Understanding the Hidden Web

Pierre Senellart

Max-Planck-Institut für Informatik, 22 August 2007
The Hidden Web

Definition (Hidden Web, Deep Web, Invisible Web)


Size estimate (2001) : 500 times larger than the surface Web.

How to understand it and benefit from its content?
The Hidden Web

Definition (Hidden Web, Deep Web, Invisible Web)


Size estimate (2001) : 500 times larger than the surface Web.

How to understand it and benefit from its content?
The Hidden Web

**Definition (Hidden Web, Deep Web, Invisible Web)**


Size estimate (2001): 500 times larger than the surface Web.

How to understand it and benefit from its content?
Understanding the Hidden Web

Purpose

- **Intensional** indexing of the Hidden Web.
- **High-level** queries.
- ⇒ a **semantic** search engine over the Hidden Web.

In a fully automatic, unsupervised, way!

- **Difficult and broad** problem.
- Use of **domain knowledge** (ontology, instances).
- Example of the database publication domain.
Understanding the Hidden Web

Purpose

- **Intensional** indexing of the Hidden Web.
- **High-level** queries.
- \( \Rightarrow \) a **semantic** search engine over the Hidden Web.

In a fully automatic, unsupervised, way!

- **Difficult and broad** problem.
- Use of **domain knowledge (ontology, instances)**.
- Example of the database publication domain.
Purpose

- Intensional indexing of the Hidden Web.
- High-level queries.
- ⇒ a semantic search engine over the Hidden Web.

In a fully automatic, unsupervised, way!

- Difficult and broad problem.
- Use of domain knowledge (ontology, instances).
- Example of the database publication domain.
Understanding the Hidden Web

**Purpose**

- **Intensional** indexing of the Hidden Web.
- **High-level** queries.
- ⇒ a **semantic** search engine over the Hidden Web.

In a fully automatic, unsupervised, way!

- **Difficult** and **broad** problem.
- Use of **domain knowledge** (ontology, instances).
- Example of the database publication domain.
Purpose

- **Intensional** indexing of the Hidden Web.
- **High-level** queries.
- ⇒ a **semantic** search engine over the Hidden Web.

In a fully automatic, unsupervised, way!

- **Difficult and broad** problem.
- Use of **domain knowledge** (ontology, instances).
- Example of the database publication domain.
Web Service Semantic Interpretation Process

WWW
Web Service Semantic Interpretation Process

1. WWW
   - discovery

2. Web service
   - discovery
   - Analyzed

3. HTML form
   - Analyzed
   - Service index
   - User discovery
   - Probing
   - Wrapper induction
   - Semantic analysis
   - Query results
   - Prob-Trees
   - Probing
   - Wrappers
   - Schema Mappings
   - Sem. Model
   - Conclusion
Web Service Semantic Interpretation Process

WWW

discovery

Web service

discovery

HTML form

probing

Analyzed form
+ result pages
Web Service Semantic Interpretation Process

WWW → discovery → HTML form

Web service → discovery → Analyzed form + result pages

discovery → probing

wrapper induction
Web Service Semantic Interpretation Process

WWW \rightarrow HTML form

Web service \rightarrow Analyzed form + result pages

discovery \rightarrow probing

wrapper induction

semantic analysis

P. Senellart (INRIA & U. Paris-Sud)  Understanding the Hidden Web  MPI-Inf., 2007/08/22
Web Service Semantic Interpretation Process

WWW → discovery → HTML form

discovery → Web service → wrapper

Web service → wrapper induction → Analyzed form + result pages

Semantic analysis → Analyzed Web service → indexing → Service index
Web Service Semantic Interpretation Process

WWW → discovery → HTML form

Web service → discovery → Analyzed form + result pages

Web service → wrapper → Analyzed Web service

Analyzed Web service → indexing → Service index

User: query → results
Imprecise Data and Imprecise Tasks

Observations

- Many needed tasks generate *imprecise* data, with some *confidence* value.
- Need for a way to manage this imprecision, to work with it throughout an entire complex process.
Imprecise Data and Imprecise Tasks

Observations

- Many needed tasks generate imprecise data, with some confidence value.
- Need for a way to manage this imprecision, to work with it throughout an entire complex process.
A Probabilistic XML Warehouse

Module 1

Module 2

Module 3

Update transaction + confidence

Query

Results + confidence

Update interface

Query interface

Probabilistic XML Warehouse
A Probabilistic XML Warehouse (Hidden Web)

Module 1
- Update interface
- Query interface
- Update transaction + confidence
- Query + confidence
- Results + confidence

Module 2
- Topic crawler
- Form analyzer
- Inf. Extractor

Module 3

Probabilistic XML Warehouse
A Probabilistic XML Warehouse (Hidden Web)
A Probabilistic XML Warehouse (Hidden Web)

- Topic crawler
- Form analyzer
- Inf. Extractor

Update transaction + confidence

Form URL?

Update interface
Query interface

Probabilistic XML Warehouse

Update interface
Query interface

Results + confidence
A Probabilistic XML Warehouse (Hidden Web)

Update interface
Query interface

Topic crawler
Form analyzer
Inf. Extractor

Update transaction + confidence
Query + confidence

URLs

Probabilistic XML Warehouse
A Probabilistic XML Warehouse (Hidden Web)

- Topic crawler
- Form analyzer
- Inf. Extractor

Update interface
- Analyzed form + confidence

Query interface
- Results + confidence

Probabilistic XML Warehouse
A Probabilistic XML Warehouse (Hidden Web)

Update transaction + confidence

Form?

Results + confidence

Update interface

Query interface

Probabilistic XML Warehouse
A Probabilistic XML Warehouse (Hidden Web)
A Probabilistic XML Warehouse (Hidden Web)

- Topic crawler
- Form analyzer
- Inf. Extractor

Update interface

Query interface

Person → ISBN + confidence

Query

Results + confidence

Probabilistic XML Warehouse
# Outline

1. Introduction
2. A Probabilistic XML Data Model
3. Probing the Hidden Web
4. Wrapper Induction from Result Pages
5. Deriving Schema Mappings from Database Instances
6. Semantic Model
7. Conclusion
Probabilistic Trees

Framework

- Unordered data trees
- Details: no attributes, no mixed content...

![Diagram of unordered data trees]

Sample space: Set of all such data trees.

Probabilistic tree (prob-tree): Representation of a discrete probability distribution over this sample space.
Probabilistic Trees

Framework
- **Unordered** data trees
- Details: no attributes, no mixed content...

Sample space: Set of all such data trees.

Probabilistic tree (prob-tree): Representation of a discrete probability distribution over this sample space.

Joint work with Serge Abiteboul.
Probabilistic Trees

Framework

- **Unordered** data trees
- Details: no attributes, no mixed content...

Sample space: Set of all such data trees.

Probabilistic tree (prob-tree): Representation of a discrete probability distribution over this sample space.

Joint work with Serge Abiteboul.
Probabilistic Trees

Framework

- **Unordered** data trees
- Details: no attributes, no mixed content...

Sample space: Set of all such data trees.

Probabilistic tree (prob-tree): Representation of a discrete probability distribution over this sample space.
Probabilistic Trees

Framework

- Unordered data trees
- Details: no attributes, no mixed content...

Sample space: Set of all such data trees.

Probabilistic tree (prob-tree): Representation of a discrete probability distribution over this sample space.

Joint work with Serge Abiteboul.

P. Senellart (INRIA & U. Paris-Sud)  Understanding the Hidden Web  MPI-Inf., 2007/08/22  8 / 32
The Prob-Tree Model

- Data tree with **event conditions** (conjunction of probabilistic events or negations of probabilistic events) **assigned to each node**.
- Probabilistic events are **boolean random variables**, assumed to be **independent**, with their own probability distribution.

<table>
<thead>
<tr>
<th>Event</th>
<th>Prob.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(w_1)</td>
<td>0.8</td>
</tr>
<tr>
<td>(w_2)</td>
<td>0.7</td>
</tr>
</tbody>
</table>
Features of the Prob-Tree Model

- Well-defined possible world semantics.
- Full expressive power, reasonable conciseness.
- Possible to apply query and updates directly on prob-trees, in an efficient way.
- Complexity study.
- Implementation available.
Features of the Prob-Tree Model

- Well-defined possible world semantics.
- Full expressive power, reasonable conciseness.
- Possible to apply query and updates directly on prob-trees, in an efficient way.
- Complexity study.
- Implementation available.

Joint work with Serge Abiteboul.
Features of the Prob-Tree Model

- Well-defined possible world semantics.
- Full expressive power, reasonable conciseness.
- Possible to apply query and updates directly on prob-trees, in an efficient way.
- Complexity study.
- Implementation available.
Features of the Prob-Tree Model

- Well-defined possible world semantics.
- Full expressive power, reasonable conciseness.
- Possible to apply query and updates directly on prob-trees, in an efficient way.
- Complexity study.
- Implementation available.
Features of the Prob-Tree Model

- Well-defined possible world semantics.
- Full expressive power, reasonable conciseness.
- Possible to apply query and updates directly on prob-trees, in an efficient way.
- Complexity study.
- Implementation available.
Outline

1. Introduction
2. A Probabilistic XML Data Model
3. Probing the Hidden Web
4. Wrapper Induction from Result Pages
5. Deriving Schema Mappings from Database Instances
6. Semantic Model
7. Conclusion
Analyzing HTML Forms

Analyzing the structure of HTML forms.

Problem

Associate to each relevant form field its corresponding domain concept.
First Step: Structural Analysis

1. Build a **context** for each field:
   - label tag;
   - id and name attributes;
   - text immediately before the field.

2. Remove stop words, stem.

3. Match this context with the concept names, extended with WordNet.

4. Obtain in this way **candidate annotations**.
First Step: Structural Analysis

1. Build a **context** for each field:
   - label tag;
   - id and name attributes;
   - text immediately before the field.

2. Remove **stop words**, stem.

3. Match this context with the concept names, extended with WordNet.

4. Obtain in this way **candidate annotations**.
First Step: Structural Analysis

1. Build a **context** for each field:
   - label tag;
   - id and name attributes;
   - text immediately before the field.

2. Remove **stop words**, **stem**.

3. **Match** this context with the concept names, extended with WordNet.

4. Obtain in this way **candidate annotations**.
First Step: Structural Analysis

1. Build a **context** for each field:
   - label tag;
   - id and name attributes;
   - text immediately before the field.

2. Remove **stop words**, **stem**.

3. **Match** this context with the concept names, extended with WordNet.

4. Obtain in this way **candidate annotations**.
Second Step: Confirm Annotations with Probing

For each field annotated with a concept $c$:

1. **Probe the field with nonsense word to get an error page.**
2. **Probe the field with instances of $c$ (chosen representatively of the frequency distribution of $c$).**
3. Compare pages obtained by probing with the error page (by using clustering along the DOM tree structure of the pages), to distinguish error pages and result pages.
4. **Confirm** the annotation if enough result pages are obtained.

In practice, very good precision and good recall; but some limitations on the kind of forms that can be dealt with.
Second Step: Confirm Annotations with Probing

For each field annotated with a concept \( c \):

1. **Probe** the field with nonsense word to get an error page.

2. **Probe** the field with instances of \( c \) (chosen representatively of the frequency distribution of \( c \)).

3. Compare pages obtained by probing with the error page (by using clustering along the DOM tree structure of the pages), to distinguish error pages and result pages.

4. **Confirm** the annotation if enough result pages are obtained.

In practice, very good precision and good recall; but some limitations on the kind of forms that can be dealt with.
Second Step: Confirm Annotations with Probing

For each field annotated with a concept $c$:

1. Probe the field with nonsense word to get an error page.
2. **Probe** the field with instances of $c$ (chosen representatively of the frequency distribution of $c$).
3. Compare pages obtained by probing with the error page (by using clustering along the DOM tree structure of the pages), to distinguish error pages and result pages.
4. Confirm the annotation if enough result pages are obtained.

In practice, very good precision and good recall; but some limitations on the kind of forms that can be dealt with.
Second Step: Confirm Annotations with Probing

For each field annotated with a concept $c$:

1. **Probing** the field with nonsense word to get an error page.

2. **Probe** the field with instances of $c$ (chosen representatively of the frequency distribution of $c$).

3. Compare pages obtained by probing with the error page (by using clustering along the DOM tree structure of the pages), to distinguish error pages and result pages.

4. **Confirm** the annotation if enough result pages are obtained.

In practice, very good precision and good recall; but some limitations on the kind of forms that can be dealt with.
Second Step: Confirm Annotations with Probing

For each field annotated with a concept $c$:

1. **Probe** the field with nonsense word to get an **error page**.
2. **Probe** the field with instances of $c$ (chosen representatively of the frequency distribution of $c$).
3. Compare pages obtained by probing with the error page (by using clustering along the DOM tree structure of the pages), to distinguish error pages and **result pages**.
4. **Confirm** the annotation if enough result pages are obtained.

In practice, **very good precision** and **good recall**; but some limitations on the kind of forms that can be dealt with.
Outline

1. Introduction
2. A Probabilistic XML Data Model
3. Probing the Hidden Web
4. Wrapper Induction from Result Pages
5. Deriving Schema Mappings from Database Instances
6. Semantic Model
7. Conclusion
Extract data from query-answer Web pages.

Issues

- What part of the Web page contains the answer?
- How to extract structured content?
Joint work with researchers from MOSTRARE (INRIA Futurs).

**Query-answer Web Pages**

**Extract** data from query-answer Web pages.

**Issues**

- **What part** of the Web page contains the answer?
- **How to extract** structured content?
Automatic Wrapper Induction with Domain Knowledge

- **Annotate** pages with knowledge domain (finite automata techniques): Both **imperfect** and **incomplete**.

- Use machine learning to **generalize** the result into a structural extraction wrapper (Conditional Random Fields).

**Joint work with researchers from MOSTRAE (INRIA Futurs).**

---

P. Senellart (INRIA & U. Paris-Sud)  Understanding the Hidden Web  MPI-Inf., 2007/08/22  17 / 32
Automatic Wrapper Induction with Domain Knowledge

- **Annotate** pages with knowledge domain (finite automata techniques): Both **imperfect** and **incomplete**.

  Showing results 1 through 25 (of 94 total) for all:xml

1. cs.LO/0601085 [abs, ps, pdf, other] :
   - Title: A Formal Foundation for DIDs
   - Authors: Riccardo Pucetti, Mickey Weisman
   - Comments: 30 pgs, preliminary version presented at WITS-04 (Workshop on Issues in the Theory of Security) 705-
   - Sub-class: Logic in Computer Science, Cryptography and Security
   - ACM-class: H.27. K.4.4

2. astro-ph/0512493 [abs, pdf] :
   - Title: VOFilter, Bridging Virtual Observatory and Industrial Office Applications
   - Authors: Cheng-bo Cui (1), Markos Dokopoulos (2), Peter Quiring (2), Jiancheng Zhao (1), Francoise Genova (3) ((1)NAO China, (2)ESO, (3) CDS)
   - Comments: Accepted for publication in CHAA (9 pages: 2 figures, 185KB)

3. cs.DS/0512061 [abs, ps, pdf, other] :
   - Title: Matching Subsequences in Trees
   - Authors: Rajeev Rastogi, Bing Li Guertz
   - Sub-class: Data Structures and Algorithms

4. cs.LO/0510025 [abs, ps, pdf, other] :
   - Title: Practical Semantic Analysis of Web Sites and Documents
   - Authors: Pierre Despeyroux (BIBA Rosquereau), BRBIA Sophie Antapolsi
   - Sub-class: Information Retrieval

5. cs.CR/0510013 [abs, pdf] :
   - Title: Safe Data Sharing and Data Dissemination on Smart Devices
   - Authors: Lie Bouquet (BIBA Rosquereau), Brice Premarencq (BIBA Rosquereau), Francois Dang Huy (BIBA Rosquereau, PRISM - UVSQ), Nicolas Dinh (BIBA Rosquereau), Philippe Bouchet (BIBA Rosquereau, PRISM - UVSQ)
   - Sub-class: Cryptography and Security; Databases

- **Use machine learning to generalize** the result into a structural extraction wrapper (Conditional Random Fields).

Joint work with researchers from MOSTARE (INRIA Futurs).

P. Senellart (INRIA & U. Paris-Sud) Understanding the Hidden Web MPI-Inf., 2007/08/22 17 / 32
Outline

1. Introduction
2. A Probabilistic XML Data Model
3. Probing the Hidden Web
4. Wrapper Induction from Result Pages
5. Deriving Schema Mappings from Database Instances
6. Semantic Model
7. Conclusion
Motivation

Analyzing the relations between different sources, or between a source and the domain knowledge.

Problem

Given two database instances $I$ and $J$ with different schemata, what is the optimal description $\Sigma$ of $J$ with respect to $I$ (with $\Sigma$ a finite set of formulæ in some logical language)?

What does optimal implies:

- Conciseness of description.
- Validity of facts predicted by $I$ and $\Sigma$.
- Facts of $J$ explained by $I$ and $\Sigma$.

(Note the asymmetry between $I$ and $J$; context of data exchange where $J$ is computed from $I$ and $\Sigma$).

Joint work with Georg Gottlob (Oxford University).
Motivation

Analyzing the relations between different sources, or between a source and the domain knowledge.

Problem

Given two database instances $I$ and $J$ with different schemata, what is the optimal description $\Sigma$ of $J$ with respect to $I$ (with $\Sigma$ a finite set of formulæ in some logical language)?

What does optimal implies:

- **Conciseness** of description.
- **Validity** of facts predicted by $I$ and $\Sigma$.
- **Facts of $J$ explained** by $I$ and $\Sigma$.

(Note the asymmetry between $I$ and $J$; context of data exchange where $J$ is computed from $I$ and $\Sigma$).
Motivation

Analyzing the relations between different sources, or between a source and the domain knowledge.

Problem

Given two database instances $I$ and $J$ with different schemata, what is the optimal description $\Sigma$ of $J$ with respect to $I$ (with $\Sigma$ a finite set of formulæ in some logical language)?

What does optimal implies:

- **Conciseness** of description.
- **Validity** of facts predicted by $I$ and $\Sigma$.
- Facts of $J$ **explained** by $I$ and $\Sigma$.

(Note the asymmetry between $I$ and $J$; context of data exchange where $J$ is computed from $I$ and $\Sigma$).

Joint work with Georg Gottlob (Oxford University).
### Example (Tuple-Generating Dependencies)

<table>
<thead>
<tr>
<th>$R$</th>
<th>$R'$</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>a</td>
</tr>
<tr>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>c</td>
<td>a</td>
</tr>
<tr>
<td>d</td>
<td>d</td>
</tr>
<tr>
<td>g h</td>
<td></td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\Sigma_0 &= \emptyset \\
\Sigma_1 &= \{ \forall x \ R(x) \rightarrow R'(x, x) \} \\
\Sigma_2 &= \{ \forall x \ R(x) \rightarrow \exists y \ R'(x, y) \} \\
\Sigma_3 &= \{ \forall x \forall y \ R(x) \land R(y) \rightarrow R'(x, y) \} \\
\Sigma_4 &= \{ \exists x \exists y \ R'(x, y) \}
\end{align*}
\]
Description based on the **minimum length** of a repair of a formula that is valid and explains all facts of $J$.

This optimality notion gives “intuitive” results for instances derived from each other with simple operations.

Detailed **complexity analysis** for various languages and decision problems. Quite high in the polynomial hierarchy (up to $\Pi_4^P$ for general tgds!).

Even for $\forall x_1 \forall x_2 \forall x_3 \ R(x_1, x_2, x_3) \rightarrow R'(x_1)$, computing the size of the minimal perfect repair is already **NP-complete**.
Results

- Description based on the **minimum length** of a repair of a formula that is valid and explains all facts of $J$.

- This optimality notion gives “**intuitive**” results for instances derived from each other with simple operations.

- Detailed **complexity analysis** for various languages and decision problems. Quite high in the polynomial hierarchy (up to $\Pi_4^P$ for general tgds!).

- Even for $\forall x_1\forall x_2\forall x_3 \ R(x_1, x_2, x_3) \rightarrow R'(x_1)$, computing the size of the minimal perfect repair is already **NP-complete**.
Results

- Description based on the **minimum length** of a repair of a formula that is valid and explains all facts of \( J \).

- This optimality notion gives “intuitive” results for instances derived from each other with simple operations.

- Detailed **complexity analysis** for various languages and decision problems. Quite high in the polynomial hierarchy (up to \( \Pi_4^P \) for general tgds!).

- Even for \( \forall x_1 \forall x_2 \forall x_3 \ R(x_1, x_2, x_3) \rightarrow R'(x_1) \), computing the size of the minimal perfect repair is already **NP-complete**.
Results

- Description based on the minimum length of a repair of a formula that is valid and explains all facts of $J$.
- This optimality notion gives “intuitive” results for instances derived from each other with simple operations.
- Detailed complexity analysis for various languages and decision problems. Quite high in the polynomial hierarchy (up to $\Pi^P_4$ for general tgds!).
- Even for $\forall x_1 \forall x_2 \forall x_3 \ R(x_1, x_2, x_3) \rightarrow R'(x_1)$, computing the size of the minimal perfect repair is already $\text{NP}$-complete.
Outline

1. Introduction

2. A Probabilistic XML Data Model

3. Probing the Hidden Web

4. Wrapper Induction from Result Pages

5. Deriving Schema Mappings from Database Instances

6. Semantic Model

7. Conclusion
Conceptual Model

- IsA ontology of concepts (simple DAG)
  
  ![Diagram of conceptual model with concepts and roles](image)

- n-ary typed roles
  - AuthorOf(Publication,Person)
  - HasName(Person,Name)
Conceptual Model

- **IsA ontology of concepts** (simple DAG)

  ![Diagram](image)

  - Person
    - Man
    - Woman
  - Publication
    - Proceedings
    - Article
    - Book

- **n-ary typed roles**
  - AuthorOf(Publication, Person)
  - HasName(Person, Name)
Semantic Representation of a Service

What is a service described by?

- A \( n \)-uple of typed input parameters.
- A complex (= nested) type of its output.
- Semantic relations between inputs and outputs (Datalog-like description).

**Definition (Complex types)**

\( S: \text{set of concepts} \)

\[ T \leftarrow S|<T,\ldots,T>|T* \]
Semantic Representation of a Service

What is a service described by?

- A \( n \)-uple of **typed** input parameters.
- A **complex** (= nested) type of its output.
- Semantic **relations** between inputs and outputs (Datalog-like description).

**Definition (Complex types)**

\[ T \leftarrow S|<T, \ldots, T>|T^* \]
Semantic Representation of a Service

What is a service described by?

- A \( n \)-uple of **typed** input parameters.
- A **complex** (= nested) type of its output.
- Semantic **relations** between inputs and outputs (Datalog-like description).

**Definition (Complex types)**

\( S: \) set of concepts

\[ T \leftarrow S|<T, \ldots, T>|T^* \]
Semantic Representation of a Service

What is a service described by?

- A \textit{n}-uple of typed input parameters.
- A \textbf{complex} (= nested) type of its output.
- Semantic \textit{relations} between inputs and outputs \textit{(Datalog-like description)}.

\textbf{Definition (Complex types)}

\[ S: \text{set of concepts} \]

\[ T \leftarrow S|< T, \ldots, T >|T^* \]
Services and Queries

Example

Service giving authors from publication titles

\[ A^* \leftarrow \text{AuthorOf}(A,P), \text{HasTitle}(P,T), \text{Input}(T) \]

Example

Query:

\[ <A,T^*>^* \leftarrow \text{AuthorOf}(A,P), \text{Article}(P), \text{HasTitle}(P,T), \text{KeywordOf}("xml",P) \]
Introduction

Prob-Trees

Probing

Wrappers

Schema Mappings

Sem. Model

Conclusion

Services and Queries

Example

Service giving authors from publication titles

\[ A^* \leftarrow \text{AuthorOf}(A,P), \text{HasTitle}(P,T), \text{Input}(T) \]

Example

Query:

\[ \langle A,T^* \rangle^* \leftarrow \text{AuthorOf}(A,P), \text{Article}(P), \]

\[ \text{HasTitle}(P,T), \text{KeywordOf}(\text{xml},P) \]
Managing Extensional Information

How to represent \textit{extensional} information (i.e. \textit{documents}) in this formalism?

\textbf{Definition}

A document is a service with no input.

Complex types: \textit{natural} representation of a DTD.

(Disjunctions $a \mid b$ simulated by $(a?, b?)$).
Web Service Indexing and Querying

Given a query, represented as an analyzed Web service, how to know which known Web services to query?

Issues

- Subsumption of input/output parameters.
- Missing input parameters.
- Composition of Web Services.
Given a **query**, represented as an analyzed Web service, how to know which known Web services to query?

**Issues**

- **Subsumption** of input/output parameters.
- **Missing** input parameters.
- **Composition** of Web Services.
Web Service Indexing and Querying

Given a query, represented as an analyzed Web service, how to know which known Web services to query?

Issues

- **Subsumption** of input/output parameters.
- **Missing** input parameters.
- **Composition** of Web Services.
Web Service Indexing and Querying

Given a **query**, represented as an analyzed Web service, how to know which known Web services to query?

**Issues**

- **Subsumption** of input/output parameters.
- **Missing** input parameters.
- **Composition** of Web Services.
Differences with Classical Database Querying

Three main differences:

- Information can be queried only through \textit{views} (Local As View).
- \textbf{Nested} types.
- \textbf{Incomplete} information.

Three sources of complexity!

Current direction of work: Using \textit{Magic} sets techniques (for evaluation of Datalog programs) restricted to appropriate \textit{binding patterns}.
Differences with Classical Database Querying

Three main differences:

- Information can be queried only through views (Local As View).
- Nested types.
- Incomplete information.

Three sources of complexity!

Current direction of work: Using Magic sets techniques (for evaluation of Datalog programs) restricted to appropriate binding patterns.
Differences with Classical Database Querying

Three main differences:

- Information can be queried only through *views* (Local As View).
- Nested types.
- Incomplete information.

Three sources of complexity!

Current direction of work: Using Magic sets techniques (for evaluation of Datalog programs) restricted to appropriate binding patterns.
Outline

1 Introduction

2 A Probabilistic XML Data Model

3 Probing the Hidden Web

4 Wrapper Induction from Result Pages

5 Deriving Schema Mappings from Database Instances

6 Semantic Model

7 Conclusion
Web Service Semantic Interpretation Process

WWW → discovery → HTML form → probing

discovery

Web service → wrapper → Analyzed form + result pages

wrapper

induction

semantic analysis

Analyzed Web service → indexing → Service index

indexing

User

query

results
Perspectives

Still a lot to do... In particular:

- Answering queries using views on the semantic model.
- Continue work on automatic wrapper induction, to get a form fully wrapped as a Web service.
- Relation between schema mapping induction and inductive logic programming.
Data Warehousing  Extraction of information from the Web, mailing lists... to build a warehouse of sociological data (with various people).

Graph, Text and Web Mining

- Similarity between nodes in graphs; application to synonym extraction (with Vincent Blondel, from UCL).
- Related nodes in a graph; application to Wikipedia (with Yann Ollivier, from ÉNS Lyon).
- PageRank prediction (with Michalis Vazirgiannis).

Machine Translation  Close relations with SYSTRAN; XML document processing, statistical and rule-based machine translation, multilingual authoring...
Other Works

Data Warehousing Extraction of information from the Web, mailing lists... to build a warehouse of sociological data (with various people).

Graph, Text and Web Mining

- Similarity between nodes in graphs; application to synonym extraction (with Vincent Blondel, from UCL).
- Related nodes in a graph; application to Wikipedia (with Yann Ollivier, from ÉNS Lyon).
- PageRank prediction (with Michalis Vazirgiannis).

Machine Translation Close relations with SYSTRAN; XML document processing, statistical and rule-based machine translation, multilingual authoring...
Other Works

Data Warehousing  Extraction of information from the Web, mailing lists... to build a toolbox of sociological data (with various people).

Graph, Text and Web Mining

- Similarity between nodes in graphs; application to synonym extraction (with Vincent Blondel, from UCL).
- Related nodes in a graph; application to Wikipedia (with Yann Ollivier, from ÉNS Lyon).
- PageRank prediction (with Michalis Vazirgiannis).

Machine Translation  Close relations with SYSTRAN; XML document processing, statistical and rule-based machine translation, multilingual authoring...
Other Works

Data Warehousing  Extraction of information from the Web, mailing lists... to build a warehouse of sociological data (with various people).

Graph, Text and Web Mining

- Similarity between nodes in graphs; application to synonym extraction (with Vincent Blondel, from UCL).
- Related nodes in a graph; application to Wikipedia (with Yann Ollivier, from ÉNS Lyon).
- PageRank prediction (with Michalis Vazirgiannis).

Machine Translation  Close relations with SYSTRAN; XML document processing, statistical and rule-based machine translation, multilingual authoring...
Other Works

Data Warehousing  Extraction of information from the Web, mailing lists... to build a *warehouse* of sociological data (with various people).

Graph, Text and Web Mining

- Similarity between *nodes in graphs*; application to synonym extraction (with Vincent Blondel, from UCL).
- Related nodes in a graph; application to Wikipedia (with Yann Ollivier, from ÉNS Lyon).
- PageRank *prediction* (with Michalis Vazirgiannis).

Machine Translation  Close relations with *SYSTRAN*; XML document processing, statistical and rule-based machine translation, multilingual authoring...