PERSONALISED NEWS RECOMMENDATION SYSTEM BASED ON USER INTERESTS

Madhu K P[1], D Manjula[2]

[1]*Research Scholar, Department of Computer Science and Engineering,*
Anna University,Chennai, India - 600025
[2]*Professor, Department of Computer Science and Engineering,*
Anna University,Chennai, India – 600025

ABSTRACT

We present "content based RSS Aggregator", a system that crawls, filters, and aggregates vast numbers of RSS feeds, delivering to each user a personalized feed based on the user interests. It consists of a three-tiered network of crawlers that scan web feeds, filters that match crawled articles to user subscriptions, and reflectors that provide recently-matching articles on each subscription as an RSS feed, which can be browsed using a standard RSS reader. In contrast to the common crawling mechanisms our system is focalized on fetching the latest news from the major and minor portals related to user's interests by utilizing their communication channels. The challenge here is that the system has to be updated with news as soon as they occur. In order to achieve this we utilize the communication channels that exist in the architecture of the WWW and more specifically in almost every modern news portals. As the RSS feeds are used by every major and minor portal it is possible to keep our crawler up to date and retain a high freshness of the offline content that is maintained in our system's database by applying algorithms in order to observe the temporal behaviour of each RSS feed. Our vision is to provide users with the ability to perform content based filtering and aggregation across millions of Web feeds, obtaining a personalized feed containing only those articles that match the user's interests. Rather than requiring users to keep tabs on a multitude of interesting sites, a user would receive near-real-time updates on their personalized RSS feed when matching articles are posted in any of the news portals.

Keywords: Filter, News Recommender, RSS Link Extractor, User Behaviour, User Interests

I INTRODUCTION

There has been a dramatic increase in the use of XML data to deliver information over the Web. In particular, personal Weblogs, news Web sites, and discussion forums are now delivering up-to-date postings to their subscribers using the RSS protocol. It is really tedious for the new users to find news articles pertaining to him from this huge amount of web resources. Also, the contents are getting updated evrey now and then. This adds to the difficulty of the user. The news articles are to be fetched and presented to the user in a timely manner. The user should be provided with the latest updates as and when it happens. To help users access new content in this RSS domain, a number of RSS aggregation services and blog search engines have been introduced recently and are gaining popularity [1]. Using these services, a user can subscribe news articles of his interests so that the user will

be notified whenever new contents appear at the sources as soon as the user logs in the service. We are handling a keyword-based search to retrieve all content containing the keyword which represents the user interest.

A set of domain names or the website links are collected. These domains are crawled for the valid RSS links. All advertisement links are filtered and removed by checking against the domain name. Certain news portal sites provide RSS service through third parties like feeds .feedburner.com. Such links are not removed as they donot match domain name. Now these RSS links are stored in the database for reducing the future crawling time and data usage. The RSS feed contents are mined based on the user interests. Using the summary of the news articles, user interests are categorized accordingly using tf-idf model and the corresponding category is updated in the database. These RSS links are periodically checked in order to get the instant feeds. The news articles under each interest are processed to obtain the topics and are recommended to the users. The topic modelling using Named Entity Recognition Technique is handled here to extract the crisp topics efficiently. These topics are suggested to the user as they are closely related to the user's area of interests.

The architecture of this type of problem usually consists of multiple subsystems which are assigned with specific roles in order to achieve high speeds. The basic parts of the system are:

- The centralized database
- The crawlers controller
- The terminals that execute the fetching and analysis

The database is used for storing permanent information, the controller is used in order to organize and distribute the procedure and finally the terminals are used in order to fetch the HTML pages from the internet.

The rest of the paper is organized as follows. In section 2, we describe about the Literature Survey, where we discussed the issues related to crawling and how it is handled.In section 3, we describe about the news recommendation system architecture. In section 4, we describe about the System implementation, where the over-all architecture of this project, Input and Output of each module, Database Scheme are discussed. In section 5, we explained about the results and performance evaluation . Next in section 6, we describe conclusion of the work.

II RELATED WORK

One of the important challenges in building an effective RSS aggregator is to minimize the delay between the publication of new content at a source and its appearance at the aggregator. Note that the aggregation can be done either at a desktop, for example, RSS feed readers or at a central server, for instance, a personalized Yahoo!/Google homepage. Although some of the developed techniques can be applied to the desktop based aggregatio, we primarily focus on the server-based aggregation scenario. The informat on in the RSS domain is often time sensitive. Most new RSS content is related to current world events, so its value and significance deteriorates rapidly as time passes. An effective RSS aggregator, therefore, has to retrieve new content quickly and make it available to its users close to real time [3].

The topic modelling is applied using Latent Dirichlet Allocation (LDA) via Variational Bayes (VB) to detect hidden topics in the document [2]. Feed management and categorization is a problem with current RSS technologies. A novel approach is presented in [4] for delivering news items from RSS feeds, based on the existing text categorization and Web service techniques. But this does not take into account the user interests. So

personalization is not done in the case of news recommendation. Personalization would result in more accurate recommendation of the news articles to the user.

The contents of the web pages are changing dynamically at different rates which means that the crawler should decide which page should be revisited by using an efficient method. This leads to creation of crawlers that have at least two basic modules, one for periodical crawling and another for incremental crawling to update the most frequently changing pages. Apart from the freshness other issues also occur when creating a crawler. Especially when creating a distributed crawler, either with terminals or multithreaded, the distribution of resources among the crawlers and the communication between them is to be considered a lot.

III NEWS RECOMMENDATION SYSTEM ARCHITECTURE

The system is meant to provide personalized news to the users based on their interests.

3.1 Preliminary Design

The crawler service takes in a list of source feeds given as URLs and periodically crawls the list to detect new articles. A naive crawler would periodically download the contents of each source feed and push all articles contained therein to the filters. The filter service receives updated articles from crawlers and matches those articles against a set of subscriptions. When an article is matched against a given subscription, each word of the subscription is marked either true or false based on whether it appears anywhere in the article; if the resulting Boolean expression evaluates to true then the article is considered to have matched the subscription. The final component is the reflector service, which receives matching articles from filters and reflects them as a personalized RSS feed for each user. The first k news article feeds will be chosen either by the hits or by the user polling rate. Then those k article feeds are delivered via web services.

3.2 Overall Architecture

Overall system Architecture designed in such a way that it reduces the data usage and execution time. Execution time mainly depends on number of HTTP requests so as data usage. Extracting news from web involves repeated crawling. Since it involves repeated crawling certain steps that are followed in each of the iterations remains same. Hence these steps are identified and executed only during first iteration and then they are persistently stored in the database for repeated crawling. The initial steps for extracting news include extracting RSS links from given domain. These RSS links are stored in the database to reduce HTTP requests in next iteration.

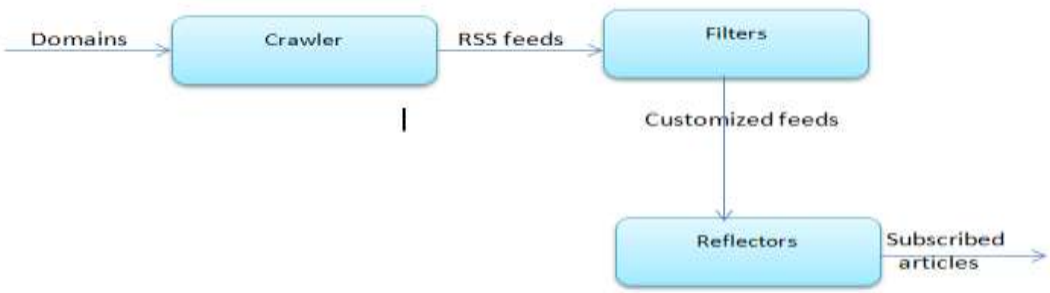

Fig. 1: Preliminary Design

It is clear that, having a central access point makes it significantly simpler to discover and access new content from a large number of diverse RSS sources. The initial step involved in this is to collect set of domain names or the website links required. Then, these domains are crawled for the valid RSS link. To get the RSS links from the domain we have to crawl the home page of the RSS links. This can be accomplished by extracting all the anchor tag and checking whether it is in valid RSS, ATOM, XML, RDF format. All advertisement-links are filtered appropriately by checking against the domain name. Certain news portal sites provide RSS service through third parties like feeds.feedburner.com so such links are not removed since they donot match domain name. Now these RSS links are stored in the database for future use in order to reduce crawling time and data usage. The RSS feed contents are mined for the user interest. Using the summary of the news articles, user interests are categorized accordingly using tf-idf (term frequency-inverse document frequency) model and the corresponding category is updated in the database. These RSS links are periodically checked in order to get the instant feeds. The news articles under each interest are processed to obtain the topics and recommended to those users. The topic modelling using Named Entity Recognition Technique is handled here to extract the crisp topics efficiently.

Now, the crawler will fetch RSS links from database and extract required details like title, description, published date, article link etc. and store it in the database. Next step involved in this process is categorising news article based on the content. A well trained categorizer will categorise the news into certain predefined categories and store it correspondingly. It contains some additional features like finding similar articles and providing recommendation to the user. The detailed architecture of the system is given in fig. 2.

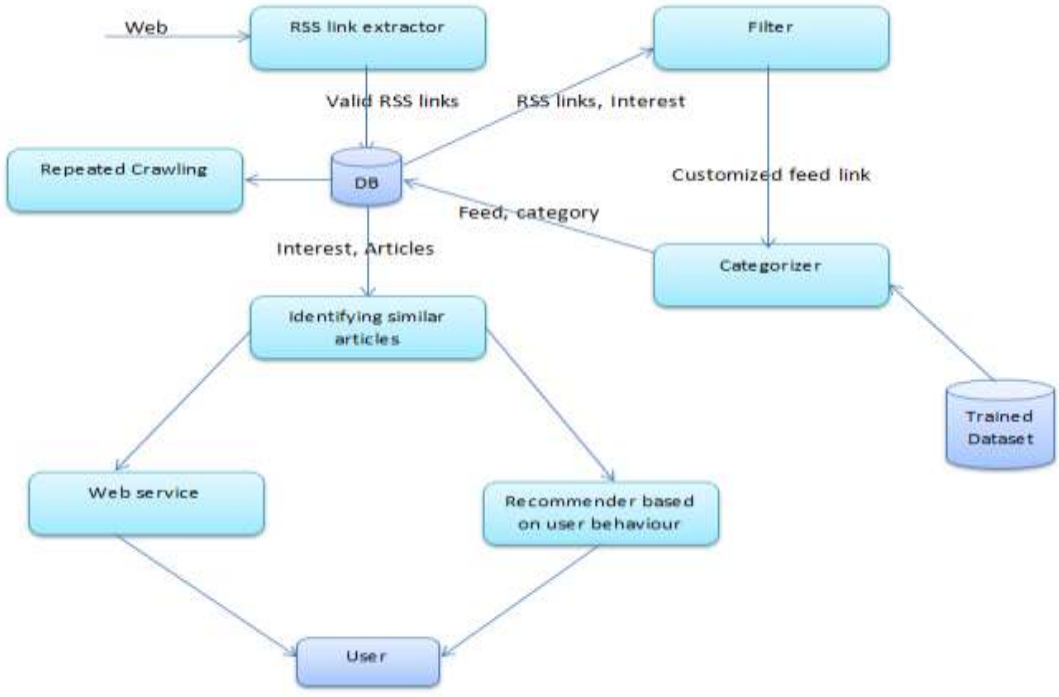

Fig. 2: Overall Architecture

3.3 Input-Output

The input and output for each of the component modules in the system is described in the table 1, given below.

The overall input for the whole system are the user interests and the news website domains. The expected output of the system are the personalized news preseneted to the user and the suggestions for the user based on the current recommendation and selection by the user. The input and the output for the individual components of the system can be comprehended from the Table 1.

Table 1: Input and output

MODULE	INPUT	OUTPUT
System	User interest and News Website Domains.	News articles and recommendation about other interest.
RSS link extractor	News Website Domains	Valid RSS from the given domain
Filter	User interest and RSS link	Required news articles
Categorization of News articles	User interest. News articles	Categorized articles
Identification of similar articles	News articles	Unique set of articles
Recommendation based on User behaviour	News articles and user interest	News articles and recommendation about other interest

3.4 Database Design

The entities of the system are identified and then they are normalized in order to reduce redundancy. In our system user, domain, interest news are identified as the entities. By normalizing these entities we get certain new entities like user interest which maintains user to interest mapping, interest domain which maintains interest and domain mapping, and rss which is used to reduce redundancy among the news fetched. The database of the system is designed in such a manner that the news articles are fetched in a timely manner without much noticeable delay and avoiding redundant news. The ER Diagram and cardinality ratio are shown in fig.3. Fig. 4 shows the ER Schema for categorization entities in the system.

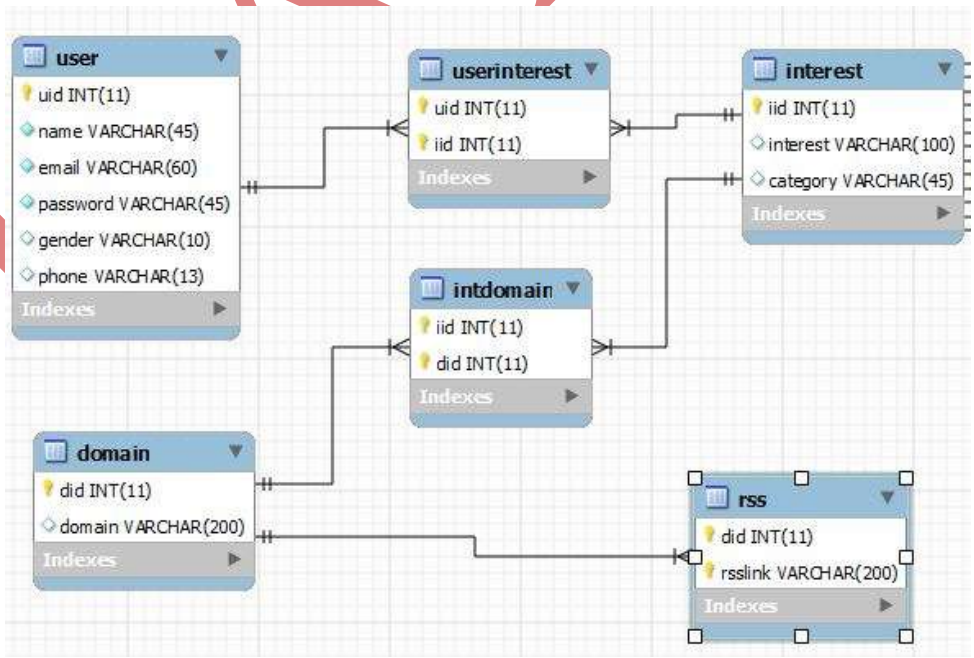

Fig. 3: ER Diagram-Overall Design

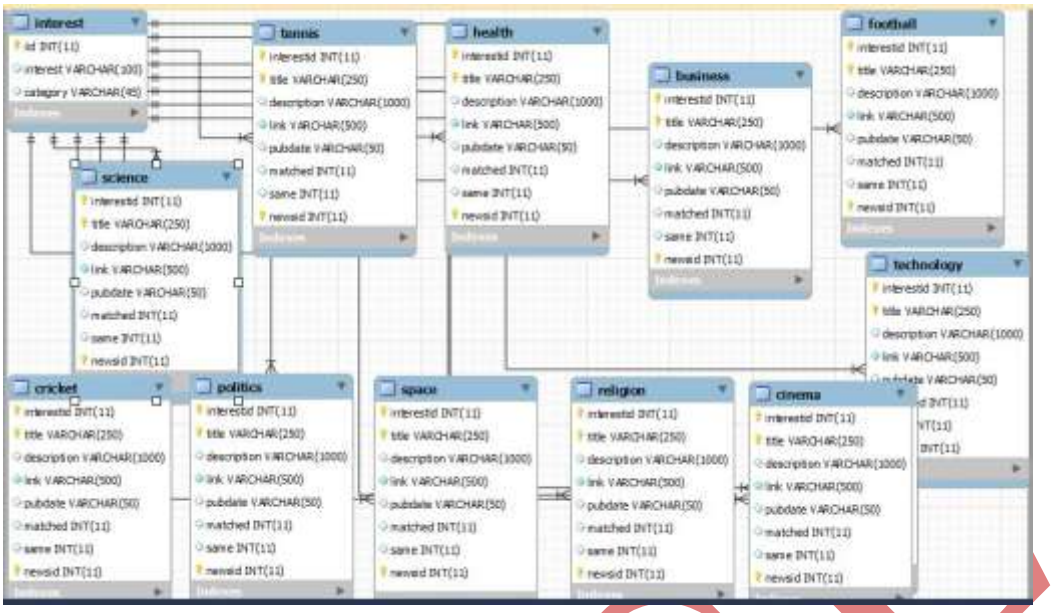

Fig. 4: ER Diagram-Category tables

IV NEWS RECOMMENDATION SYSTEM IMPLEMENTATION

The system is meant for providing users with the news articles which are closely corelated to the users' interests. The users are asked to input their interests once hey login to the system. The news is then grouped based on the interests of the users. The news which best matches the users will be recommended and displayed to the users. The news articles are getting updated frequently. These updates are to be taken into account and the users are to be presented with the latest news articles.

4.1. RSS Link Extractor

For the given News Website Domains this module will fetch all valid RSS link from the domain and store it in the database to avoid unnecessary crawling for the RSS links in future. The RSS feed link in
each given domain will be the output of this module. The main functionalities of this module is to crawl all links in the given domain and retrieve RSS links alone by eliminating ad links and other non-domain links. Two kinds of information are usually forwarded for download: (a) URL to XML file and (b) URL to plain HTML file which has to be downloaded. In parallel, the crawler examines the outputs of the crawled data and stores any information to the database. A crawler has to be adaptive on each URL that it is searching and the workload has to be distributed in order to access parallel a huge amount of data. It is expected that a crawler that is processing multiple RSS at the same time (parallelism) will be faster than a crawler that is accessing its feed URL in a serial manner, though we have to observe if the adaptation algorithmic procedure (file checking for duplicate entries and RSS changes) consumes too much time. In order to avoid unnecessary crawling the header packet is checked for last-modified date. If the last-modified date of the page is greater than the one stored in the database then the page will be crawled else it will move on to other RSS links in order to save time.

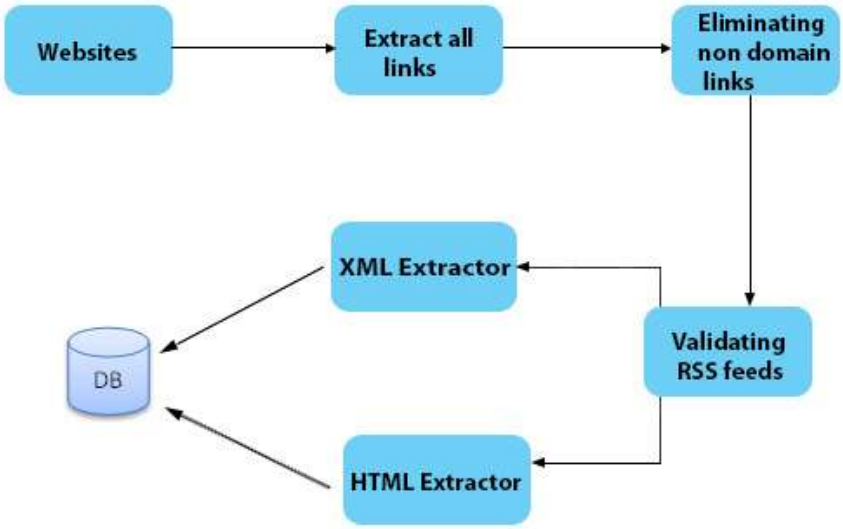

Fig. 5: RSS Link Extractor

4.2. Filter

Here fetched RSS link will be given as the input along with the interest. In this module, RSS link will be crawled and check against the given interest. Customised or Personalised News article summary along with its topic, last modified date, published date and time will be fetched. Finally the extracted data will be sent to the categorization module.It is shown in fig. 6.

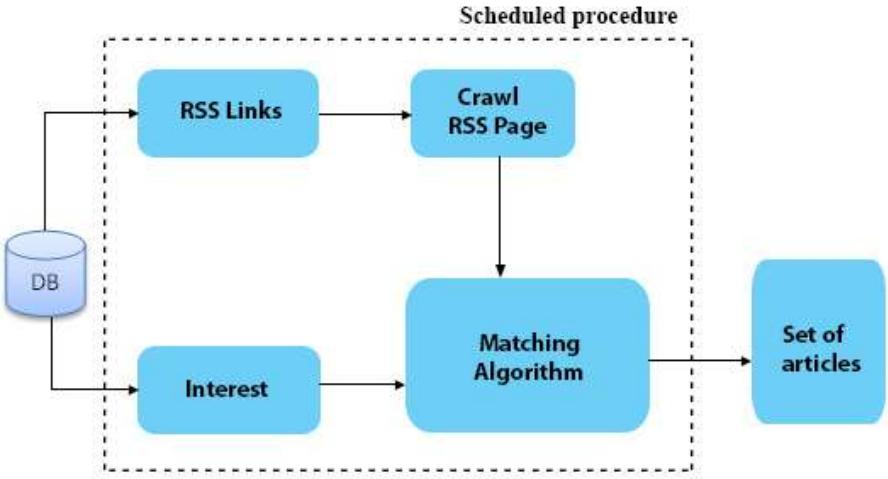

Fig. 6: Filter

4.3. Categorization of News Articles

In this module supervised machine leaning is used to categorize the news articles. The training phase involves pre-processing such as tokenization and stop-word removal. Then the topical terms are extracted via term extractor and fed into the corpus. Now these documents in the corpus are converted into sparse vectors (Bag-of-words). Then it is stored locally in hard-disk as Market Matrix Format[1]. During the testing phase, the news article contents are preprocessed same as in training phase and converted to sparse vectors. TF-IDF is applied on both the locally stored (.mm) file and pre-processed new article. TF-IDF[2] will give weightage to both frequently occurring terms and the least occurring terms. Then by comparing the sparse-matrix similarity the

article contents are fitted to one of its correct categories. The list of categories of news articles that were considered in the proposed personalized news recommender system are:

- Business
- Politics
- Cricket
- Football
- Tennis
- Other Sports
- Science and Technology
- Health
- Entertainment
- Space
- Religion
- Electronics

4.4. Identification of Similar News

In this module the articles under the same category and same interest are checked for its similarity and similar news are identified. Here the similarity is checked using Word Sense Disambiguation for its semantic role. The main functions involved are:

- Converting documents to semantic representation.
- Indexes documents in the vector representation, for faster retrieval
- For a given query document, return ids of the most similar documents from the index

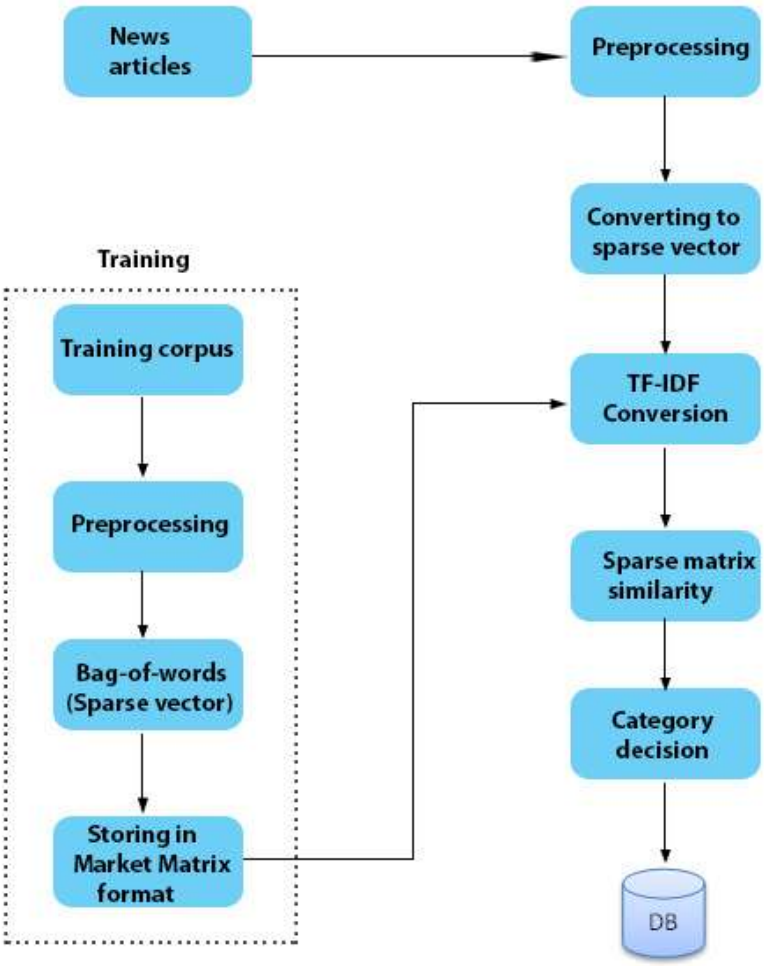

Fig. 6: Categorization of News article

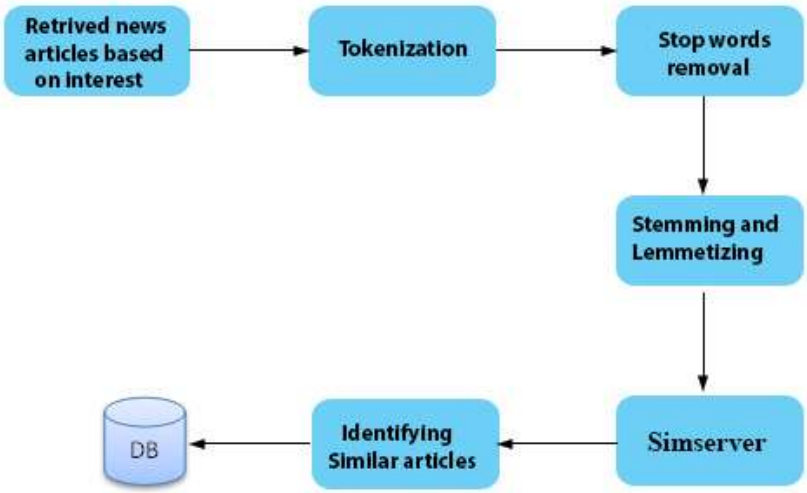

Fig. 7: Identification of similar news

4.5. Recommendation based on User Behaviour

Here news articles summary from the database will be retrieved using interest and the content is delivered to the appropriate user. Then the topics from the news articles are fetched using topic modelling and this topic will be suggested to the user. Based on the user behaviour such as time spent on reading the news articles, no of times the article is opened we narrow down their area of interest and can provide the personalized news to them.

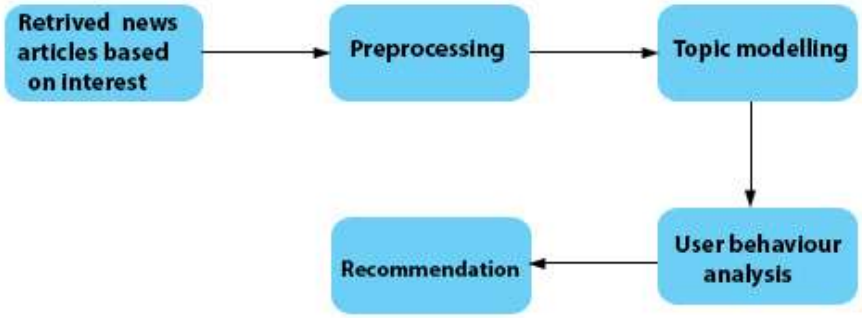

Fig. 8: Recommendation

4.6. User Interface Description

New user has to sign up on our Personalized News Portal. Then the user will enter the portal after being verified by their credentials. Initially the user has to enter his/her interest. If it is already existed, the news will be displayed. If it is a new interest, then it will take some time for displaying the news. The user can remove the interest whenever he/she wants. Then the best part here is, the user will be given recommendation based on the interests that he/she has subscribed earlier. The news will be provided to the user with news title, news description, source of the news such as www.espncricinfo.com, www.thehindu.com and the published time.

This system provides an attractive interface with many functions. The home page contains Login form with two fields user-name or email-id and password field. It also provides sign up form for new users. The news feed page contains an large column for presenting news feeds. This page also has a column that provides users to add new interests to his/her profile. Here, provision to delete an interest from the user profile is also provided. The news

feed page has column where user can select his/her interest to view news feeds.

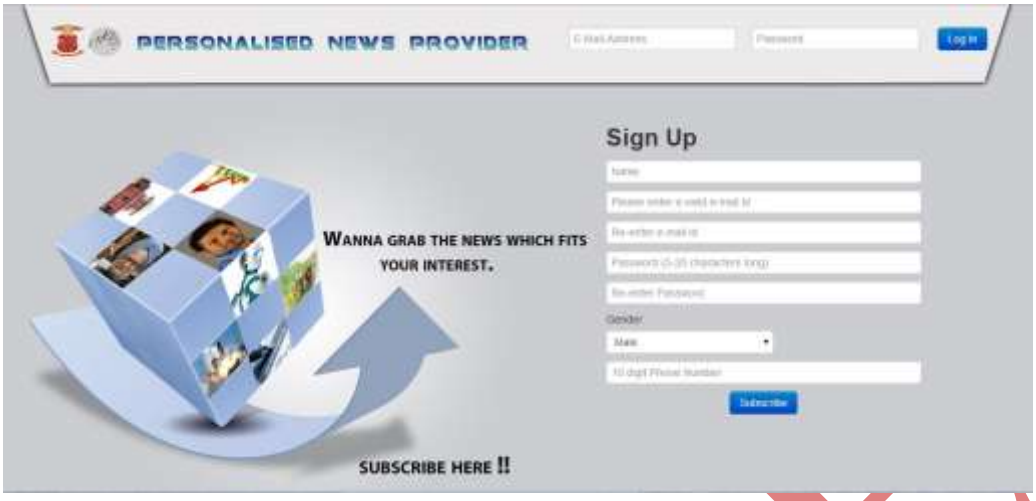

Fig. 9: Website

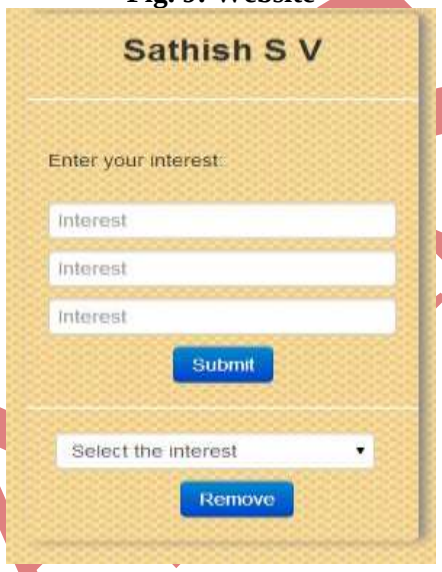

Fig. 9: Express User Interest

V EXPERIMENTAL EVALUATION AND ANALYSIS

An elaborated study and analysis was performed on the recommendation results obtained from the personalized news recommender system. The input datasets for the experiments were collected from http://qwone.com/~jason/20Newsgroups/ (20Newsgroups - Dataset) and http://www.thehindu.com (TheHindu archive). The detailed discussion of the results with the analysis is given in the next section.

5.1 Results

The ultimate output of the personalized news recommender system is an optimally recommended set of news articles to the user. Also, a set of recommended domains which are closely correlated with the user's interests and the recommendations are also suggested to the user. In the training phase, the news articles fetched are tokenized and stop-words are removed. Then, the topics are extracted via term extractor and fed into the corpus. Now these documents in the corpus are converted into sparse vectors. Then it is stored locally in hard-disk as Market Matrix Format. In the testing phase, the news article contents are preprocessed same as in training phase and converted to

sparse vectors. TF-IDF is applied on both the locally stored (.mm) file and pre-processed new article. Then by comparing the sparse-matrix similarity the article contents are fitted to one of its correct categories. The list of categories of news articles that were considered in the proposed personalized news recommender system are given in section 4.3. These categories refer to the domains of news articles that were considered in the system. Any topic which belongs to one among these categories can be specified by the user as his current interest. Fig. 10 shows the user intersts already input by the user and the suggestions to the user for new domains which may be related to his interests. The personalized news articles presented to the user based on his interests are shown in fig. 11.

Fig. 10: User Interest Displayed

Fig. 11: Personalized News

5.2 Performance Evaluation

The performance evaluation was done by considering the relevance of the news presented to the user based on his interests. A set of users were selected for our testing and they were asked to input their interests. The sytem was used to fetch news articles from various sources. Then, the users were asked to select news articles for reading from among the fetched articles. This was noted and for the same set of users, news articles were presented automaticlly by the system using our implemented method. Both the results were compared. Our system has shown accuracy upto 83%.

VI CONCLUSION

Due to the dynamism of the Web, crawlers form the backbone of applications that facilitate Web information retrieval. In this work, we described the architecture and implementation details of our crawling system. We explained the importance of extracting only content from web pages and how this can be implemented by our crawler and how we identify the similar articles and recommend the user based on their interests. In our mechanism the focus is put on the crawling news form each domain. A major open issue for future work is a detailed study of how the system could become even more distributed, retaining though quality of the content of the crawled pages.

VII ACKNOWLEDGEMENTS

The authors would like to express sincere gratitute to all our collegues and friends who have rendered their inevitable help for bringing this work a successful one.

REFERENCES

[1] “Bloglines”, http://www.bloglines.com, 2006.

[2] Jiming Liu Jia Zeng, William K. Cheung, Learning topic models by belief propagation, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 35, No. 5, May 2013.

[3] Andhyun-Kyu Cho Ka Cheung Sia, Junghoo Cho, Efficient monitoring algorithm for fast news alerts, IEEE Transactions On Knowledge And Data Engineering, Vol. 19, No. 7, 2007.

[4] Shang Gao-Robert Dew Ying Zhao Subrata Saha, Atul Sajjanhar, Delivering categorized news items using rss feeds and web services, Proc. 10th IEEE International Conference on Computer and Information Technology, 2010.

Biographical Notes

Mr. Madhu K. P. is presently pursuing Ph. D. in the Department of Computer Science and Engineering (Specialization in Location Based Recommender Systems), Anna University, Chennai, India.

Dr. D. Manjula is working as a Professor in the Department of Computer Science and Engineering, Anna University, Chennai, India.