User preference ordering of semantized web data

Peter Vojtáš

Kolokvium KSI MFF UK

Tuesday March 8th, 2011 in S8
Flooded by web - dependent on SE – no feedback

- Flooded by web
- Dependent on SE
- No user feedback
- Semantic web
- SWOT of sem. Web
- Project of web semantic(fic/z)ation = conceptualization = is GAV/LAV possible on the scale of Web (read only)
- (referee) quite ambitious goal, and of foremost importance ... the gap can probably not be bridged otherwise than by automatic semantization ...
Content

• **Motivation**, use cases, examples
• Fagin **model** – ordered RDF (triple) distributed data, user preference, top-k algorithm
• Different **users** – interface, formulation of user preference learning, index for preference order
• **Calculus** for user preference ordering of semantized web data
• **Dimensions** of the problem
• Conclusions, future work, problems, plans
Motivation – e-shop – find best (top-k) for user...

• Classical e-shop
  – Strict criteria on attributes
    • Price ≤100$, HDD ≥ 1T, RAM ≥ 2G
  – Conjunction of criteria
  – Simple ordering by price, name
  – Many or few objects
  – Same answer for every user

• From the point of view of producer, service provider, seller, meta-eshop, buyer, user dependent multicriterial ordering, ...
Motivation – no metadata, URL?, readability, ...

• safe cars and roads
  – From which data, where are those data?
  – Text understanding – key word and/or NLP

• judge and bankruptcy clearing process
  – Example 200M debts, left 10M, judge gets 1M
  – Published by law (no computer readable format prescribed e.g. flash, “bad” pdf) OCR?

• public financing
  – Where and when published

• multimedia – which features model user query

• Can Web2.0, social dimension help?
Software Engineering Methodologies

- Waterfall
  - Problem known
  - Solution known
- XP, Agile, ...
  - Problem known
  - Solution unknown
- Jan Rajlich Incremental changes
- Just a start up idea?
- Eric Ries Lean startup
  - Problem unknown
  - Solution unknown
- A social network
- Web Semantization
Fagin (almost lean startup) idea

• Looking for multimedia clip (one server)
  – loud
  – dominantly red

• Looking for a NY restaurant
  – Italian cuisine, Zagat Review
  – price level
  – Close to, safe neighborhood, ...
  – Parking possibilities

• Garlic project of IBM Almaden Research Centre
Model Fagin – Lotem - Naor

Objects \{R_i : i \leq N\}, m attributes
R has user preference score
\(x_1^R, x_2^R, \ldots, x_m^R \in [0, 1]\)
Data in m ordered lists \(L_1, \ldots, L_m\)
Entry in \(L_i\) has form \((R, x_i^R)\) – RDF
Data access:
- Sequential by score ordering- \(c_S\)
- Direct (by “name of” R) – \(c_R\)
- total price \(s^*c_S + r^*c_R\)
Combination function \(t:\[0,1]^n \rightarrow [0,1]\), monotone, i.e.
\[x_i \leq y_i \text{ implies } t(x_1, \ldots, x_m) \leq t(y_1, \ldots, y_m)\]
e.g. \(t(x_1^R, x_2^R, \ldots, x_5^R) = w_1^*x_1^R + w_2^*x_2^R + \ldots + w_5^*x_5^R\)

Random access Fagin threshold algorithm

1. **Do sorted access in parallel** to each of the m sorted lists $L_i$: As an object $R$ is seen, **do random access** to the other lists to find the grade $x_i^R$ of object $R$ in every list $L_i$

Then **compute** the grade $t(R)$: If this grade is one of the $k$ highest we have seen, then **remember** object $R$ and its grade $t(R)$ (ties are broken arbitrarily, so that only $k$ objects and their grades need to be remembered at any time).
Random access Fagin threshold algorithm

2. For each list \( L_i \); let \( x_i \) be the grade of the last object seen under sorted access. **Define** the threshold value

\[
\tau = t(x_1, \ldots, x_m)
\]

As soon as at least \( k \) objects have been seen whose grade

\[
t(R) \geq \tau
\]

then halt else go to 1.

3. Let \( Y \) be a set containing the \( k \) objects that have been seen with the highest grades. The **output** is then the graded set

\[
\{(R, t(R)) \mid R \in Y\}
\]

ordered by \( t(R) \):

TA is correct and instance optimal

**Theorem.** If the aggregation function \( t \) is monotone, then TA correctly finds the top \( k \) answers.

**Proof.** Assume that \( z \) was not seen, then \( x_i^z \leq x_m \) and hence
\[
   t(x_1^z, \ldots, x_m^z) \leq \tau = t(x_1, \ldots, x_m)
\]
For every \( y \) in \( Y \) we have \( t(y) \geq \tau \) therefore \( t(z) \leq t(y) \).

**Theorem.** Assume that the aggregation function \( t \) is monotone.
Let \( D \) be the class of all databases.
Let \( A \) be the class of all algorithms that correctly find the top \( k \) answers for \( t \) for every database and that do not make wild guesses.

Then TA is instance optimal over \( A \) and \( D \).
Proof of instance optimality of TA

**Proof.** Let $\mathcal{A} \in \mathcal{A}$ a $\mathcal{D} \in \mathcal{D}$. Result is $Y_{\mathcal{A}}$ and in each list $L_i$ algorithm $\mathcal{A}$ made $d_i$ sequential steps, last seen score is $x_i$ a threshold is $\tau_{\mathcal{A}}$.

$\mathcal{A}$ has seen $a$ objects, $\text{price}_{\mathcal{A}} \geq a^* c_S$. Put $d = \max d_i$, we have $d \leq a$.

As $\mathcal{A}$ does not make wild guesses, so $\mathcal{TA}$ has seen after $d^* m$ steps all object seen by $\mathcal{A}$ and $\tau_{\mathcal{TA}} \leq \tau_{\mathcal{A}}$.

Now a crucial property: For $R \in Y_{\mathcal{A}}$ we have $t(R) \geq \tau_{\mathcal{A}}$.

Assume not, create a database $\mathcal{D}'$, such that, instead of $d_i + 1$st object we insert record $(V, x_i)$. As $\mathcal{A}$ does not guess, on $\mathcal{D}'$ runs in same way as on $\mathcal{D}$. Object $V$ was not seen and result is $Y_{\mathcal{A}}$, contradiction.

Proof of instance optimality of TA

As $\tau_{TA} \leq \tau_{\mathcal{A}} \leq t(R)$,

**TA** stops at the latest in step $d$.

**Price of TA** computation is

$$d \cdot m \cdot c_s + d \cdot m \cdot (m-1) \cdot c_R$$

**Ratio of TA and $\mathcal{A}$ prices** (use $d/a \leq 1$) is

$$\frac{dmc_s + dm(m-1)c_R}{ac_s} \leq m + m(m-1) \frac{c_R}{c_S}$$

a constant $\square$

---

Implementations and experiments

In practice can $c_R/c_S$ be $> 1000$, for 5 attributes is $m(m-1) = 20$, so many improvements and heuristics are possible.

Worst case – 2 attributes – needs to visit half of data

3 attributes – $2/3$ ...

Object is seen in step $f$ and in top-k certified in step $c$.

Depending on distribution, (e.g. exponential in collection, R. Lencses) we can have $f < 1\%$ and $c < 10\%$.

Tokaf (A. Eckhardt), P. Gursky, ...
Local preferences $f_i^u : D_{A_i} \rightarrow [0,1]$

Transform the data cube $\Pi D_{A_i}$ into $[0,1]^N$

Global preferences - monotone

Aggregation $t^u : [0,1]^N \rightarrow [0,1]$

User preference =

$t^u(f_1^u(R.A_1), f_2^u(R.A_2), \ldots, f_N^u(R.A_N))$
New user v - from list \( L_i^u \) to list \( L_i^v \) - reorder domains?
Navigate the index! New structure

New user – learn her model from rating (DP-B. Václav)
Learning parameters of Fagin’s model - definition

given $r^u$

find $f^u_i, t^u$

such that

$r^u = \underline{r^u} \mid S$ ? or

$\mid r^u - \underline{r^u} \downarrow S \mid < \varepsilon$ ? or

orders induced by

$r^u = \underline{r^u} \mid S$ coincide?

Do not contradict?

...
Learning parameters of Fagin’s model

Approximation of \( r_u \) by \( \underline{r}_u \)?

Consider \( r_u \) as classification to \( \left| \text{rng}(r_u) \right| \) classes?

Optimization? Learn the model

\[
\underline{r}_u(R) = t_u(f^u_1(R.A_1), f^u_2(R.A_2), \ldots, f^u_N(R.A_N))
\]

ANN

\[
F(x) = \sum_{i=1}^{m} \alpha_i \varphi \left( \sum_{j=1}^{N} w_{ij} x_j + b_i \right)
\]

More Alan Eckhardt
Data calculi for Fagin, Datalog, ... fuzzy logic

• Domain relational calculi! Datalog! ... RDF, all known from classical querying... fuzzify?

• Which logic? Which connectives?

• Lukasiewicz? \( \&_L(x,y) = \max\{0, x+y-1\} \)
  \[ \rightarrow _L(x,y) = \min\{1, 1-x+y\} \]

• Product? \( \&_P(x,y) = x*y \)
  \[ \rightarrow _P(x,y) \approx \min\{1, y/x\} \]

• Goedel? \( \&_G(x,y) = \min\{x, y\} \)
  \[ \rightarrow _G(x,y) \approx y \]
Fuzzy logic? Tautologies?

\((A \to (B \to C)) \to ((A \to B) \to (A \to C))\) is not a tautology of Lukasiewicz logic

\[ \to_L(x, y) = \min\{1, 1-x+y\} \]

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>(B → C)</th>
<th>(A → (B → C))</th>
<th>(A → B)</th>
<th>(A → C)</th>
<th>((A → B) → (A → C))</th>
<th>(A → (B → C)) → ((A → B) → (A → C))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\frac{3}{4})</td>
<td>(\frac{1}{2})</td>
<td>(\frac{1}{4})</td>
<td>(\frac{3}{4})</td>
<td>(1)</td>
<td>(\frac{3}{4})</td>
<td>(\frac{1}{2})</td>
<td>(\frac{3}{4})</td>
<td>(\frac{3}{4})</td>
</tr>
</tbody>
</table>

???
(A → (B → C)) → ((A → B) → (A → C)) is not a tautology of Lukasiewicz logic \( \rightarrow_{\text{L}}(x,y) = \min\{1, 1-x+y\} \)

Pavelka’s graded logic

Hájek’s comparative notion of truth

Datalog does not need tautologies!! No logical axioms.

Relational algebra like Goedel-Bernays set theory, ...

Fuzzy – vague, controller,... **Fuzzy as preference**
Generalize \{0,1\} to \([0,1]\)-Datalog – still challenging

• Should our rules be implications or clauses?
  • \(B \rightarrow H \equiv \lnot B \lor H\)

• should our computation be refutation or query answering?
  • \(\lnot B \lor H, \lnot H \) infer \(\lnot B\), or ?-H, \(H \leftarrow B \ldots \) ?-B

• is unification touched by this or not?
  • Fuzzy similarity? \(\rightarrow\) new type of indexes?
Fuzzy Datalog

• [0,1]-Datalog with implicative rules
• No refutation – querying = fuzzy modus pones
  \[ B \rightarrow H.r,B.b \]
  \[ H.f \rightarrow (b,r) \]
• Computed answer x is correct
• Correct answer y, \( \varepsilon > 0 \), \( \exists \) computed answer \( x_\varepsilon > y - \varepsilon \)
• Continuous semantics, fixpoint (GAP of Kifer Subrahmanian is not continuous)
• similarity axioms are a Datalog program
How to learn fuzzy rules? Fuzzy ILP?

- ILP assumes background knowledge $B$, positive examples $E^+$, negative examples $E^-$, we look for hypothesis $H$ such that
  
  $$B \cup H \models E^+ \text{ and } B \cup H \not\models E^-$$

- In fuzzy case $B$: $B \rightarrow [0, 1]$
  
  $E$: $E \rightarrow [0, 1]$, we look for
  
  $H$: $H \rightarrow [0, 1]$, such that

  E degree matches? order matches? Doesn’t cheat?
Fuzzy as preference – not a mathematical generalization

Computed correct

Monotone
Precision, recall by degree

\[ \tau(L_1, L_2) = \frac{n_c - n_d}{1/2 \times n \times (n - 1)} \]
New problems

• Are $R$ and $r^u$ **monotonizable**? Wrts to data indexes, measures

• Having $R$ and set of users $U$ and rating **monotone**
  – ? $\exists$ **ordering** of domains, local preferences are **simple** (left/right shoulder, peak, valley)?

• Small training data, fast response e.g. $< 0.5$ sec.

• Experiments repeatable, reusable, portable,…

• Improvement from the point of view of
  – Software Engineering, Database, …
  – Human user
GACR project Web Semantization P202/10/0761

Change architecture? No Trisolda – Bobox? Whole .cz?
Where are data from? Where is user model from?

• Semantized web data
  • Known ontology
    - Web information extraction (D. Maruščák)
    - WIE, annotation from texts (J. Dědek)
  • Unknown ontology
    - Ontology (microformats) user assisted (social network), store
      Semantized data (semantization rules) on server - D. Fišer

• Learning user preference (ordering)
  • registered, known basket, collaborative, ...
  • explicit rating (A. Eckhardt)
  • implicit behaviour (L. Peška)
  • filtering (I. Lašek)

what next?
Product Development at Lean Startup

Unit of Progress: Validated Learning About Customers ($$$)

Customer Development

Problem: unknown

Solution: unknown

Hypotheses, Experiments, Insights

Data, Feedback, Insights

User Stories

Architectural Spike

Release Planning

Iteration

Acceptance Tests

Small Releases
• Raise cooperation of universities and companies
  • Building research teams
  • building start-up teams
  • human experiments on prototypes
  • creation of human preference golden standards

ICT-2011.1.5 Networked Media and Search Systems
Call identifier: FP7 - ICT-2011.1.5

• Project proposal participation
  • especially with our social network
professors from Saarbruecken who teach DB or IR and have projects on XML

drama with three women making a prophecy to a British nobleman that he will become king

the woman from Paris whom I met at the PC meeting chaired by Raghu Ramakrishnan

→ “Semantic Search”:
  • exploit structure and annotations in the data
  • exploit background knowledge (ontologies/thesauri + statistics)
  • connect/join/fuse information fragments

XML-IR Example (1)

Which professors from Saarbruecken (SB) are teaching IR and have research projects on XML?

XML-IR Example (2)

Need to combine DB and IR techniques with logics, statistics, AI, ML, NLP for ranked retrieval

// Professor [/* = „Saarbruecken“]
// Course [/* = „IR“]
// Research [/* = „XML“]
Conclusions, future work, problems, plans ...

- Problems, use-cases, models, algorithms, prototypes, benchmarks, experiments and measures, ...
- Top-k for XML (do we need RDF(S), OWL, ...?)
  - Product pages
  - Textual resources
  - Unknown schema, location, machine readability
  - Training (experts, users), automation
  - Domain dependent? Domain independent?
- Or, enforce by law, government, community, third party solution via web information extraction?
Questions, comments,....

Thank you