Using Wikipedia Content to Derive an Ontology for Modeling and Recommending Web Pages across Systems

Pei-Chia Chang
Department of Information & Computer Science
1680 East-West Road
Honolulu, HI 96822, USA
1-808-2209701
pcchang@hawaii.edu

Luz M. Quiroga
Department of Information & Computer Science
1680 East-West Road
Honolulu, HI 96822, USA
1-808-9569988
lquiroga@hawaii.edu

ABSTRACT
In this work, we are building a cross-system recommender at the client side that uses the Wikipedia's content to derive an ontology for content and user modeling. We speculate the collaborative content of Wikipedia may cover many of the topical areas that people are generally interested in and the vocabulary may be closer to the general public users and updated sooner. Using the Wikipedia derived ontology as a shared platform to model web pages also addresses the issue of cross system recommendations, which generally requires a unified protocol or a mediator. Preliminary tests of our system may indicate that our derived ontology is a fair content model that maps an unknown webpage to its related topical categories. Once page topics can be identified, user models are formulated through analyzing usage pages. Eventually, we will formally evaluate the topicality-based user model

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval-- Information filtering; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval -- Clustering D.3.3

General Terms
User Modeling, Wikipedia, Management, Measurement, Documentation, Design, Experimentation.

Keywords
Recommender, Agent, User Modeling, Ontology.

1. INTRODUCTION
User modeling through content is one common solution in recommending web pages across systems [3,7,9,14]. In this work, we are interested in using the collaborative content of Wikipedia to derive an ontology as a unified knowledge base for modeling web pages. Wikipedia is one of the world's largest collaborative knowledge bases. Although there are only a few contributors(less than 10% of user population) to the content of Wikipedia[8], it has a huge pool of readers. As Sussan describes, "with Web 2.0 products, it is the user's engagement with the website that literally drives it."[13] Similarly, we speculate Wikipedia's content and its vocabulary may cover recent and popular topical areas that people are generally interested in. The language in Wikipedia may be closer to what the general public use, instead controlled by domain experts. We emphasize the topics, but not content accuracy, from Wikipedia may reflect the dynamic information on the Internet.

Conference'04, Month 1–2, 2004, City, State, Country.

Our recommender formulates a user model based on the browsing behavior at a client side and the usage pages mapped to the derived ontology. Given the research potentials of Wikipedia's content, we are interested in the performance of recommending web pages based on the Wikipedia derived ontology. Our research question is "Does the recommender based on the Wikipedia's content model provide topically relevant recommendations?"

2. Related Work
Content-based recommenders include WebWatcher[6], Syskill & Webert[10], WebMate[5], and ifWeb[2]. WebWatcher and WebMate adopt TF-IDF, the vector space model and similarity clustering. Syskill & Webert rely on feature extraction, particularly expected information gain[11], which relies on the co-existence of related keywords, and relevance feedback. The system formulates the profile vector that consists of keywords from pages of positive ratings and against pages of negative ratings. Then, Bayesian classifier is employed to determine a page's topics, and its similarity with the profile vector. ifWeb employs a semantic network and consistency-based user modeling shell[4]. In general, these four systems apply statistical approaches, such as TF-IDF or expected information gain for keywords extraction and a cluster or classifier for similarity identification. Our work borrows Wikipedia's categorization system and augments it with keywords identified by predefined heuristics as topical indices. A full listing of existing categories and indexes in Wikipedia can be found at http://en.wikipedia.org/wiki/Portal:Contents/Categorical_index.
In our study, page classification depends on the frequency of those indexing keywords appeared in a web page. Our difference from the previous systems is the use of Wikipedia's collaborative categorization system to derive an ontology that is augmented with heuristic information extraction from Wikipedia's content.

3. Method Description

3.1 System Architecture

Our recommender uses the Wikipedia's content to derive an ontology for content and user modeling. With the ontology, our system automatically assigns the Wikipedia category(s) to a new page that pass a category's threshold value, which formulates a "categorical" vector space model for the page. The system also captures user interests in the user model through the categories. Pages of similar topics with the profile will be recommended.

Figure 1 depicts the system's architecture, which will be explained in the following paragraphs.

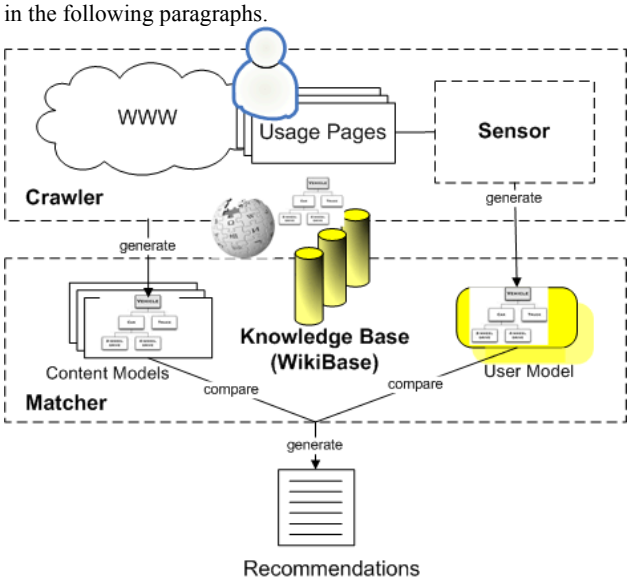

Figure 1 System Architecture

There are four major components in the system -- the crawler, the Wikipedia knowledge base (WikiBase), the sensor, and the matcher. To begin with the top part of the graph, the crawler fetches those hyperlinked pages from the usage pages as well as queries search engines based on the user model managed by the sensor. Utilizing the sensor, the crawler generates a corresponding content model for each newly fetched page.

Every component in the system uses WikiBase, which stores ontologies, keywords, content models and the user model respectively. We construct WikiBase by combining the Wikipedia's categorization system with heuristic information extraction on keywords. Heuristics include page titles, categorical labels, anchor texts, italic, bold, and TF-IDF terms. In order to associate keywords with categories, we extract heuristic keywords form pages labeled as one of the categories by the Wikipedia's editors. Therefore, each category has a list of keywords to be utilized by the sensor.

The sensor manages the user model and maps usage pages into content models. It calculates a page's topical relevance and formulates the corresponding content model according to the WikiBase's keyword weight and the word frequency of the web page. The user model is updated, on a frequency basis, by the sensor whenever it maps a usage page into a content model. Therefore, the user model is constantly evolved. In other words, if a user accesses a specific categorical topic in multiple times or through multiple pages, the user will score higher in the corresponding category of the user model. Keyword weighting and sensing formulas are defined below.

Definitions:
$|K_j|$, the number of keywords extracted for heuristic type j
$|K \in c|$, the number of keywords in category c
$|Categories|$, the number of categories in the knowledge base
$freq(K_{ij})$ is the frequency of keyword K_{ij} for heuristic type j
$max(K_1, \ldots, K_m)$ is the maximum value among the m elements

The weight of keyword K_{ij} among m heuristics is:

$$W(K_{ij}) = \sum a_j \left(\frac{freq(K_{ij})}{\max(freq(K_{1j}), \ldots, freq(K_{|Kj|j}))} \right)$$

$1 \le i \le |K_j|, \ 1 \le j \le m,$
a_j is a weighting coefficient assigned to heuristic j.

A page's Relevance Score R_c to a category c is:

$$R_c = \sum d \, W(K_i \in c) \ \alpha^{freq(K_i \in c)}$$

$1 \le i \le |K \in c|, \ 0.5 < \alpha < 1, \ 1 \le c \le |Categories|$

$$\begin{cases} 0 < d < 0.5, & \text{partial match} \\ d = 1, & \text{full match} \end{cases}$$

As for the matcher, it compares the cosine similarity of those crawler-retrieved pages with the user model and then generates recommendations. In addition to cosine similarity, the matcher also relies on the ontological structure of WikiBase. With the structure, topical association among web pages can be revealed and it also helps to identify if a user is interested in particular domains or not. We define two indices (diversity and specificity) to represent the coverage of user interests. The following describes the procedure.

At the beginning, construct a minimal spanning tree that traverses all the identified categories in the user model. Identified categories are those categories with a Relevance Score over a predefined-threshold. In order to connect identified categories together, connecting nodes, such as parents or neighbors of the identified categories may be added to the tree. Definitions of the two indices are as follows:

Diversity index: count the number of edges of the minimal spanning tree and normalize it by dividing the number of identified nodes, excluding connecting nodes, in the spanning tree.
Specificity index: sum the minimal distances from the root to all identified categories respectively and normalize it by dividing the number of those identified categories, excluding connecting nodes.

3.2 Evaluation Method

We plan to recruit a few participants (< 10) in the computer science domain where we derive WikiBase. Each of them has to rate a collection (> 300) of web pages based on topical relevance and novelty. They have to provide certain web pages (>30) of their interest in advance as the usage source of formulating the

user models. Afterward, they have to rate the collection. The ratings will be divided into a training and validation set. Our system will tune the keyword weight based on the training data. We will compare our system performance with the SMART system[12], which utilizes vector space model as well.

4. Current Status & Discussion

4.1 Current Status

We have built WikiBase in the computer science domain, listed at http://www2.hawaii.edu/~pcchang/ICS699/results.html. We selected the domain due to its rich data. Two preliminary tests were conducted on two computer science professionals. In the first one, w tested the following pages for their topical relevance.

http://www.algosort.com/ (A)
http://tc.eserver.org (B)

Considering only the top two ranking, page A is sensed as "algorithms" and "genetic algorithms" categories; page B is sensed as "human-computer interaction" and "usability" categories. In the evaluation of the classification result, both participants' rankings are the same as the system's ranking, considering only the top two.

In the second test, we selected fourteen pages, listed in the appendix, from four topical areas – algorithm, data mining, human computer interaction (HCI) and computer games. Both participants have to evaluate at least five categorical keywords of each page. They have to provide the degree of agreement from 1 (disagree) to 5(agree) about the following statement. "The given phrase is a topical keyword of the page." The given phrase is a categorical label generated by the sensor for each page. The following table summarizes the ratings.

	Participant 1	Participant 2
Algorithm (3 pages)	3.88	2.83
Data Mining (4 pages)	3.95	3.40
HCI (4 pages)	4.22	4.27
Games (3 pages)	2.67	2.2
Average	3.67	3.2

Table 1 Evaluation of Categorical Keywords

From the result, the ranking of both participants' average scores is: HCI, Data Mining, Algorithm, and Games. Except for the game topic, the agreement score is around 4 for participant 1 and 3.5 for participant 2. We suspect that due the wide coverage of computer games, our system performs worse in that category. Another reason may be because of the nature of computer science category. It reflects the common scientific techniques of theory for producing computer games, which is different from the tested pages that viewing computer games from a player's perspective.

We are still in the process of tuning up the keyword weight by utilizing the computer science pages from Open Directory Project (DOP) [1]. Pages in ODP are manually categorized by its users and we use the classification to evaluate our sensor. As for evaluating the recommendations, we are training the matcher with pages of a different topical coverage. Eventually, we will apply the evaluation method described earlier.

4.2 Discussion

Using the Wikipedia categories as an ontological model yields a simple user profile. This modeling approach benefits significantly in cross-system recommendations. Our recommendation engine works at the client side, which eases the privacy concern of disclosing sensitive information at web servers. Combining categories with heuristic information extractions leaves rooms for the selection of heuristics. Different domains or user groups are able to apply heuristics of interest. Given the above mentioned advantages, we are looking forward to see the results of our evaluation.

5. Future Work

The Wikipedia content and categorization system play an important role in our method to generate recommendations. Our work emphasizes the framework to automate the ontology generation and its performance in recommendations. Nevertheless, the quality of Wikipedia content is controversial. It will be worthwhile to adopt the same framework to another Wikipedia-like platform with a different user group, such as domain experts, to ensure the content quality.

Another interesting area is to study the content statistics, such as volume or the granularity of the categories, with recommendation performance. Not every domain in Wikipedia contains rich categories and articles like computer science. Therefore, the performance of recommendations may be related to some of the statistics.

6. Appendix

Due to limited space, only 1st page of each selected topic displays the categorical keywords.

Algorithms
http://www.algosort.com/
(Algorithms, Genetic algorithms, Root-finding algorithms, Networking algorithms, Disk scheduling algorithms)
http://www.oopweb.com/Algorithms/Files/Algorithms.html
http://cgm.cs.mcgill.ca/~godfried/teaching/algorithms-web.html

Data Mining
http://www.data-mining-guide.net/
(Databases, Algorithms, Knowledge representation, Natural language processing, Knowledge discovery in databases, Machine learning, Data mining)
http://www.thearling.com/
http://databases.about.com/od/datamining/Data_Mining_and_Data_Warehousing.htm
http://www.ccsu.edu/datamining/resources.html

HCI
http://www.pcd-innovations.com/
(Human-computer interaction, Human-computer interaction researchers, Usability, Computer science organizations, Artificial intelligence, Software development)

http://www.nathan.com/
http://nooface.net/
http://www.hcibib.org/

Games
http://www.robinlionheart.com/gamedev/genres.xhtml
(Image processing, Computer programming, Demo effects
Regression analysis, Computer graphics)
http://open-site.org/Games/Video_Games/
http://www.literature-study-online.com/essays/alice_video.html

7. REFERENCES

[1] http://www.dmoz.org/

[2] Asnicar, F., & Tasso, C. 1997. ifWeb: A Prototype of User Model-Based Intelligent Agent for Document Filtering and Navigation in the World Wide Web. In Proceedings of the 6th International Conference on User Modeling.

[3] Billsus, D., & Pazzani, M. 1999. A Personal News Agent that Talks, Learns and Explains. In Proceedings of the 3rd Ann. Conf. Autonomous Agents.

[4] Brajnik, G., & Tasso, C. 1994. A Shell for Developing Non-Monotonic User Modeling Systems. Human-Computer Studies, 40, 31-62.

[5] Chen, L., & Sycara, K. 1998. WebMate: a personal agent for browsing and searching. In Proceedings of the second international conference on Autonomous agents, Minneapolis, Minnesota, United States

[6] Joachims, T., Freitag, D., & Mitchell, T. 1997. WebWatcher: A Tour Guide for the World Wide Web. In Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence.

[7] Mooney, R. J., & Roy, L. 2000. Content-based book recommending using learning for text categorization. In Proceedings of the fifth ACM conference on Digital libraries, San Antonio, Texas, United States.

[8] Ortega, F., Gonzalez-Barahona, J. M., & Robles, G. 2008. On the Inequality of Contributions to Wikipedia. In Proceedings of the 41st Annual Hawaii International Conference on System Sciences.

[9] Pazzani, M., & Billsus, D. 1997. Learning and Revising User Profiles: The Identification of Interesting Web Sites. Machine Learning, 27, 313-331.

[10] Pazzani, M., Muramatsu, J., & Billsus, D. 1996. Syskill & Webert: Identifying Interesting Web Sites. In Proceedings of the Thirteenth National Conference on Artificial Intelligence, Portland, Oregon, United States.

[11] Quinlan, J. R. 1986. Induction of Decision Trees. Machine Learning, 1(1), 81-106.

[12] Salton, G., & Lesk, M. E. 1965. The SMART automatic document retrieval systems -- an illustration. Commun. ACM, 8(6), 391-398.

[13] Sussan, G. 2007. Web 2.0 The Academic Library and the Net Gen Student (pp. 35): ALA editions.

[14] Zhang, Y., Callan, J., & Minka, T. 2002. Novelty and Redundancy Detection in Adaptive Filtering. In Proceedings of the 25th Ann. Int'l ACM SIGIR Conf.