

# WebMate : A Personal Agent for Browsing and Searching

Liren Chen and Katia Sycara  
Carnegie Mellon University

## Introduction

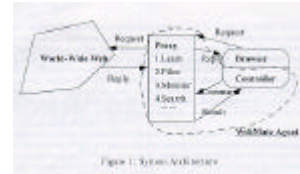
- WebMate : an agent helping users to effectively browse and search the web
  - Art
    - multiple TF-IDF vectors
    - Trigger Pair model for keyword refinement
    - Relevance feedback during the search
  - using these techniques
    - provides effective browsing and searching
    - sends to users personal newspaper

## Features

- Searching enhancement
  - Parallel search, searching keywords refinement, relevant feedback
- Browsing assistant
  - Learning, Recommending, Alias, Monitoring bookmarks, More like, Sending page to friends, Prefetching hiperlinks
- Offline browsing
  - downloading following pages, getting references and printing it out
- Filtering HTTP header, cookie, block animation to speed up
- Checkinghtml page : error finding, dead links,
- Dynamically setting up : search engines, dictionary
- Programming in java

## WebMate architecture

- Learning user interests
  - a personalized newspaper
- Helping the user refine search
- system architecture



- HTTP proxy(monitoring) + applet controller(interface)

## Learning profile to compile personal newspaper

### Profile Representation and Learning Algorithm

- filtering task : judge whether relevant or not based on the user profile
- multiple user interests : single user profile, ask explicitly
  - WebMate learns the categories automatically
  - updates the profile incrementally and continuously
    - other systems do not like this.
  - Using TF-IDF with multiple vectors representation

$$IDF(w) = \log \frac{D}{d_f(w)}$$

### Multi TF-IDF vector learning

- N : domains of interest (category ) - predefined number
- initial profile set V, |V|=0
- M : preset number of elements of a vector
- for each positive example ("I like it")
  1. Preprocess :
    - Parse HTML page, deleting the stop words, stemming the plural noun to single form, giving more weights to title word
  2. Extract the TF-IDF vector for this document  $v_j$
  3. IF  $|V| < N$  ( $|V|$  is the number of vectors in the profile set V), then
    - Add  $v_j$  to V
  4. Otherwise, calculate the cosine similarity between every two TF-IDF vectors including the vectors in the profile set V and the new document vector  $v_j$ . Assume the profile set V is  $\{v_1, v_2, \dots, v_m\}$

$$Sim(v_j, v_k) = \frac{v_j \cdot v_k}{|v_j| \times |v_k|} \quad \{j, k \in \{1, 2, \dots, m, j\}\}$$

- 5. Combine the two vectors  $V_i$  and  $V_m$  with the greatest similarity

$$V_i = V_i + \beta \cdot V_m \quad (i, m) = \arg \max_{(i, m)} \text{Sim}(V_i, V_m) \quad i, m \in \{1, 2, \dots, n, d\}$$

- 6. Sort the weights in the new vector  $V_i$  in decreasing order and keep the highest M elements
- this algorithm is run whenever a user marks a document as "I like it", thus the user profile is incrementally updated

### Compiling personal newspaper

- automatically spide a list of URLs that the user wants monitored
  - parse the html page
  - extract the links of each headline
  - fetches those pages
  - constructs the TF-IDF vector for each of those pages
  - calculate the similarity with the profile
  - If the similarity is greater than some threshold, then recommend

- If the user does not provide any URLs that he would like to be the information sources

- WebMate constructs a query using current profile

### Experiments

Date	Accuracy in top 10	Accuracy in top 20	Accuracy in whole
Sep. 16	78%	60%	17/55=31%
Sep. 17	46%	35%	11/43=26%
Sep. 18	56%	35%	3/43=7%
Sep. 19	64%	65%	16/76=21%
Sep. 20	66%	40%	9/29=31%
Sep. 22	46%	40%	12/46=26%
Sep. 23	56%	30%	18/76=24%
Sep. 24	66%	34%	10/18=44%
Average	57%	44%	10.4%

Table 1: Experiments Results

## Search refinement by keywords expansion and relevance feedback

### Trigger Pairs model to extract relevant words

- single keywords are ambiguous
  - "stock" has more than 10 definition in the WordNet
  - pruning : manual query expansion, semi-manual, automatic
    - manual expansion : user may not be able to provide the best refinement words
    - "Best" - most frequently co-occur with the word in its intended meaning
- "trigger pair" (S, T)
  - If a word S is significantly correlated with another word T, then (S, T) is considered a "trigger pair", with S being the trigger and T the triggered word

- In the trigger Pairs Model (S, T) is different from (T, S), so the Trigger Pairs Model is different from the method of using co-occurrence of two words that is generally used in other keywords expansion

- Mutual information (MI) : considers the words order

$$(s, t) = P(s, t) \log \frac{P(s, t)}{P(s)P(t)}$$

### Broadcast News Corpus

- set the maximum distance between S and T : 500
- Trigger Pairs : sorted in decreasing order
  - car ← {motor, auto, model, maker, vehicle, ford, buick, honda, inventory, assembly, chevrolet, sale, nissan, incen.tif, pontiac, planet, toyota, dealer, chrysler}
  - music ← {musical, symphony, orchestra, composer, song, concert, tune, concerto, sound, musician, classical, album, violin, violinist, jazz, audience, conductor, play, audio, rock, cello, perform, dance}

### Wall Street Journal Corpus

- Trigger Pairs : domain specific

### Keywords Expansion

- Trigger Pair method can provide several candidate refinement keywords

- $S_1 = \{s_{11}, s_{12}, \dots, s_{1n}\} \rightarrow K_1, S_1$  is the triggers set to  $K_1, s_{11}, s_{12}, \dots, s_{1n}$  are sorted in decreasing order of the mutual information.  
 $S_2 = \{s_{21}, s_{22}, \dots, s_{2m}\} \rightarrow K_2, S_2$  is the triggers set to  $K_2$   
 $\dots$   
 $S_n = \{s_{n1}, s_{n2}, \dots, s_{nm}\} \rightarrow K_n, S_n$  is the triggers set to  $K_n$
- $S = S_1 \cup \dots \cup S_n$  and  $\{S_1 \cap S_2, S_1 \cap \dots \cap S_n\}$  and  $\{S_1, S_2, \dots, S_n\}$  is one of the combinations of n sets out of m. The words in the S are sorted in decreasing order of mutual information.
- If  $|S| \geq N$ , let the top N words in the S be the refinement words and stop.
- otherwise, let  $n = n - 1$ , goto 2.

- $K1 = \text{charge}$  and  $S1 = \{\text{federal, investigation, attorney, plead, indict, \dots}\}$
- $K2 = \text{fee}$  and  $S2 = \{\text{pay, dollar, million, bank, service, tax, raise, \dots}\}$
- $K = \{K1, K2\} = \{\text{charge, fee}\}$  and  $S = S_1 \cup S_2 = \{\text{million, pay, dollar, tax, service, federal, client, \dots}\}$ 
  - so triggers, such as million, pay, dollar, tax, service, help confine and disambiguate the meaning of the word "charge"

### Examples : keyword "stock"

From Lycos:

- 1) YOSEMITE STOCK PHOTOS, ROCK CLIMBING PHOTOS, Yosemite PHOTOS
- 2) YOSEMITE STOCK PHOTOS, ROCK CLIMBING PHOTOS
- 3) YOSEMITE STOCK PHOTOS, PHOTOS PHOTOS
- 4) Stock Information from Apple
- 5) STOCK GRAPHICS & PHOTOS
- 6) American Stock Trends & Trend Report Page
- 7) STOCK CHARTS
- 8) GROWER STOCKS, HUNTER TUL, DECLAMER
- 9) Stock Information from Apple
- 10) Delta Stock

Only 2 hits are relevant to the financial meaning of "stock" in the top 10.  
From Altavista:

1. K of C Global Stock Quotes
2. Michael Furey Photography/Photography, Photography, stock (photos/stock photos)
3. ROCKY Posters - Stock & in Italy - Stock Report - Tuesday September 3, 1991
4. Cuban Stock Report - Friday, October 3, 1991
5. Stock 4 All: EDGAR (US&C)
6. NET IPOV - Liu Sida - Myster - stock exchange
7. The Official Vancouver Stock Exchange
8. Stock Club
9. RUSSELL HIRMAN DECLARES PREFERRED STOCK DIVIDEND
10. The Dallas Stock Exchange

### {stock share}

From Lycos:

- 1) Share, Stock or CD Secured Loan
- 2) Share / Stock Option Scheme Administration
- 3) Altavista - Stock, Share Division
- 4) One Share of Stock, Inc. - Overview Info
- 5) One Share of Stock - Product Line
- 6) Altavista New Zealand: Stock And Share Market Links (22Sep1996)
- 7) Altavista New Zealand: Stock And Share Market Links (22Sep1996)
- 8) Money: 858 can buy share of stock in a company
- 9) ONE SHARE OF STOCK, INC. - Order Form
- 10) One Share of Stock, Inc. - Company Info

These results are all relevant to the financial meaning of the word "stock".  
From Altavista:

1. South Africa: Stock market: Share price index (distribution format)
2. Domestic: Stock market: Share price index (base page)
3. ONE SHARE OF STOCK, INC.
4. Chile: Stock market: Share price index (base page)
5. American financial website show stock market money portfolio bank central f
6. Singapore: Stock market: Share price index (distribution format)
7. Mexico: Stock market: Share price index (base page)
8. Yohankende: Stock market: Share price index (base page)
9. Ireland: Stock market: Share price index (distribution format)
10. Japan: Stock market: Share price index (base page)

### Relevance feedback

Central problem in relevance feedback :

- searching "features" (words, phrases) from relevant documents
- calculating weights for these features in the context of a new query

given a relevant page

- first, looks for the keywords (assume  $K_i$  is one of the keywords)
- for each keyword  $K(i)$ , extracts the chunk
 
$$W_{-3}W_{-4}W_{-3}W_{-2}W_{-1}K_iW_1W_2W_3W_4W_5$$
- then, a bag of chunks are collected and passed to the processes of deleting the stop words and calculating the frequency
- top several frequent words are used to expand the current search keywords

### For example

suppose a user gives text as a relevant feedback to the search keywords "intelligent agent"

- URL : "http://www.cs.cmu.edu/~softagents"
- using our method
  - {software structure reusable architecture technology} used to expand the search "intelligent agent"

- 1) The Agent Building Shell: Programming Cooperative Enterprise Agents (http://www.inzorro.com/ELL/ABS-page/ABS-astroic)
- 2) The Agent Building Shell: Programming Cooperative Enterprise Agents (http://www.inzorro.com/ELL/ABS-page/ABS-astroic)
- 3) An Architecture for Supporting Quasi-agent Engines in the WWW (http://www.cba.uic.edu/~slim/11a/submitted/viewing)
- 4) Knowledge Sharing Papers (http://apps.stanford.edu/knowledge-sharing/papers/K)
- 5) Knowledge Sharing Papers (http://apps.stanford.edu/knowledge-sharing/papers/U)
- 6) Knowledge Sharing Papers (http://apps.stanford.edu/knowledge-sharing/papers/I)
- 7) The Agent Building Shell: Programming Cooperative (http://www.inzorro.com/ELL/ABS-page/ABS-astroic)
- 8) Special Issue AI in Medicine Editorial Special Issue Artificial Intelligence in Medicine "Architectures for Intelligent Systems Based on Reusable Components" (http://www.sci.psy.uva.nl/inst/Schreiber/papers/Mu)
- 9) CS 791A - Agent Architectures for Information Gathering (http://mitpress.cs.mit.edu/~jgordon/)
- 10) Introduction Protocol for Software Agents on the World Wide Web (http://rlab.jc.man.ac.uk/rlabman/www-96/interact)

