extraction of data and schemas from data-intensive web sources

2.2 Settori scientifico-disciplinari interessati dal Programma di Ricerca

ING-INF/05 - Sistemi di elaborazione delle informazioni

2.3 Parole chiave

Testo italiano
ESTRAZIONE DI DATI DA WEB ; WEB STRUCTURE MINING ; INFERENZA DI SCHEMI

Testo inglese
WEB DATA EXTRACTION ; WEB STRUCTURE MINING ; SCHEMA INFERENCE

2.4 Base di partenza scientifica nazionale o internazionale

Testo italiano
L'attività di ricerca dell'Unità si inserisce nel contesto di riferimento descritto nel Modello A della proposta. Più specificatamente, l'obiettivo è lo studio e lo sviluppo di strumenti e tecniche innovative per l'automazione del processo di estrazione di dati da sorgenti Web data-intesive.

La comunità di ricerca internazione ha mostrato un crescente interesse per lo studio di tecniche per la generazione automatica di programmi, chiamati wrapper, per l'estrazione di dati da pagine web. Proposte recenti hanno significativamente aumentato il livello di automazione del processo di generazione.

Le prime proposte presentano linguaggi ad hoc per la codifica manuale di wrapper (Atzeni, Mecca 1997), (Sahuguet, Azavant 1999), (Crescenzi, Mecca 1998). Successivamente l'attenzione si è spostata verso tecniche per la generazione semi-automatica di wrapper. Questi approcci sono basati su tecniche di Machine learning supervisionato (Kushmerick 1997), (Muslea et al. 1999), (Soderland 1999), che richiedono un coinvolgimento umano e assunzioni apriori sull'organizzazione dei dati nelle pagine (Adelberg 1998), (Embley et al. 1999). Infine, con il progetto RoadRunner (Crescenzi, et al 2001) si è cercato di automatizzare completamente la generazione del wrapper.

Questo lavoro ha mostrato come la generazione automatica dei wrapper a partire da un insieme di pagine web sia un problema riconducibile al problema di inferire una grammatica (regolare) a partire da un insieme di stringhe appartenenti al linguaggio che genera. Assieme ad altri lavori (Chidlovskii 2000), pone una maggiore attenzione verso i risultati di inferenza di grammatiche (Gold 1967), (Angluin 1980), (Fernau 2000) e che vanta oramai studi decennali.

La base di partenza dello studio sono proprio i risultati del progetto RoadRunner. Queste tecniche, che sono state implementate e sperimentate in un prototipo (Crescenzi et al. 2002), consentono di inferire automaticamente la grammatica che descrive l'organizzazione dei dati nelle pagine di un sito Web a partire da un campione di pagine esemplificative.

L'idea di fondo per la generazione automatica del wrapper nasce dall'osservazione che in un sito Web data-intensive le pagine sono generate da appositi script a partire da dati memorizzati in una base di dati; pagine generate dallo stesso programma hanno forti similarità nella struttura (perché tutte generate a partire dallo stesso schema di base di dati) pur presentando contenuti diversi (perché ciascuna pagina presenta una diversa parte della istanza di base di dati con tale schema). Si pensi ad esempio alle pagine relative ai prodotti in un sito di commercio elettronico, o alle pagine di risposta ad una form di ricerca in un sito di pagine gialle. Sulla base di questa osservazione è stato sviluppato un algoritmo che riceve in ingresso un insieme di pagine strutturalmente omogenee e, analizzandone similarità e differenze, è in grado di inferire un wrapper capace di estrarre i dati che originariamente risiedevano nel database.

(Arasu & Garcia-Molina 2003) hanno proposto un approccio differente. Le pagine web sono modellate come liste di token e sulla base di una analisi della frequenza delle occorrenze, i token sono raggruppati in classi di equivalenza. Ogni classe corrisponde ad un frammento di un template comune, e viene utilizzata per ricostruirne una descrizione complessiva. (Chang et Al. 2001) hanno sviluppato l'algoritmo IEPAD, un sistema che riconosce pattern per l'estrazione di dati da pagine web senza bisogno di esempi positivi forniti dall'utente. IEPAD si basa su varie tecniche per la scoperta di pattern, tra le quali tecniche per l'allineamento di strighe e per il pattern matching.

La disponibilità di sistemi che consentono di generare automaticamente un wrapper apre interessanti prospettive. Tuttavia, la costruzione automatica di wrapper per un gran numero di siti web solleva alcuni interessanti problemi. La generazione automatica di wrapper per un rilevante numero di siti tuttora solleva diverse questioni. Anzitutto, un problema che condiziona pesantemente la scalabilità dell'approccio consiste nel trovare le collezioni di pagine per alimentare i sistemi automatici per la generazione dei wrapper; attualmente, questi campioni di pagine sono scelti manualmente. Inoltre, una volta che si è generata una libreria di wrapper che copra un intero sito, bisogna saper scegliere quale wrapper utilizzare su una data pagina. Per finire, è necessario un meccanismo per navigare il sito che consenta di raggiungere pagine di un data classe; è quindi necessario un modello per descrivere i percorsi navigazionali tra le classi identificate.

In letteratura esistono diversi lavori che sono collegati a queste tematiche, anche se in ambiti diversi e con obiettivi differenti: modellazione dei dati di siti di grandi dimensioni, classificazione di documenti web, inferenza della struttura dei siti. Sebbene questi lavori siano correlati alle tematiche di studio proposte, nessuno degli approcci già esistenti risponde alle nostre necessità, come di seguito viene discusso.

Modellazione di siti web per l'estrazione di dati
La questione di modellare la struttura logica dei siti web per facilitare l'estrazione dei dati è stata studiata in diversi progetti di ricerca. Uno dei primi è il progetto Araneus (Atzeni et al. 1997, Atzeni et al 2002). Nel progetto Araneus una pagina web viene vista come un oggetto munito di identificatore (l'URL) ed un insieme di attributi. La nozione di schema di pagina viene introdotta per modellare classi di pagine omogenee. Gli attributi di una pagina possono essere di tipo semplice o composto. Gli attributi semplici corrispondono essenzialmente a testo, immagini o link. Gli attributi composti modellano collezioni (liste) di oggetti. Uno schema di sito è quindi una collezione di schemi di pagine. I dati sono estratti associando un wrapper a ciascuna schema di pagina. Nel progetto Araneus i wrapper venivavo costruiti utilizzando speciali linguaggi procedurali dedicati, chiamato Cut and Paste (Mecca & Atzeni 1999), o adottando Minerva (Crescenzi & Mecca 1999), un formalismo dichiarativo per la scrittura veloce di wrapper. La nozione di schema di sito risulta efficace per l'estrazione dei dati; gli schemi di pagina descrivono classi di pagine che condividono una medesima struttura, e gli attributi di tipo link che le connettono modellano i percorsi per raggiungere le istanze. Ad ogni modo, nel progetto Araneus gli schemi di un sito erano progettati manualmente, con una specifica fase di reverse-engineering.

Un contributo più recente è quello del progetto Wiccap (Liu et al 2002). In Wiccap, i dati sono proiettati su una struttura logica gerarchica. Il focus è sull'usabilità del modello: l'obiettivo è quello di mappare l'informazione di un sito web in una organizzazione logica di concetti, come percepiti da ordinari utenti. I nodi della struttura creata possono contenere dati estratti da diverse pagine, integrate per comporre concetti uniformi. I nodi sono associati con delle regole di mapping, vale a dire primitive per estrarre i dati dalla struttura fisica delle pagine e mapparli sul modello desiderato. La creazione delle viste gerarchiche risulta facilitata da un insieme di tool visuali.

Classificazione di documenti web sulla base della struttura
La classificazione di documenti HTML sulla base della struttura è stata recentemente affrontata in (Crescenzi et al 2002, Flesca et al 2002). Entrambi gli approcci considerano in input un ampio insieme di pagine HTML (o XML), e le raggruppano sulla base di proprietà legate alla frequenza ed alla distribuzione dei tag.

Bertino et Al. hanno proposto un approccio per la classificazione di documenti XML (Bertino et al 2004) basato su un algoritmo di matching per misurare la similarità strutturale tra un documento XML ed un DTD. L'algoritmo confronta la struttura del documento contro quella dettata dal DTD. Le similarità e le differenze sono valutate per calcolare un indice di similarità numerico. L'algoritmo di matching è quindi sfruttato per la classificazione di documenti XML rispetto ad un insieme di DTDs.

Inferenza della struttura di siti web
In (Liu et al 2004) è stato sviluppato un algoritmo, SEW, per scoprire lo scheletro di sito web. Lo scheletro descrive la struttura ipertestuale in cui i contenuti sono organizzati. Si assumono solo strutture gerachiche i cui nodi sono pagine di contenuto oppure pagine di navigazione. Le prime sono pagine che forniscono i contenuti informativi, mentre le seconde sono pagine che contegono i link necessari per navigare le prime. SEW cerca di scoprire automaticamente l'organizzazione gerarchica delle pagine di contenuto e di navigazione basandosi su una combinazione di euristiche, indipendenti dal dominio, per identificare i più rilevanti insiemi di link di ciascuna pagina. Rispetto ai nostri obiettivi, SEW distingue solo due tipi di classe di pagine predefinite, ovvero pagine navigazionali e pagine di contenuto, mentre noi siamo interessati a classificare le pagine in accordo alla loro struttura, senza alcuna assunzione sul numero di classi di pagine e sulla loro caratteristiche.

Un problema simile è stato studiato da Kao et al (Kao et al 2003), che hanno sviluppato una tecnica per analizzare la struttura di siti web di notizie. L'obiettivo della loro proposta è quello di identificare, sempre all'interno di un sito di notizie, le pagine di indice, e le pagine che contengono notizie. La tecnica si basa sull'analisi dell'entropia al fine di eliminare le ridondanze della struttura ipertestuale, e riesce cosi' a distillare gli aspetti salienti dalla complessità iniziale. Un tecnica collegata, anch'essa basata sull'analisi di entropia, elimina le informazioni ridondanti, come i pannelli navigazionali e le inserzioni pubblicitarie. Anche in questo caso l'approccio è valido solo per un certo tipo di topologia di sito web.

Testo inglese
The research activity of the Research Unit fits in the framework discussed in Model A of this proposal. More specifically, it aims at studying and developing tools and techniques for automatically extracting data from data-intensive data sources.

The international research community has recently shown an increasing interest in the study of techniques for the generation of web wrappers, i.e. programs that extract data from web pages and convert them into a more structured format (typically XML). In particular, recent works have significantly augmented the desidered level of automation of the wrapper generation process.

Early approaches proposed ad-hoc languages for manually coding the wrappers (Atzeni, Mecca 1997), (Crescenzi, Mecca 1998). Thereafter several researchers have studied semi-automatic techniques for wrapper generation. These studies resulted in supervisioned approaches of different inspirations in which the wrapper can be automatically generated only after the user has provided a set of positive examples (Kushmerick 1997), (Muslea et al. 1999), (Soderland 1999), eventually along with additional

information about the schema of data in the pages (Adelberg 1998), (Embley et al. 1999).

Recently, the RoadRunner Project (Crescenzi, Mecca, Merialdo 2001) has tried to push further the level of automation of wrapper generation. This project has also shown that generating a wrapper for a set of sample of sample web page corresponds to inferring a (regular) grammar for the HTML code in the page. This work, along with others (Chidlovskii 2000), seems to show an increasing attention towards the results produced by the grammar inference community (Gold 1967), (Angluin 1980), (Fernau 2000), a research field with a long tradition.

The results of the RoadRunner project for automatic wrapper generation are the starting point of our study. These techniques, which have been implemented and tested in a prototype (Crescenzi et al. 2002), allows us to automatically infer a grammar describing the organization of data in the pages of a data-intensive web site, based on a set of sample web pages. The main idea underlying the wrapper generation process is that pages from data-intensive web site are generated by specific programs called scripts; pages generated by the same script exhibit strong similarities in the structure (because they all are produced from the same database schema) but different contents (because each page presents different parts of the instance of that schema). To give an example, one can either consider the pages of e-commerce web site about products or the pages produced by a yellow-page web site as answer to a query. Basing on these considerations, it has been developed an algorithm which takes as input a set of structurally homogeneous web pages, analyzes and exploits similarities and differences of the given pages to infer a wrapper capable of extracting the data coming from the database.

To the best of our knowledge there are three different approaches that are similar in spirits to RoaRunner. Arasu and Garcia-Molina have recently proposed a different approach (Arasu & Garcia-Molina 2003). They represent a web page with a list of tokens. Their system analyzes the frequencies of occurrences of tokens in a set of sample pages to group tokens into equivalence classes. Each class of tokens corresponds to a fragment of the common template, which is then used by the system to infer an overall description of the template. (Chang et al. 2001) developed IEPAD (an acronym for Information Extraction based on PAttern Discovery), a system that discovers extraction patterns from Web pages without user-labeled examples. IEPAD applies several pattern discovery techniques, including PAT-trees, multiple string alignments and pattern matching algorithms. Extractors generated by IEPAD can be generalized over unseen pages from the same Web data source.

The availability of systems that are able to automaticcaly generate wrappers opens interesting perspectives. However, automatically building wrappers for a large number of web site poses several issues. A first problem, which significantly affects the scalability of the approaches based on wrappers (as argued in (Crescenzi et at 2001b) and in (Arasu & Garcia-Molina 2003), is how to collect the sample pages to feed the wrapper generation system. This corresponds to identify clusters of structurally homogeneous sample pages; presently, sample pages are chosen manually. Also, once a library of wrappers for a web site have been generated, one has to choose which wrapper to apply over a target page. Finally, there is the need of a mechanism to navigate the site in order to reach target classes of pages; in other words, navigational paths among the discovered clusters of pages have to be described as well.

In the literature, there are sevelar works that are related to these. Although some of the ideas developed in these works can help us, none of them is suitable for our goals, as discussed in the following.

Web site modeling for data extraction purposes
The issue of modeling the logical structure of web sites for extraction purposes has been studied in several research projects. A pioneering approach is that proposed in the Araneus project (Atzeni et al. 1997, Atzeni et al 2002), where a web page is considered as an object with an identifier (the URL) and a set of attributes. The notion of page scheme is then introduced to model sets of homogeneous pages. Attributes of a page may have simple or complex type. Simple attributes correspond essentially to text, images or links to other pages. Complex attributes model possibly nested collections (lists) of objects. A site scheme is then a collection of page-schemes. To extract data from web pages, a wrapper is associated to each page scheme. Wrappers in Araneus are built either by means of a procedural language, called Cut and Paste (Mecca & Atzeni 1999), or adopting Minerva (Crescenzi & Mecca 1999), a formalism and a tool for rapidly writing wrappers in a more declarative fashion. The notion of site scheme is effective for wrapping purposes; page-schemes describe classes of pages sharing the same structure, and the link attributes connecting them describe the paths to reach instances. However, in Araneus site schemes were designed manually, according to a reverse engineering approach.

A more recent contribution is that of the Wiccap project (Liu et al 2002). In Wiccap, web data are mapped onto a hierarchical logical structure. The focus is on the usability of the model: the goal is to map information from a web site into a logical organization of concepts, as they would be perceived by ordinary users. Nodes of the target structure can then contain data extracted from several pages, integrated to compose a uniform concept. Nodes are associated with mapping rules, i.e. primitives that extract data from the physical structure and map them onto the target model. To ease the burden of creating the hierarchical view over a target web site, a suite of visual tools has been developed (Liu et al 2002).

Classification of web documents based on their structure
The issue of classifying HTML web pages according to their structure has been recently addressed in (Crescenzi et al 2002, Flesca et al 2002). Both the approaches developed in (Crescenzi et al 2002) and (Flesca et al 2002) take as input a large set of HTML (or XML) pages, and create clusters of pages based on properties related to the frequency and the distribution of tags. A drawback of these approches is that they do not address the issue of crawling the target web-site. It is assumed that the clustering is done on the whole site

Bertino et al. have proposed an approach to classify XML documents (Bertino et al 2004). They develop a matching algorithm for measuring thestructural similarity between an XML document and a given DTD. The algorithm compares the structure on the document against those on the DTD. Commonalities and differences are then evaluated tocompute a numerical rank that represents the structuralsimilarity. The matching algorithm is then exploited for the classification of XML documents against a set of DTDs.

Web site structure inference
Liu et al. have developed an algorithm, called SEW, for discovering the skeleton of a target website (Liu et al 2004). The skeleton

here refers to the hypertextual structurethroughout the delivered contents are organized. It is assumedthat a skeleton is a hierarchical structure whose nodes are either content pages or navigation pages. The former arethose pages providing information contents; the latter are pagescontaining links to content pages. The SEW algorithm aims ataautomatically discovering the hierarchical organization ofnavigational and content pages. It relies on a combination ofseveral domain independent heuristics to identify the mostimportant set of links within each page. Compared to our goals, SEW only distinguishes two predefined classes of pages,navigational vs. content pages, whereas we aim at classifyingpages according to their structure, without any a priori assumption about the number of classes and the exposed features.

A similar problem has been studied also by Kao et al (Kao et al 2004), who have developed a technique for analyzing news web sites. The goal of their proposal is to identify, within a news web site, pages of indexes to news, and pages containingnews. An entropy based analysis is performed to eliminate theredundancy of the hyperlinked structure, thus distilling its complexity. A companion technique, also based on entropy analysis, eliminates redundant information, such as navigational panels and advertisements. Also in this case the approach focuses on a specific typology of web sites (news), and there is an assumption about the classes of pages to be discovered (index pages vs news pages).

2.4.a Riferimenti bibliografici

(Adelberg, 1998) B. Adelberg. "NoDoSE a tool for semi-automatically extracting structured and semistructured data from text documents". In ACM SIGMOD, 1998.

(Angluin 1980) Angluin, D. "Inductive inference of formal languages from positive data". Information and Control, (45):117-135, 1980

(Arasu et Al. 2003) Arasu, A. and Garcia-Molina, H. "Extracting Structured Data from Web Pages ". In ACM SIGMOD, 2003

(Arlotta et al, 2003) Luigi Arlotta ,Valter Crescenzi ,Giansalvatore Mecca , Paolo Merialdo: Automatic annotation of data extracted from large Web sites. WebDB 2003 : 7-12

(Atzeni, Mecca 1997) Mecca, G. and P. Atzeni "Cut and Paste", Journal of Computing and System Sciences, Special issue on PODS'97, 1999.

(Bertino et Al. 2004) Bertino, E, Guerrini, G., and Mesiti, M. "A matching algorithm for measuring the structural similarity between an XML document and a DTD and its applications". Inf. Syst. 29(1): 23-46 (2004)

(Chang et Al. 2001) Chang, C.H., and Liu, S.C. "IEPAD: information extraction based on pattern discovery". WWW 2001: 681-688

(Chidlovskii 2000) Chidlovskii, B. "Wrapper Generation by k-reversible Grammar Induction". Proc. Int. Workshop on Machine Learning and Information Extraction (ECAI'00), 61-72, 2000.

(Crescenzi, Mecca 1998) Crescenzi, V. and Mecca, G. "Grammars have exceptions". Information Systems 23(8): 539-565 (1998).

(Crescenzi, Mecca, Merialdo 2001) Crescenzi, V., Mecca, G. and Merialdo, P. "RoadRunner: Towards Automatic Data Extraction from Large Web Sites" VLDB 2001: 109-118

(Crescenzi et al. 2002) Crescenzi, V., Mecca, G. and Merialdo, P. "RoadRunner: Towards Automatic Data Extraction from Large Web Sites" SIGMOD 2002, Demo Program

(Davulcu et al. 2000) Davulcu, H., Yang, G., Kifer, M. and Ramakrishnan, I. V. "Computational Aspects of Resilient Data Extraction from Semistructured Sources". In PODS 2000.

(Embley et al. 1999) Embley, D. W. and Y., J. Y. S. N. (1999). "Record-boundary discovery in web documents". In ACM SIGMOD International Conf. on Management of Data, pages 467--478.

(Fernau 2000) Fernau, H. "On learning function distinguishable languages". Technical Report WSI-2000-13, Whilhem-Schickard-Institut fur Informatil, 2000.

(Flesca et al. 2002) Flesca, S., Manco, G., Masciari, E., Pontieri, L., Pugliese, A. "Detecting structural similarities between xml documents". In WebDB 2002, pages 55-60, 2002

(Garofalakis et al. 2000) Garofalakis, M., Gionis, A., Rastogi, R., Seshadri, S., and Shim, K. "XTRACT: A system for extracting document type descriptors from XML documents". In ACM SIGMOD 2000.

(Gold 1967) Gold, E. M., "Language Identificatio in the Limit". Information and Control, 10(5):447-474, 1967.

(Gruser et al. 1998) Gruser, J. R., Rashid, L., Vidal, E., and L., B. (1998). "Wrapper generation for web accessible data sources". In CoopIS'98, pages 14-23.

(Hammer et al. 1997) Hammer, J., Garcia-Molina, H., Cho, J., Aranha, R., and Crespo, A. (1997). "Extracting semistructured information from the Web". In Proceedings of the Workshop on the Management of Semistructured Data (in conjunction with ACM SIGMOD 1997)

(Kao et al. 2004) Kao, H.Y., Lin, S.H., Ho, J.M., and Chen, M.S. "Mining web informative structures and contents based on entropy

analysis". IEEE Transc. on Knowledge and Data Engineering, 16(1): 41-44, January 2004.

(Kushmerick et al. 1997) Kushmerick, N., Weld, D. S., and Doorenbos, R. (1997). "Wrapper induction for information extraction". In IJCAI'97

(Kushmerick 2000) "Wrapper induction: Efficiency and expressiveness". Artificial Intelligence, 118:15-68

(Liu et al. 2002) Liu, Z., Li, F. and Ng, W.K. "Wiccap Data Model: Mapping Physical Websites to Logical Views". In ER 2002.

(Liu et al. 2004) Lie, Z., Ng W.K, and Lim, E.P. "An automated algorithm for extracting website skeleton". In DASFAA 2004, 799-811

(Muslea et al. 1999) Muslea, I., Minton, S., and Knoblock, C. A. (1999). "A hierarchical approach to wrapper induction". In Proceedings of the Third Annual Conference on Autonomous Agents, pages 190--197.

(Russel Norvig 1994) "Artificial Intelligence: A Modern Approach". Prenctice Hall, 1994.

(Sahuguet, Azavant, 1999) Sahuguet, A. and Azavant, F. (1999). "Web ecology: Recycling HTML pages as XML documents using W4F". In Proceedings of the Second Workshop on the Web and Databases (WebDB'99)

(Soderland 1997) Soderland, S. "Learning to extract text-based information from the World Wide Web". In KDD'97, pages 251--254.

(Wang et Al. 2003) Wang, J. and Lochovsky, F. H. "LData extraction and label assignment for web databases." In WWW 2003: 187-196

2.5 Descrizione del programma e dei compiti dell'Unità di Ricerca

Testo italiano
L'unita' di Roma Tre e' impegnata nel progetto prevalentemente nelle attivita' del Tema 1 (CREAZIONE ED ESTENSIONE DI UNA ONTOLOGIA DI DOMINIO); ma partecipa anche alle attivita' del Tema 2 (SEMANTICA EMERGENTE: SCOPERTA DI MAPPING SEMANTICI TRA ONTOLOGIE).

L'unita' ha approfondite competenze nelle problematiche relative alla estrazione di dati da siti web data-intensive. In particolare, l'unita' ha sviluppato un sistema, chiamato RoadRunner (Crescenzi et al 2001, Crescenzi et al. 2002), per la generazione automatica di web wrapper per sorgenti web data-intensive (vedi Base di partenza scientifica). Dato un insieme di pagine campione simili nella struttura, il sistema genera un wrapper. Il wrapper può essere successivamente essere applicato per estrarre dati da pagine che abbiano la stessa struttura delle pagine campione. Numerosi esperimenti hanno dimostrato l'efficacia e l'efficienza del approccio.

L'esperienza maturata nello sviluppo del sistema RoadRunner (cosi' come lo stesso sistema) costituisce la base di partenza per le attivita' dell'unita', i cui contributi principali saranno nel sottotema 1.2 (Aggiunta di una nuova sorgente informativa alla Ontologia di dominio) in collaborazione con l'unita' di Modena. Come discusso nel Modello A, per estensione di una ontologia di dominio si intende l'aggiunta di una nuova sorgente informativa. Nel caso di una sorgente data-intensive questa operazione comporta: (i) inferenza di uno schema che descriva l'organizzazione dei dati offerti dalla sorgente, (ii) la definizione di programmi, wrapper, che permettano di estrarre i dati dalla sorgente e organizzarli in un formato strutturato, (iii) la attribuzione di una semantica allo schema e ai dati estratti, e (iv) la estensione della Vista Globale Virtuale.

La tecniche sviluppate per la generazione automatica di wrapper risolvono parzialmente e limitatamente i punti precendenti. Con riferimento ai primi due punti, gli approcci in letteratura sono infatti in grado di inferire uno schema, e il programma di estrazione ad esso associato, per un insieme di pagine strutturalmente omogenee. I moderni siti web organizzano le proprie pagine in numerose classi (di pagine simili), in una struttura ipertestuale complessa e articolata. Per generare i wrapper necessari ad estrarre dati da un intero sito (o da piu' di un sito) e' quindi necessario descrivere l'organizzazione delle pagine del sito. Ad oggi tale descrizione puo' essere fatta solo fatta manualmente, limitando severamente la scalabilita' dell'intero approccio.

Per illustrare il problema ci avvaliamo di un esempio. Consideriamo il sito dei campionati del mondo di calcio. Il sito offre alcune miglaia di pagine, fortemente interconnesse. Queste pagine presentano informazioni sulle squadre, le partite, i giocatori, etc. Il sito e' organizzato in maniera regolare; ad esempio abbiamo una pagina per ogni giocatore, una pagina per ogni squadra, una pagina per ogni incontro, e cosi' via. Anche le pagine, al loro interno sono fortemente strutturate: tutte le pagine dei calciatori hanno un template comune, atraverso il quale vengono presentate le stesse informazioni intensionali (il nome, il ruolo, un breve biografia, etc.). Analogamente, tutte le pagine delle squadre presentano le stesse informazioni intensionli in un template comune. Inoltre, una regolarita' interna alle pagine, e interna al sito e' espressa anche dai collegamenti ipertestuali. Ad esempio, tutte le pagine dei calciatori hanno un link alla squadra in cui militano, un insieme di link alle pagine degli altri membri della squadra, etc.

Per poter estrarre informazioni da questo sito sarebbe necessario generare un wrapper per ogni classe di pagina (uno per le pagine dei calciatri, uno per le pagine delle squadre, etc). Quindi, per applicare i wrapper al fine di estrarre continuativamente i dati, e' necessario avere una descrizione delle associazioni ipertestuali tra le varie classi di pagine. Infatti, osserviamo che l'estensione di ogni classe di pagina puo' cambiare nel tempo. Ad esempio, durante lo svolgimento dei mondiali, ogni giorno verranno aggiunte pagine relative alla descrizione degli incontri. Solo avendo a disposizione i percorsi necessari a raggiungere l'estensioni delle varie classi di pagine e' possibile raggiunge le istanze.

Per poter quindi estrarre dati da siti data intesive quali quello del nostro esempio, e' necessario generare una descrizione della struttura del sito (o schema del sito). In questa descrizione devono essere evidenziate le classi di pagine offerte dal sito, e i

collegamenti tra esse.

L'obiettivo principale dello studio e' quello di definire generare automaticamente la descrizione di un sito. Un requisito importante e' l'efficienza di queste tecniche. Nel nostro contesto possiamo dire che l'efficienza corrisponde a visitare una porzione limitata del sito. In pratica si vuole generare un descrizione delle calssi di pagine e delle loro interconnessioni, visitando un piccolo, ma significativo, numero di pagine.

E' importante notare che avere a disposizione uno schema del sito permette di affrontare anche il problema della attribuzione di semantica alle informazioni estratte (corrispondente al punto (iii) di cui sopra). La descrizione a cui abbiamo fatto riferimento precedentemente mira ad individuare classi di pagine. Le istanze di queste classi sono pagine che hanno in comune una struttura. E' ragionevaole attendersi che le pagine di ogni classe offrano le stesse informazioni intensionali; e che i collegamenti tra le classi rappresentino associazioni concettuali. Con riferimento al nostro esempio la classe di pagine dei calciatori e' associata a quella delle squadre. L'obiettivo e' quindi quello di studiare tecniche che, analizzando i contenuti delle pagine di ciascuna classe e i collegamenti alle altre pagine, associno una semantica alle classi e alle associazioni. La strada che che si intende perseguire e' quella di complementare ed estendere le tecniche per la annotazione degli schemi dei wrapper generati automaticamente studiate dalla unita' (Arlotta et al - 2003) con le tecniche di basate su catene lessicali oggetto di recenti studi della unita' di Modena.

Nel seguito descriviamo l'articolazione del programma di ricerca, e i prodotti previsti per ciascuna delle fasi. I prodotti elencati includono sia quelli per i quali l'Unita' ha una diretta responsabilita', sia quelli frutto di lavoro congiunto con le altre unita'.

PRIMA FASE (6 MESI)
Durante la prima fase l'unita' lavorera' congiuntamente a tutte le altre Unità, alla definizione dell'architettura metodologica e funzionale di riferimento per l'intero progetto (prodotto D0.R1). Inoltre collaborera' con le unita altre unita' alla analisi critica dei linguaggi e degli standard emergenti per le ontologie (prodotto D1.R1)
Prodotti:
D0.R1 Rapporto sull'architettura metodologica e funzionale di riferimento (in collaborazione con Modena e Reggio Emilia - MO, Bologna - BO, Trento - TN)
D1.R1: Analisi Critica dei linguaggi e standard emergenti per le ontologie (in collaborazione con BO,MO,TN)

SECONDA FASE (6 MESI)
Durante la seconda fase l'unita' si concentrera' prevalentemente sulla definizione di tecniche che consentano di inferire automaticamente la descrizione di un sito data-intensive. Le tecniche proposte veranno descritte in un rapporto tecnico (prodotto D1.R5).
Infine si lavorerà, congiuntamente a tutte le altre Unità, alla definizione delle interfacce dei componenti per il prototipo integrato (prodotto D0.R2).
Prodotti:
D0.R2 Specifiche delle interfacce dei componenti del prototipo integrato (in collaborazione con MO, BO, TN)
D1.R5 Definizione di tecniche per inferire automaticamente lo schema di un sito data-intensive

TERZA FASE (12 MESI)
Nella terza fase del progetto verranno sviluppato e sperimentato il prototipo per la inferenza automatica dello schema di un sito data-intensive (prodotto D1.P4). Inoltre, congiuntamente alla unita' di Modena, si studieranno tecniche basate su catene lessicali per associare semantica allo schema di un sito data-intensive. Questo studio congiunto portera' alla produzione di un rapporto tecnico (prodotto D1.R6). Infine si collaborerà con le altre Unità all'integrazione dei prototipi realizzati durante il progetto (prodotto D0.P1).
Prodotti:
D0.P1 Prototipo integrato di sistema (in collaborazione con MO, BO, TN)
D1.R6 Definizione di tecniche per associare semantica allo schema di un sito data-intensive basate su catene lessicali (in collaborazione con MO)
D1.P4 Prototipo per inferire automaticamente lo schema di un sito data-intensive

Testo inglese
The research unit of Roma Tre is mainly involved in the activities of the first Theme of the project; but we also participate the activities of the second Theme.

Our unit has studied issues related to the extraction of data from data-intensive web sites. In particular, we have developed a system, called RoadRunner (Crescenzi et al 2001, Crescenzi et al. 2002) to automatically generate wrappers for pages from data-intensive web sites (see scientific basis). Given a small set of pages similar in structure, the system generates a wrapper. the wrapper can then be applied to extract data from pages that share the same structure as the input pages. Several experiments on real life web sites have demonstrated the effectiveness and the efficiency of the approach.

The experiences we have maturated in developing the roadRunner system (and the system as well) represent the basis of the activities of the research unit. Our main contributions concentrate on the theme 1.2 (Adding a new information source to the domain ontology) in collaboration with the research unit of Modena. As discussed in the Model A of the proposal, the extension of a domain ontology corresponds to the adding of a new information source. In the case of a data-intensive web source, this process involves the following tasks: (i) inferring a schema that describes the organization of data offered by the source, (ii) definition of wrappers to extract the data from the source, (iii) providing semantics to the extracted data and schema, and (iv) extension of the Global Virtual View.

The techniques developed for the automatic generation of wrappers represent a partial and limited solution to the above tasks. With respect to the first two tasks, approaches in the literature can infer a schema, and its associated wrapper, for a set of structurally

homogeneous pages. Modern web sites organize their pages in several classes (each class containing similar pages), in a complex and articulated hypertextual structure. In order to generate wrappers to extract data from a whole site (or from more than one site) we need to understand and describe the organization of pages in the site. Presently, such a description can be depicted only manually, drastically limiting the scalability of the approach.

The following example illustrates the issue. Consider the web site of a sport event of worldwide interest. It contains thousands of pages containing information about teams, players, matches, and news. The site content is organized in a regular way; for example we may find one page for each player, one page for each team, and so on. These pages are themselves well-structured. For instance all the player pages share the same structure and, at the intensional level, they present similar information (the name of the player, his current club, a short biography, etc.). Similarly, all team pages share a common structure and a common intensional information, which are different from those of the players. Also, pages contain links to one another, in order to provide effective navigation paths that reflect semantic relationships; for example, every team page contains links to the pages of its players.

In order to extract data from this site, we need to generate a wrapper for each class of pages (one for player pages, one for team pages, etc.). Then once the wrappers are generated, in order to continuously extract data from the site, we need a description of the hypertext paths connecting the various classes of pages. Observe that the extension of the extension of every class of pages may evolve. Continuing with our example, every day one or more new match pages can be added to the site. Only if we have the paths that lead to the extensions of classes of pages offered by the site, we can reach the instances.

Then, in order to extract data from data-intensive web site we have to generate a description (a schema) of the site structure. Such a description should emphasize the classes of pages offered by the site add the hypertextual connections among them.

The main goal of our study is to define and develop techniques to automatically generate the description of a data-intensive web site. An important requirement is the efficiency of the proposed technique. In this context, the efficiency is related to the number of pages to visit in order to generate the description. We aim at inferring a schema for the site exploring a small yet representative portion of its pages.

It is worth observing that reasoning on the site schema it is possible to address the issue of associating semantics to the extracted data as well (task (iii)). The site schema describes classes of pages with similar structure. It is likely that pages in the same class carry the same intensional information, and that links among classes represent conceptual associations. Consider again our running example, the class of player pages is connected to the class of team pages. We aim at studying techniques to associate semantics to classes and associations by analyzing the contents of pages of each class and the links to pages of other classes. Our unit has proposed techniques for annotating the schemas associated with automatically generated wrappers (Arlotta et al - 2003). The direction we will follow is complementing and extending these techniques with the recent studies about lexical chains proposed by the research unit of Modena.

We now describe the various phases in which the project will be divided.

FIRST PHASE (6 MONTHS)
During the first phase the research unit will work with all the other units involved in the project in order to define the methodological and functional architecture for the whole project (deliverable D0.R1). Also we will collaborate with the other research units to develop a critical analysis of the emerging standards and languages for for ontologies (deliverable D1.R1)
Deliverables:
D0.R1 Technical Report describing the methodological and functional architecture of the project (in collaboration with Modena e Reggio Emilia - MO, Bologna - BO, Trento - TN)
D1.R1: Technical Report describing a critical analysis of ontology languages and standards (in collaboration with BO,MO,TN)

SECOND PHASE (6 MONTHS)
During the second phase the research unit will concentrate on the development of techniques to automatically infer the schema of a data-intensive web site. The proposed techniques will described in a technical report (deliverable D1.R5). In addition, the unit will work together with the other units to the definition of the interfaces of the components for the integrated prototype (deliverable D0.R2).
Deliverables:
D0.R2 Definitions of the interfaces of the components of the integrated prototype (in collaboration with MO, BO, TN)
D1.R5 Technical Report describing techniques to automatically infer the schema of a data-intensive web site

THIRD PHASE (12 MONTHS)
During the third phase of the project the research unit will develop and experiment the prototype for automatically inferring a schema of a data-intensive web site. (deliverable D1.P4). Also, the research unit, together with the unit of Modena, will develop techniques for associating semantics to the schema a web site (deliverable D1.R6). Finally, the unit will collaborate with the other units at the integration of the prototypes developed in the project (deliverable D0.P1).
Deliverables:
D0.P1 Integrated system prototype (in collaboration with MO, BO, TN)
D1.R6 Technical Report describing techniques for associating semantics to the schema a web site (in collaboration with MO)
D1.P4 Prototype for the automatic inference of the schema of a data-intensive web site

2.6 Descrizione delle attrezzature già disponibili ed utilizzabili per la ricerca proposta con valore patrimoniale superiore a 25.000 Euro

Testo italiano

Nessuna

Testo inglese

Nessuna

2.7 Descrizione delle Grandi attrezzature da acquisire (GA)

Testo italiano

Nessuna

Testo inglese

Nessuna

2.8 Mesi uomo complessivi dedicati al programma

		Numero	Mesi uomo 1° anno	Mesi uomo 2° anno	Totale mesi uomo
Personale universitario dell'Università sede dell'Unità di Ricerca		*4*	*15*	*15*	*30*
Personale universitario di altre Università		*0*	*0*	*0*	*0*
Titolari di assegni di ricerca		*0*			
Titolari di borse	*Dottorato*	*0*			
	Post-dottorato	*0*			
	Scuola di Specializzazione	*0*			
Personale a contratto	*Assegnisti*	*0*			
	Borsisti	*2*	*9*	*19*	*28*
	Dottorandi	*0*			
	Altre tipologie	*0*			
Personale extrauniversitario		*0*			
TOTALE		**6**	**24**	**34**	**58**

3.1 Costo complessivo del Programma dell'Unità di Ricerca

Testo italiano

Voce di spesa	Spesa in Euro	Descrizione
Materiale inventariabile	*14.000*	*2 Laptop (fascia alta), 1 Stampante, 2 Desktop (fascia alta)*
Grandi Attrezzature		
Materiale di consumo e funzionamento	*2.500*	*cancelleria, CD, DVD*
Spese per calcolo ed elaborazione dati		
Personale a contratto	*40.500*	*sviluppatori software per la progettazione e l'implementazione dei prototipi*
Servizi esterni	*2.500*	*fotocopie, poster, traduzioni*
Missioni	*20.000*	*Incontri di coordinamento con le altre unita', spese di missione per la partecipazione a convegni*
Pubblicazioni		
Partecipazione / Organizzazione convegni	*4.000*	*Iscrizione a convegni internazionali e nazionali*
Altro		
TOTALE	**83.500**	

Testo inglese

Voce di spesa	Spesa in Euro	Descrizione
Materiale inventariabile	*14.000*	*2 Laptops (top level), 1 Printer, 2 Desktops (top level)*
Grandi Attrezzature		
Materiale di consumo e funzionamento	*2.500*	*chancellery, CD, DVD*
Spese per calcolo ed elaborazione dati		
Personale a contratto	*40.500*	*software developers involved in the project for designing and implementing the prototypes*
Servizi esterni	*2.500*	*copies, posters, translations*
Missioni	*20.000*	*Meetings with the other research units, participation to conferences mission costs*
Pubblicazioni		
Partecipazione / Organizzazione convegni	*4.000*	*International conferences fees*
Altro		
TOTALE	**83.500**	

3.2 Costo complessivo del Programma di Ricerca

		Descrizione
Costo complessivo del Programma dell'Unità di Ricerca	*83.500*	
Fondi disponibili (RD)	*25.100*	*utili contratti conto terzi (responsabile prof. Paolo Atzeni)*
Fondi acquisibili (RA)		
Cofinanziamento di altre amministrazioni		
Cofinanziamento richiesto al MIUR	*58.400*	

3.3.1 Certifico la dichiarata disponibilità e l'utilizzabilità dei fondi di Ateneo (RD e RA)

SI

(per la copia da depositare presso l'Ateneo e per l'assenso alla diffusione via Internet delle informazioni riguardanti i programmi finanziati e la loro elaborazione necessaria alle valutazioni; legge del 31.12.96 n° 675 sulla "Tutela dei dati personali")

Firma _____ Data 20/03/2004 ore 14:34