Abstract—A new ontology-supported and hybrid recommending information system for scholars was proposed. Not only can it fast integrate specific domain documents, but also it can extract important information from them through the hybrid filtering technology to take information integration and recommendation ranking. The experiment outcomes proved that the reliability and validity measurements of the whole system performance can achieve the high-level outcomes of information recommendation. Furthermore, this paper also discussed and investigated the advantages and shortcomings of the construction of a recommendation system with different approaches and accordingly provided the design philosophy of customized services for recommendation systems.

Keywords— Ontology, Webpage Crawlers, Webpage Classifiers, Information Extractor, Information Recommender.

I. INTRODUCTION

Nowadays, most of search engines adopted the way of keyword-based query which has the problem: the keywords entered by users were not completed and not able to obviously indicate the query demands of users. Furthermore, there are so many keywords being the same words with different meaning in different fields. The system would finally produce many complicated cross-field query outcomes when system didn’t respectively managed to classifying query requisition and specifying fields [7], which resulted in the information demanders may spend much time to filter out the available information. In technical literatures, lots of system employed the concept of ontology as the core technology for solving the above problems, for example, WebSifter II [10], OBIGA [2], Swoogle [3], etc. Ontology can provide complete semantic models, which means in specified domain all related entities, attributes and base knowledge among entities have sharing and reuse characteristics which could used solving the problems of common sharing and communication. To describe the structure of the knowledge content through ontology can accomplish the knowledge core in a specified domain and automatically learn related information, communication, accessing and even induce new knowledge. This paper also relied on the concept of ontology for solving the problem of returned many complicated query outcomes by the same words with different meaning or the different words with same meaning among information sources.

Data mining employed the techniques of statistic analysis, information classification, or machine learning to provide important decision bases of information systems in according with significant reasons, relationships, or their potential rule models found from the huge amount of datum. The six common data mining methods include description, estimation, clustering, classification, prediction, and association [16]. This paper also especially combined a recommender system with the classification and related association rules of SPSS Clementine [16], a data mining tool, to mine out usefully important information from huge datum, and then to explore the superiority of constructing information recommenders with different structures or approaches and accordingly provided the design philosophy of customized services for recommendation systems.

II. BACKGROUND KNOWLEDGE AND DEVELOPING TECHNIQUES

A. Ontology

Ontology [1] can provide complete semantic models, which has sharing and reuse characteristics that could used solving the problems of common sharing and communication; hence, ontology is a powerful tool to construct and maintain an information system [19]. This paper adopted Protégé [6] to construct the scholar ontology and its most special feature is that used multi components to edit and make ontology and led knowledge workers to constructing knowledge management system based on ontology; furthermore, users could transfer to different formats of ontology such as RDF(S), OWL, XML or directly inherit into database just like MySQL and MS SQL Server, which have better supported function than other tool [4].
B. Regular Expression

Regular expression is a character queue to describe specified order, which could used to search matched pattern in another character queue. Regular expression can use universal words, set of words, and some quantifiers as specifying ways [8]. There were two supported classes for this expression: Pattern and Matcher, and we would use Pattern to define a Regular expression. If we want to conduct pattern matching with other character queue, we would use Matcher. However, the regularization of webpage classifier meant the system removed the unmeaning words, including continuous blank spaces, line feed, Tab character, punctuation marks, and so on, in classification from being classified documents for up-rising the precision rate of classification [20].

C. Processing of Classification

The conception of TF was first proposed by Salton and McGill [14] while IDF was proposed by Spark Jones [15]. The reasons why they proposed the two methodologies was that the importance of every term appearing in the document was not quite the same, most of all, the importance was not necessarily the same even appearing in different articles. Therefore, combining the two methodologies could measure the importance of the feature term. Traditional statistics classifier must be accompanied with proper methodology of extracting feature and fetch out the proper features from training web-pages so as to gain classification precision. Hence, the quality of set of term feature would decide the classification precision. From the related literature, there were many resolution such as [15], an extracting mechanism with squared related constancy and Entropy; Tan [17] especially proposed one fairer feature-select method, in which all kinds of document could be treated fairly and the input dimension could be decreased more. From those theories above, the conception of either taking advantage of information theory or machine learning was inclining to use symbols (or numbers) calculation. The processes and its outcomes are difficult to be understood. This paper adapted domain ontology to support the processing of classification that it not only arose the classifier’s efficiency but also made the processing and outcomes of the classification easily understood.

D. Developing Techniques

The developing tool of our system is Borland JBuilder2006. It is an integrated development environment of Java, which have a fine human-machine interface and code debugging mechanism to make a fast integration of each code block when the system was developed, and accordingly reduce the time of system development. In addition, Java [11] provides lots of functions and methods to integrate web applications and databases. In the view of extensibility, Java is absolutely the optimal choice for solving the problem of cross platform. This system adapted MS SQL Server as backend knowledge-database sharing platform based on ontology. MS SQL Server is one mostly used relational database management system. SQL (Structured Query Language) is one query language to get the data in the database.

III. SYSTEM ARCHITECTURE

A. Webpage Crawler

Figure 2 showed the operation system structure of the webpage crawler [20]. Firstly, Keyword Keying transferred input characters into URI code and then embedded into the search engine’s query URL. Furthermore, Search Engine Linking declared an URL object and added the query URL on well transferred URI code, and then read its content line by line for outputting the content as text file as final analysis reference. Moreover, Regular Processing used different regular expressions to search for whether there is matched URL so as to completely fetch out related hyperlinks corresponding to the conditions for outputting them into a text file to provide the system for further processing. Pure Text Extracting used the hyperlink file to read its content with an iterative loop line by line, and then the system deleted the html tags from source file and remained only the text content so as to let system conduct further processing and analyzing. Finally, Content Filtering judged whether the webpage was the range we queried. It linked the ontology database and fetched out the content to compare content of the webpage. If did, the content, its URL and related titles were stored by the system for supporting further processing and analyzing.

B. Webpage Classifier

Figure 3 illustrated the system operational structure of the webpage classifier [21]. The Source Text originated from the webpage crawler was described before. Firstly, the system filtered out non-semantic characters such as the continuous space, Tab, \n, \t characters, etc., which were used to divide or beautify the content and had nothing to do with semantic expression. The initial task includes loading stop word database, ontology database, formal database, and document array so as to manage afterwards. Furthermore, this system employed the CKIP (short for Chinese Knowledge and Information Processing) segmentation system [12] to be an assistant tool for doing the word segmentation, whose output contained segmented words and their corresponding attribute tags. Preprocessing contains: Format Transformation, which included attribute tag deleting, full-space character replaced with half-space character, etc.; Segmentation Fixing, which employed the domain ontology to fix those segmentation errors in the specific domain; Stemming, which transferred different word type into stem; and Stop Word Filtering, which not only used “stop list” to store those words and ought to be excluded but also employed the attribute tags after segmentation to assist in stop word deleting process.

Deduplication Processing proceeded complete cross-comparison between those documents and system database for deleting duplication webpages to avoid duplicated operation of system backend and accordingly enhance its performance. TF/IDF
Processing carried out word weight calculation with TF and IDF datum. The IDF process includes Related Document Collection, which accumulated 500 pieces of webpages from the member of Taiwanese Association for Artificial Intelligence as the document specimen; and IDF Calculation, which deleted the duplicated vocabularies from the specimen of documents and then calculated number of appearances of each remained in the specimen of documents as the IDF.

Cosine Similarity [5] compared with TF/IDF difference of each vocabulary among specimen of classes to calculate those classes’ similarities. This system takes one step ahead to do the following three processes for discriminating most similar documents in domains. Firstly, the system raised the weight of the keywords appeared in the title, namely, Title Weight Raising. Furthermore, the weights of the keywords were assigned according to their level values located in the ontology hierarchy, namely, Hierarchical Weight Raising, to help for discriminating most similar keywords in domains. Finally, threshold Filtering means the system aimed at TF/IDF values of keywords and filtered out the lower value of vocabularies before delivered them into cosine similarity calculation for avoiding too many noises and affecting the similarity calculation [9], whose value of this system was set to 0.7. Finally, the system stores the final classifications into the proper data folds so as to conveniently process afterwards.

C. Information Extractor

Figure 4. Architecture of the information extractor

Figure 4 illustrates the architecture of the information extractor. It combined datum both from the webpage crawler and from the webpage classifier and read the document’s html source code, URL belonged classification, and corresponding file name to extract significant information from them. Preprocessing works mainly include URL Fixing and Sub-webpage Crawling. The former means the URL of the sub-webpage may be written in the internal hyperlink form of the website. But the system directly employed the webpage crawler to get that webpage and then returned an empty webpage. Therefore, the system needs to use the getHost() method in Java’s API for getting the host URL of that webpage and then combined it with the internal hyperlink, and accordingly got its correct sub-webpage. The latter means the specific information may exist in its sub-webpages, therefore, the system should crawl down to the specific sub-webpages in the next level. Regular Processing employed the regular expression technique to aim at the three types of the system needed to proceed with significant information analysis and extraction, including course information, website recommendation, and academic activities. The system outputs the Significant Information Files after regular processing as text files based on different classes so as to conveniently process afterwards.

D. Information Recommender

The information recommender only choose the data folds of the different classification so as to fast, precisely, and effectively query corresponding ones about scholar information, and accordingly integrated and ranked those significant information for recommending to users. Figure 5 illustrates the architecture of the information recommender. The system used the number of duplicated hyperlinks to be the base of Website Recommendation, i.e., the weight of the website is higher when many people recommend it on this class. The system would consult whether some similar Course Information with the classic scholar pattern existed. If did, its weight is higher. As the same word, if the course information appeared in the significant vocabularies of the belonged classification, so is higher its weight. Academic Activities recommendation as same as the operation of course information, the system used the classic scholar pattern and class keyword to carry out weight promoting process. The system also consulted the scholar ontology database to retrieve keywords of each class for convenient processing of information recommendation. The source of the classic pattern of each class in this system derived from 50 classic members of Taiwanese Association for Artificial Intelligence and Taiwan Fuzzy Systems Association, which extracted personal information from their webpages as specimens of Classic Scholar pattern. Finally, the information recommender produces the recommending information according to the order of keyword weight through the integrated interface available to users.

E. Integrated Interface

Figure 6 illustrates the integrated interface of this system. It is not only to be a communication bridge but also presents the operation process of the webpage crawler, webpage classifier, information extractor, and information recommender, respectively. Finally, the interface also provides the processing function of IDF to users for conveniently adding related IDF data by themselves. The user can click those tags in the left upper area of the interface screen to view the processing procedure of each system stage for deeply understanding their operations.

Figure 6. Integrated interface of this system

F. Construction of Ontology Database

Our ontology is a knowledge sharing database which was constructed for specific domain. That is to say in which took advantage of built ontology database of some scholars to support
whole system operation. In the mentioned ontology database, it included two constructed stages; one is statistics and analysis of related concepts of scholars, the other is construction of ontology database. The procedures were detailed as below:

First of all, the system conducted statistics and survey of homepage of related scholars to fetch out the related concepts and their synonym appearing in the homepage. Figure 7 indicated the structure of domain ontology of scholars in Protégé. In application, the system combined these related concepts that would be conveniently interpreted by the webpage crawler to compare with content of the queried webpage, and if the compared outcomes were corresponding to any item among the matched condition for web page querying. Figure 8 shows the second stage of ontology constructing of scholars, in which the main part work is to transfer the ontology built with Protégé into MS SQL database. The procedures are as following:

1. Exporting an XML file constructed with Protégé knowledge base and then importing into MS Excel for correcting.
2. Finally importing MS Excel into MS SQL Sever to finish the ontology construction of this system.

![Ontology database transferring procedures of scholars](image)

**Figure 8.** Ontology database transferring procedures of scholars

**IV. SYSTEM DISPLAY AND PERFORMANCE VERIFICATION**

**A. System Display**

![Execution screen of webpage crawler](image)

**Figure 9.** Execution screen of the ontology-supported webpage crawler

Figure 9 shows the execution screen of the ontology-supported webpage crawler, while Figure 10 displays the information recommendation results.

![Result screen of information recommender](image)

**Figure 10.** Result screen of information recommender

**B. Verification and Comparison of System Performance**

The information recommendation meant the optimal recommending have chosen from a group of related information sets. That wonderfully possessed different approaches to the same purpose as whether sampling specimens can be on behaving of degree of sampling body in huge amount of datum. In the sampling survey domain, the reliability was usually employed to measure the degree of precision of sampling system itself, while the validity was emphasized whether it can be correct to reflect the properties of the appearance of things. This paper employed the aid of mathematic model, provided by J.P. Peter [13] in 1979 and cited by lots of papers, to represent the definitions of the reliability and validity [22].

In this experiment, we randomly chose 100 data from the personal webpages of the member of Taiwanese Association for Artificial Intelligence to carry different 3 times of recommendation experiments out respectively. The significant information recommendation of this experiment was asserted by the domain experts, including observed values, true values, error values, and related variances. We respectively calculate reliabilities and validities of information recommendation in different professional domains as shown in TABLE I, while the total average results were shown in TABLE II. The average values of reliability and validity were 0.88 and 0.78, respectively. From the technical literature [18], we know the regular-level values of reliability and validity were 0.7 and 0.5, respectively, which verify and validate our experiment results with high-level outcomes of information recommendation.

<table>
<thead>
<tr>
<th>TABLE I. RESULTS OF 3 TIMES OF RECOMMENDATIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TABLE II. TOTAL AVERAGE RESULTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance</td>
</tr>
<tr>
<td>-------------</td>
</tr>
<tr>
<td>Average Reliability</td>
</tr>
<tr>
<td>Average Validity</td>
</tr>
</tbody>
</table>

In addition, to explore the superiority of constructing information recommenders with different structures or approaches, this paper also especially combined a data mining tool SPSS Clementine with the domain ontology to mine out usefully important information from huge datum, and then to employ Java to develop an another information recommender for scholars [22], in
which can recommend suitably important information to scholars. Figure 14 showed the operation system structure of this recommender. The recommender also employed the CKIP segmentation system to be a front-end assistant tool, which segment webpage contents of scholars and filter out most part of stop words. The CKIP segmentation system could make segmentation errors in some terminologies of specific domains. Those errors could make an enormity wrong effect in the precision rate of backend word matching. The function of SegmentationFixing emphasized to solve those problems, detailed function included: loading stop word database, deleting non-semantic characters, filtering stop words, re-segmenting process supported by the domain ontology, and term frequency statistics.

![Figure 11. Operation system structure of the recommender with SPSS Clementine](image)

SPSS Clementine System entered the text file after processed by Segmentation Fixing and combined against the domain ontology database to be the bases of classification statistics and recommending match analysis. This processing divided into two stages: the first stage judged the scholar webpage whether fit in with specific domain, i.e., content-based filtering (same as the function of Content Filtering of the webpage crawler described in Section III.A); the second stage extracted the significant information from the scholar webpage to be the important recommending base. In the recommending mechanism, not only has the general normality recommending but also adding the optimal recommending module. For example, if existed the duplication list of giving lessons between a scholar’s and some related scholars’, these significant courses will be the optimal information of “Courses” recommending, i.e., collaborative filtering (just like the role of Classic Schol ar Pattern of the information recommender described in Section III.D). Recommending Display showed the recommending results after mining by SPSS Clementine, which employed Java to implement the user interface of this recommender as shown in Figure 15.

![Figure 12. Recommendation results of the recommender with SPSS Clementine](image)

The experiment used the same set of domain ontology, specimen of webpages, and verification rules (i.e., reliability and validity) and the system presented almost same outputs to do the related experiments. TABLE III illustrated the reliabilities of some scholars’ recommending information on “Course Information” and “Academic Activities” which are 0.856 and 0.756 in average, respectively. TABLE IV illustrated the validities of some scholars’ recommending information on “Course Information” and “Academic Activities” which are 0.856 and 0.756 in average, respectively. Finally, what merits attention is: the last column Professional Classification of each scholar can be accurately shown that prove the recommendation system with SPSS Clementine has its accuracy and availability.

### TABLE III. RESULTS OF THE RELIABILITY OF CLASSIFICATION INFORMATION

<table>
<thead>
<tr>
<th>Domain</th>
<th>Course Information</th>
<th>Academic Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.S. Ho (F5-0-01)</td>
<td>12 0.92 2 7 0.71</td>
<td>AI</td>
</tr>
<tr>
<td>T.W. Kuo (F5-0-03)</td>
<td>0 2 1 0 1</td>
<td>AI</td>
</tr>
<tr>
<td>S.Y. Yang (F5-0-01)</td>
<td>0 3 1 0 1</td>
<td>AI</td>
</tr>
<tr>
<td>S.M. Chen (F5-0-01)</td>
<td>7 11 0.36 5 7 0.57</td>
<td>Fuzzy</td>
</tr>
<tr>
<td>W.L. Hsu (F5-0-01)</td>
<td>0 1 1 2 4 0.5</td>
<td>AI</td>
</tr>
<tr>
<td>Average</td>
<td>0.856</td>
<td>0.756</td>
</tr>
</tbody>
</table>

### TABLE IV. RESULTS OF THE VALIDITY OF CLASSIFICATION INFORMATION

<table>
<thead>
<tr>
<th>Domain</th>
<th>Course Information</th>
<th>Academic Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>C.S. Ho (F5-0-01)</td>
<td>11 12 0.92 2 7 0.71</td>
<td>AI</td>
</tr>
<tr>
<td>T.W. Kuo (F5-0-03)</td>
<td>2 2 1 1 1</td>
<td>AI</td>
</tr>
<tr>
<td>S.Y. Yang (F5-0-01)</td>
<td>5 5 1 1 1</td>
<td>AI</td>
</tr>
<tr>
<td>S.M. Chen (F5-0-01)</td>
<td>4 11 0.36 5 7 0.57</td>
<td>Fuzzy</td>
</tr>
<tr>
<td>W.L. Hsu (F5-0-01)</td>
<td>1 1 1 2 4 0.5</td>
<td>AI</td>
</tr>
<tr>
<td>Average</td>
<td>0.806</td>
<td>0.708</td>
</tr>
</tbody>
</table>

After comparing TABLES II, III, and IV, the values of recommending reliability and validity of our recommendation system (i.e., 0.88 and 0.78) were not only better than the regular ones of the literatures (i.e., 0.7 and 0.5) but also showed the excellent performance than the ones of the recommender with SPSS Clementine (i.e., 0.806). From the above comparison, the approach with SPSS Clementine not only has its availability and system performance but also can reduce the whole time of system construction. However, its shortcomings include the learning path and time of SPSS Clementine is too sharp and so long, moreover, its limitations on input and output cannot completely fit with some specific applications for customized services. Our approach could fortuitously solve the above problems and spread out the excellent system performance indeed and accordingly provided the design philosophy of customized services to recommendation systems.

V. CONCLUSION AND DISCUSSION

This paper had focused on developing an ontology-supported and hybrid recommending information system for scholars. Not only can it fast integrate specific domain documents, but also it can extract important information from them through the hybrid filtering technology to take information integration and recommendation ranking. The experiment outcomes proved that the reliability and validity measurements of the whole system performance can achieve the high-level outcomes of information recommendation. Furthermore, this paper also discussed and investigated the advantages and shortcomings of the construction of a recommendation system with different approaches and accordingly
provided the design philosophy of customized services for recommendation systems.

This paper continually employed the related techniques of the ontology-supported webpage crawler, namely OntoCrawler [20], and the ontology-supported webpage classifier, namely OntoClassifier [21], published by our laboratory before. This system employed the CKIP segmentation system (described in Section III.B) to assist and improve the original classifier’s drawback for catching up on regrets resulting from only processing English documents. Furthermore, it largely modified the classification rules to adopt the concepts of TF/IDF rules and cosine similarity method for calculating the degree of webpage similarity, which improved the performance of document classification supported by ontologies, and accordingly enhanced the precision rate of classification. On the viewpoint of the ontology-supported webpage crawler, it is very complete working on the whole functions. In this paper, however, we only modified part of webpage catching mechanism and output file format for conveniently and effectively combining with the operation of each sub-system in the backend.

In the future, we will focus on practicable modulation of each sub-system in this system. Currently, all sub-system modulation was not workable, therefore, the system cannot properly operate on the different domains via purely modified the ontologies only, which still have to add and modify part of program codes for achieving this ultra goal. Looking forward to practicable modulation of each sub-system, the system could provide more perfect human-machine interface, which not only assists system users to add the domain ontologies but also make the system properly operating on those ontologies to take one step ahead for extending the practical applications of this system.

ACKNOWLEDGMENT
The authors would like to thank Ssu-Hsien Lu, Ting-An Chen, Chi-Feng Wu, Hung-Chun Chiang, Hsieh-I Lo, and You-Jen Chang for their assistance in related sub-system implementation and experiments. The partial work was supported by the National Science Council, Taiwan, R.O.C., under Grants NSC-98-2221-E-129-012 and NSC-99-2623-E-129-002-ET, and the Ministry of Education, Taiwan, R.O.C., under Grant Skill of Taiwan (1) Word No. 0990045921s.

REFERENCES
[16] SPSS Taiwan Corporation, Handout of training course on SPSS Clementine, Taipei, Taiwan, 2006.