Finding Unexpected Information

Taken from the paper :

"Discovering Unexpected Information from your Competitor's Web Sites"

by Bing Liu, Yiming Ma, and Philip S. Yu

Presented by Zheyuan Yu

What is ‘Unexpected Information’ ?

- Relevant but unknown
- Contradicts user’s existing beliefs or expectations
- E.g. A company wants to know what it does not know about competitors

Existing Extraction Methods

- Manual Browsing
- Search Engine – user-specified keywords
- Web query – languages (SQL) search through info. resources (XML)
- User preference approach – info. given according to set preference categories

Problems with Existing Methods

- Only information expected by or already known to user is returned
- User cannot search for something he doesn’t know he is looking for
- Manual examination takes too long

Proposed approach

- Aim: Finding interesting/unexpected information
- To find what is unexpected, we need to know what the user has known?
- It becomes a problem of comparing user's website with competitor's website to find similar and different information.

How to represent the page’s information

- Documents and Queries are represented as vectors.
- Position 1 corresponds to term 1, position 2 to term 2, position t to term t

$$D_i = w_{d_{i1}}, w_{d_{i2}}, \ldots, w_{d_{it}}$$

$$Q = w_{q1}, w_{q2,} \ldots, w_{qt}$$

$$w = 0 \text{ if a term is absent}$$

Weight

- tf x idf measure:
  - term frequency (tf)
  - inverse document frequency (idf)

$$tf_{i,j} = \frac{f_{i,j}}{\max_l f_{i,j}}$$

$$idf_{i,j} = \log \frac{N}{n_i}$$

$$Weight : w_{i,j} = tf_{i,j} * idf_{i,j}$$

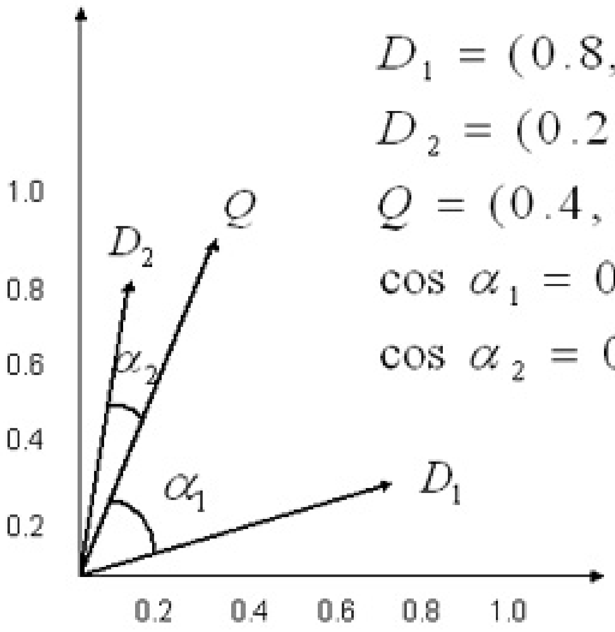
How to calculate the similarity

$$D_i = < w_{d_{i1}}, w_{d_{i2}}, \ldots, w_{d_{it}} >$$

$$Q = < w_{q1}, w_{q2,} \ldots, w_{qt} >$$

$$sim(Q,D) = \frac{\vec{Q} \cdot \vec{D}}{|\vec{Q}| \times |\vec{D}|}$$

$$= \frac{\sum_{j=1}^{t} w_{qj} * w_{d_{ij}}}{\sqrt{\sum_{j=1}^{t} (w_{qj})^2 * \sum_{j=1}^{t} (w_{d_{ij}})^2}}$$



$D_1 = (0.8, 0.3)$

$D_2 = (0.2, 0.7)$

$Q = (0.4, 0.8)$

$\cos \alpha_1 = 0.74$

$\cos \alpha_2 = 0.98$

Compare Two Web Sites (1): Similar Pages

- Goal – find pages in the competitor site that closely match a page in the user site
- Method – given a u_j (user page) in U (user web site), for all c_i (competitor page) in C (competitor web site) compute:

$(u_j$ dot $c_i) / (|u_j|$ cross $|c_i|)$

Then rank pages in descending order

Compare Two Web Sites (2): Unexpected Terms

- Goal – find unexpected terms in a competitor page relative to a user page
- Method – given a u_j in U and a c_i in C, find unexp. term k_r by computing:

$$\mathrm{unexpT}_{rji} = \begin{cases} 1-(tf_{rj} / tf_{ri}), & \text{if } (tf_{rj} / tf_{ri}) <= 1 \\ 0, & \text{otherwise} \end{cases}$$

Then rank the k terms in descending order

Compare Two Web Sites (3): Unexpected Pages

- Goal – find unexpected pages in the competitor site relative to the user site
- Method – combine all the pages in U to form a single document and all the pages in C to form another single document

$$unexp P_i = \frac{\sum_{r=1}^{m} unexp T_{r,c,u}}{m}$$

This is necessary because information on a topic can be contained entirely on one page or spread through many, as web site structures vary

Compare Two Web Sites (4) Unexpected Concepts

- Goal – find unexpected concepts in a competitor page relative to a user page
  - More meaningful than keywords
  - Less information for user to look at
- Method – first use association rule algorithm (Apriori used – next slide) to discover concepts

  Each page is mined separately because concepts tend to be page based

  Unexpected term comparison is then done with concepts in place of keywords

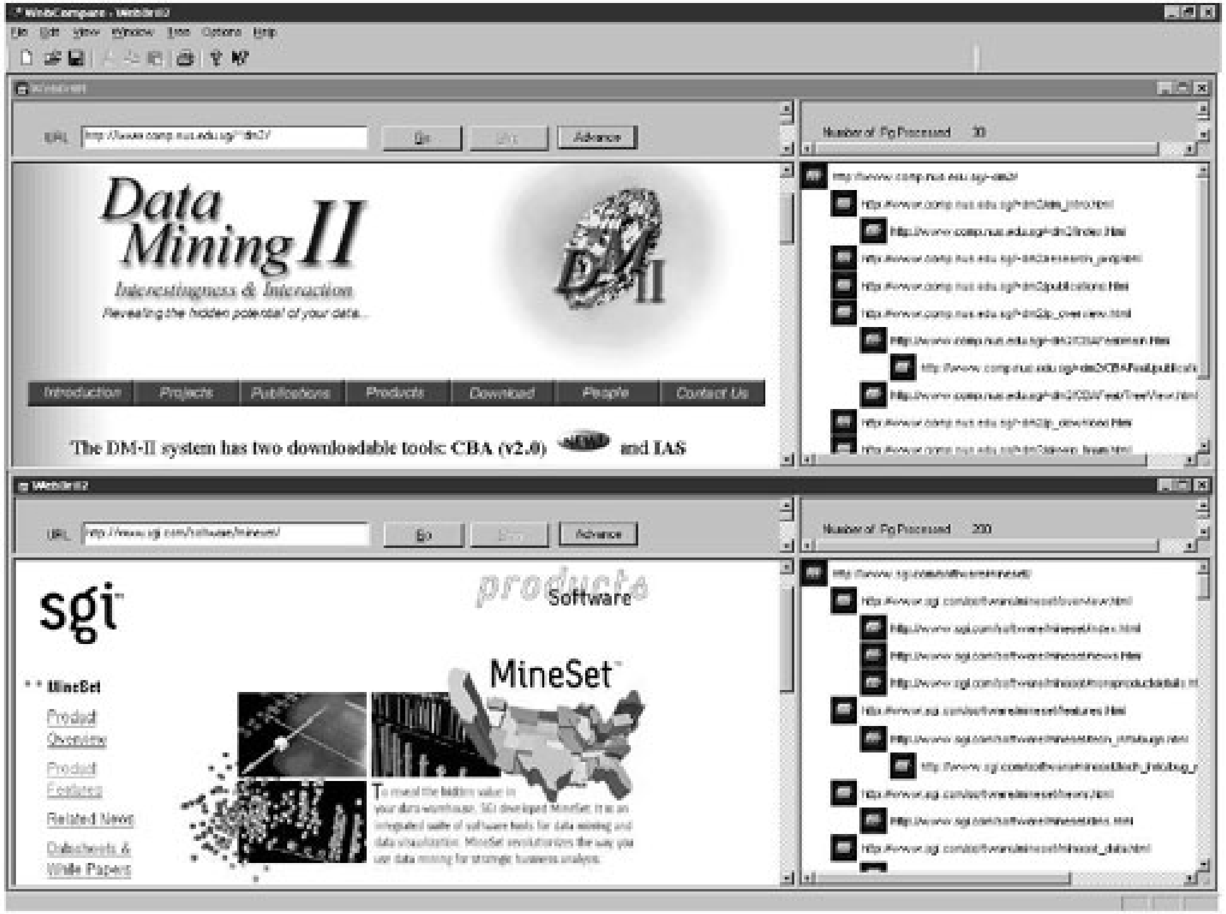
Unexpected Concepts – Apriori Algorithm

- Keywords in each sentence are a transaction
- The set of all sentences is a dataset
- Treat concepts as terms, using method 2 to find unexpected concepts
- Support = count(k_1 U k_2)
- Confidence = count(k_1 U k_2) / count (k_1)
- Candidates pruned based on sup. & con.

Compare Tow Web Sites (5): Outgoing Links

- Goal – Find all outgoing links in C that are not in U
- Method – Links are simply collected by the crawler when it initially explores the U and C sites

System Screenshot



URL: http://www.comp.nus.edu.sg/~dm2/

Number of Pg Processed 30

Data Mining II

Interestingness & Interaction

Revealing the hidden potential of your data...

The DM-II system has two downloadable tools: CBA (v2.0) NEW and IAS

http://www.comp.nus.edu.sg/~dm2/
http://www.comp.nus.edu.sg/~dm2/dm_intro.html
http://www.comp.nus.edu.sg/~dm2/index.html
http://www.comp.nus.edu.sg/~dm2/research_proj.html
http://www.comp.nus.edu.sg/~dm2/publications.html
http://www.comp.nus.edu.sg/~dm2/p_overview.html
http://www.comp.nus.edu.sg/~dm2/[illegible].html
http://www.comp.nus.edu.sg/~dm2/CBA/[illegible]publications
http://www.comp.nus.edu.sg/~dm2/[illegible]/TreeView.html
http://www.comp.nus.edu.sg/~dm2/p_download.html
http://www.comp.nus.edu.sg/~dm2/[illegible].html

URL: http://www.sgi.com/software/mineset/

Number of Pg Processed 200

To reveal the hidden value in your data warehouse, SGI developed MineSet. It is an integrated suite of software tools for data mining and data visualization. MineSet revolutionizes the way you use data mining for strategic business analysis.

http://www.sgi.com/software/mineset/
http://www.sgi.com/software/mineset/overview.html
http://www.sgi.com/software/mineset/index.html
http://www.sgi.com/software/mineset/news.html
http://www.sgi.com/software/mineset/moreproductdetails.html
http://www.sgi.com/software/mineset/features.html
http://www.sgi.com/software/mineset/tech_info/bugs.html
http://www.sgi.com/software/mineset/tech_info/bug_[illegible]
http://www.sgi.com/software/mineset/news.html
http://www.sgi.com/software/mineset/[illegible].html
http://www.sgi.com/software/mineset/mineset_data.html

Summary of Use

- User selects a topic of interest, identifies a page of his own that deals with the topic.
- User then can find pages in a competitor’s site that deal with the same topic, giving the user an idea of the quantity and location of these pages (method 1)
- User can scan these pages for unexpected information (method 2, method3)

Summary of Use (cont'd)

- User can then manually browse similar pages with interesting unexpected information
- User can find unexpected pages based on concepts (method 4)
- User can examine unexpected outgoing links for more information or to add the links to his own pages (method 5)
- Experiments include comparison for travel company, private education institution and diving company. Many piece of unexpected information discovered.

Time Complexity: Linear in the number of pages.

- **‘Web Crawling’**, one-time, is O(N) where n is number/size of pages
- **‘Extraction and Mining’**, one-time, is $O(K^2N)$, where K is number of keywords
- **‘Corresponding Page’** is $O(T_CN_C+N_uN_C)$, where N_C is number/size of pages in C, T_C is maximum amount of terms in any page in C and N_u is size of the page in U (weighting time + similarity computation)
- **‘Unexpected Terms’** is $O(T_c)$, where T_c is the amount of terms in the page in C

Time Complexity (cont'd): Linear in the number of pages.

- **'Unexpected Pages'** is $O(T_U N_U + T_C N_C)$, where T_U is maximum terms in a U page and N_U is number/size of pages in U, and T_C & N_C have similar meanings for C (time for merging - $unexpP_i$ is $T_C N_C$)
- **'Unexpected Concepts'** is $O(Co_c)$, where Co_c is the amount of concepts in the page in C
- **'Unexpected Links'** is $O(L_c)$ where L_c is the amount of links in C
- Assuming size (or # of keywords) on an average page is constant, then all comparison algorithms are basically linear in the number of pages involved

Efficiency

- Experiments run on a PII 350 PC w/ 64MB RAM

Process	Similar Page	Unexp. Terms	Unexp. Pages	Assoc. Mining
Avg. Time (ms)	12.3	17.5	21.1	19.7

- All computations can be done efficiently
- Unexpected pages can be found for a 50 page competitor site in about a second

Future Application

- Research tool to find related topics
- Shopping comparison between 2 sites

Summary

- Unexpected information is interesting
- Proposed a number of methods
- Techniques proposed are practical and efficient

References

- Liu, Bing,Yiming Ma, Philip S. Yu.

  Discovering Unexpected Information from Your Competitor’s Web Sites. *Proceedings of The Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001),* August 26-29, 2001, San Francisco, USA.

Thank you!

Any questions?

☺