Uncertainty Issues in Automating Process Connecting Web and User

A. Eckhardt, T. Horváth*, D. Maruščák, R. Novotný*, P. Vojtáš

Charles University Prague
* P. J. Safarik University Kosice
Motivation

Users looking for a hotel, notebook, car,… on the web without semantic labels

Automating the process brings uncertainty
Motivation

Human understandable not machine readable

only conjunctive user queries – no individual user preferences – boring click through
Outline

• Motivation
• Uncertainty in web content mining
  - discovering data regions, data records
  - attribute values extraction
• Middleware
• Uncertainty in user preference mining
  - learning attribute preference
  - learning combination function
• Experiments – uncertainty issues detected
• Conclusions
Uncertainty in web content mining

- Crawling web (not here)
- Discovering relevant pages (not here)
- Discovering data regions
- Discovering data records
Uncertainty in web content mining

Search over DOM form of page

Non-contiguous records
Ontology based attribute value extraction
Additional low level extraction ontology

<owl:DatatypeProperty rdf:ID="hasPrice">
  <rdfs:domain rdf:resource="#Hotel"/>
  <p1:maxLength
    rdf:datatype="http://www.w3.org/2001/XMLSchema#string">10</p1:maxLength>
  <p1:pattern
    rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    ($)? \d\{1,10\} \.(\d\{1,3\})</p1:pattern>
  <rdfs:label
    rdf:datatype="http://www.w3.org/2001/XMLSchema#string">PRICE</rdfs:label>
</owl:DatatypeProperty>
Middleware, top-k, user model

- Storing data extracted from web pages
- Supporting top-k queries based on user’s combination of user’s attribute preferences

![Diagram](image-url)

- Price
- Distance
- Number of objects
- Rating
- Price ranges: [0, 50), [50, 100), [100, 150), [150, 200)
- Rating ranges: [0, 30), [30, 60), [60, 90), [90, 120), [120, 150), [150, 180)
Uncertainty in user preference mining

• Detecting attribute preferences – cheap or expensive, close or far…domain dependent
• Combination of attribute preferences helps order incomparable objects – IGAP method

User1_hotel(H) good in degree at least @( f₁(x), f₂(y), …)
IF User1_hotel_price(x) good in degree at least f₁(x) AND
User1_hotel_distance(y) good in degree at least f₂(y)
Implementation, experiments

• Modular implementation which allows additional modules to be incorporated (e.g. querying with preference-based querying)
• Communication between modules is based on the traditional Observer/Listener design pattern
• Middleware system for performing top-k queries over RDF data
• As a Java library, our system can be used either on the server side, for example in a Web service, or on the client side
• General method using B+ trees to simulate arbitrary fuzzy ordering of a domain
Identified uncertainty issues

- identifying HTML nodes with relevant information in the sub-tree,
- tuning similarity measures for discovery of similar tag subtrees,
- identifying single data records in non-contiguous html source,
- extracting attribute values
- learning user’s preferences of particular attributes
- learn the user preference combination function.
Conclusions

• Automating the process of user dependent web search – causes uncertainty
• Identified uncertainty issues in our approach (other approaches may have other uncertainty challenges)
• Whole process, querying, results are uncertain – creating a web service we need UIF – Uncertainty Interchange Format
Conclusions

• Implementation and experiments in different domains

  some domains are “easier” to mine – e.g. notebooks – results are more certain

  some domains are “more difficult” – e.g. hotels – results are more uncertain

• Human training time, learning ontology,…

• Low level extraction ontology

• Future work
Thank you

Questions?