Personalized News Categorization Through Scalable Text Classification

Ioannis Antonellis, Christos Bouras, and Vassilis Poulopoulos

Research Academic Computer Technology Institute N. Kazantzaki, University Campus,
GR-26500 Patras, Greece
Computer Engineering and Informatics Department, University of Patras,
GR-26500 Patras, Greece
{antonell, bouras, poulop}@ceid.upatras.gr
http://ru6.cti.gr

Abstract. Existing news portals on the WWW aim to provide users with numerous articles that are categorized into specific topics. Such a categorization procedure improves presentation of the information to the end-user. We further improve usability of these systems by presenting the architecture of a personalized news classification system that exploits user's awareness of a topic in order to classify the articles in a 'per-user' manner. The system's classification procedure bases upon a new text analysis and classification technique that represents documents using the vector space representation of their sentences. Traditional 'term-to-documents' matrix is replaced by a 'term-to-sentences' matrix that permits capturing more topic concepts of every document.

1 Introduction

Information that exists on the World Wide Web and the users that have access to it or produce it have reached outrageous numbers. This state is not static, but a dynamic continuingly changing condition that converts the Internet into a chaotic system. It is estimated that more than two billion pages exist at present while the number of the Internet users is uncountable. The consequence of the popularity of the Web as a global information system is that it is flooded with a large amount of data and information and hence, finding useful information on the Web is often a tedious and frustrating experience. The solution to finding information is search engines, but their main problem is that they search every corner of the Web and often the results, even to well defined queries, are hundreds of pages.

We focus on the needs of the Internet users who access news information from major or minor news portals. From a very brief search we found more than thirty portals that exist only in USA. This means that if one wants to find information regarding to a specific topic, he will have to search one by one, at least the major portals, and try to find the news of his preference. A better solution is to access every site and search for a specific topic if a search field exists in the portals. The problem becomes bigger for someone who would like to track a specific topic daily (or more times per day).

X. Zhou et al. (Eds.): APWeb 2006, LNCS 3841, pp. 391 – 401, 2006.

Classification of information into specific categories can give solution to some of the aforementioned issues. However, it is not possible to provide personalized results as standard classification procedure does not involve users' interests. All classification algorithms that have been proposed in the past, in order to achieve qualitative and efficient categorization of text, such as Naïve Bayesian method, support vector machines, decision trees and others, classify a document di to a category cj regardless of the target group that will use categorized results.

Many well-known systems try to solve this problem by creating rss feeds or personalized micro-sites where a user can add his own interests and watch the most recent and popular issues on them. The RSS feeds have become very popular and most of the news portals use them. But still, the problem is the filtering of information. Regarding the personalization issue, the attempts that have been made from the major search engines and portals include only the issue of viewing already categorized content according to the user's interests. This means that the user is not included into the classification procedure.

MyYahoo is a very representative example [12]. Following the login the user is empowered with functionality that helps to personalize the page. More specifically, the user can add his special interests on news issues by selecting general topics from a list. Every time the user accesses the web page, the more recent results on the topic are displayed. This procedure seems very helpful but it does not include the user into the classification and rating procedure. Another representative example is the service that is provided by the Google and more specifically the news service [9]. The page that appears is fully customizable and the user can add his own query to the appearing results but his choice is not included in the categorization mechanism but only to the rating mechanism of the entire web.

In this paper, the proposed news portal architecture bases upon scalable text classification, in order to include the user in the classification procedure. Without having prior knowledge of user's interests, the system is able to provide him articles that match his profile. The user specifies the level of his expertise on different topics and the system relies on a new text analysis technique in order to achieve scalable classification results. Articles are decomposed into the vector representation of their sentences and classification bases upon the similarity of the category vectors and the sentences vectors (instead of the document-article vectors). This procedure enables the system to capture articles that refer to several topics, while their general meaning is different.

The rest of the paper is structured as follows. Section 2 presents the general architecture of the system where the main feature is distribution of workload and modularity of the mechanism. Section 3 describes how personalization is implemented in our portal, in order to exploit user's awareness of a topic and further enhance the categorization procedure. A new text analysis technique is presented and analyzed and we introduce a new scalable classification algorithm that relies on this technique in order to provide personalized classification results. Section 4 refers to the role of the user to the core functionality of categorization. In section 5 experimental evaluation of our portal is presented and section 6 introduces some concluding remarks and issues about future work on the system.

2 General Architecture of the System

The system consists of distributed sub-systems that cooperate in order to provide end-user with categorized news articles from the web that meet his personal needs. The main features of the architecture are:

2.1 Modularity: Creating Autonomous Subsystems

The core mechanism of the system we created can be described as a general manager and a main database. This is the module where everything starts from and concludes to. The subsystems of the mechanism can work autonomously but the general manager is responsible for the cooperation of them.

As we can see from Figure 1 the whole system consists of a manager, a database system and seven subsystems.

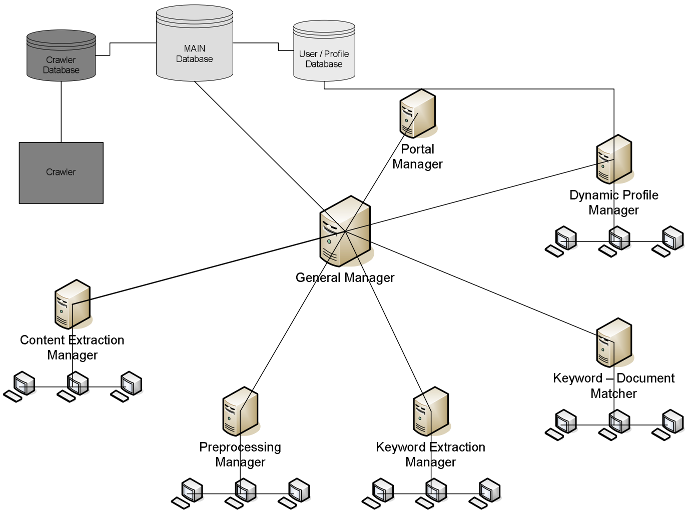


Fig. 1. General Architecture

The crawler sub-system is responsible for fetching web documents that contain useful news articles. Except from a standard crawler mechanism, it also maintains a list of RSS urls from many major portals. Content extraction manager uses the web components technique [5], [6] and some heuristics, in order to extract the text from the fetched web documents. Preprocessing manager, Keyword Extraction manager, Keyword – Document matcher and Dynamic Profile manager are implementing the Scalable Classification Algorithm that we introduce in Section 3.

2.2 Distributing the Procedure

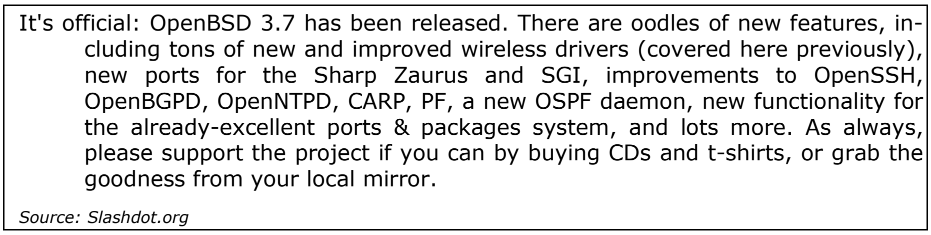
The procedure of retrieving, analyzing and categorizing content from the World Wide Web is sequential because each step needs the previous to be completed in order to start. This does not preserve the implementation of a distributed system for the completion of each step, but introduces a limitation that step number N+1 cannot be started if step N is not completed. This means that step N for the process on text X can be completed in parallel with step N for the process of text Y.

3 Personalizing the Portal

Presentation of the articles to the users must capture user-profile information in order to improve end-user results. Instead of treating this procedure as a standard text classification problem, we also consider dynamic changes of Web users' behavior and 'on-the-fly' definition of the category topics.

The main technique that our system exploits in order to provide personalized results is the use of scalable text classifiers instead of standard text classifiers. Scalable classifiers, permit the classification of an article into many different categories (multi-label classification). In addition, using the article decomposition that we present below (Section 3.1) we can exploit user's expertise in a category in order to relax or tighten a carefully selected similarity threshold and provide users with a wider or tighter set of answers.

Consider, for example, the text article of Figure 2 and Web users A and B. A is a journalism that needs information about Linux in order to write an article about open source software in general, while B is an experienced system administrator looking instructions on installing OpenBSD 3.6.



It's official: OpenBSD 3.7 has been released. There are oodles of new features, including tons of new and improved wireless drivers (covered here previously), new ports for the Sharp Zaurus and SGI, improvements to OpenSSH, OpenBGPD, OpenNTPD, CARP, PF, a new OSPF daemon, new functionality for the already-excellent ports & packages system, and lots more. As always, please support the project if you can by buying CDs and t-shirts, or grab the goodness from your local mirror.

Source: Slashdot.org

Fig. 2. Example News Article

A well-trained standard classification system would then provide the above document to both users, as it is clearly related to open source software and to OpenBSD operating system. However, it is obvious that although user A would need such a decision, it is useless for user B to come across this article.

Trying to investigate the cause of user's B disappointment, we see that standard text classification systems lack the ability to provide 'per-user' results. However, user's knowledge of a topic should be taken into account while providing him with the results. It is more possible that a user who is aware of a category (e.g. user B knows a lot about Linux) would need less and more precise results, while non-expert users (such as the journalism) will be satisfied with a variety of results.

Scalable text classification problem can be seen as a variant of the classical classification where many similarity classes are introduced and permit different, multi-label classification results depending on the similarity class.

Definition 1. (Scalable Text Classification) let $C = \left\{c_1 \ ,\dots,c_{|C|}\right\}$ a set of growing set of categories and $D = \left\{d_1,\dots,d_{|D|}\right\}$ a growing set of documents. A scalable text classifier that defines p similarity classes is a function $\Phi = D \times C \to \Re^p$.

It follows from Definition 1 that given an initial test set of k training data (text documents) TrD = {trd1, trd2, …, trdk} already classified into specific m training categories from a well-defined domain TrC = {trc1, trc2, …, trcm}, the scalable text classifier is a function that not only maps new text documents to a member of the TrC set using the training data information but also:

- Defines p similarity classes and p corresponding similarity functions that map a document into a specific category c. Similarity classes can be shown as different ways to interpret the general meaning (concept) of a text document.
- Permits the classification of each document into different categories depending on the similarity class that is used.
- Permits the definition of new members and the erasure of existing ones from the categories set. That means that the initial set TrC could be transformed into a newly defined set C with or without all the original members, as well as new ones.

4 Text Analysis Using Document Decomposition into Its Sentences

Having the vector space representation of a document, it is clear that we have no information on how such a vector has been constructed, as it can be decomposed in infinite ways into a number of components.

Definition 2. (Document Decomposition into Sentences) Let $\vec{d}_i = \left[v_1, v_2, \dots, v_k\right]$ the vector representation of a document $\vec{d}_i$. A document decomposition into its sentences is a decomposition of vector $\vec{d}_i$ of the form $\vec{d}_i = \vec{s}_1 + \vec{s}_2 + \ldots + \vec{s}_n$, where component $\vec{s}_k$ is a vector $\vec{s}_k = \left[v'_1, v'_2, \dots, v'_{|s_k|}\right]$ representing k-th sentence of document.

Using a decomposition that Definition 2 provides us, we can therefore compute the standard cosine similarity using Equation 1. A modified version of a 'term-to-document' matrix, that we call it 'term-to-sentences' matrix can also be used to include information about the sentences decomposition. Figure 3 provides an example.

$$\cos\left(\vec{d}_i, \vec{c}_j\right) = \frac{\vec{d}_i \cdot \vec{c}_j}{\left\|\vec{d}\right\| \left\|\vec{c}_j\right\|} = \frac{\sum_{k=1}^{n} \vec{s}_k \cdot \vec{c}_j}{\left\|\sum_{k=1}^{n} \vec{s}_k\right\| \left\|\vec{c}_j\right\|} \tag{1}$$

	D_1	D_2							…	D_n
			s_1	s_2	s_3	s_4	…	s_k		
t_1		a_1	a_{11}	a_{12}	a_{13}	a_{14}	…	a_{1k}		
t_2		a_2	a_{21}	a_{22}	a_{23}	a_{24}	…	a_{2k}		
t_3		a_3	a_{31}	a_{32}	a_{33}	a_{34}	…	a_{3k}		
t_4		a_4	a_{41}	a_{42}	a_{43}	a_{44}	…	a_{4k}		
t_5		a_5	a_{51}	a_{52}	a_{53}	a_{54}	…	a_{5k}		
t_6		a_6	a_{61}	a_{62}	a_{63}	a_{64}	…	a_{6k}		
t_7		a_7	a_{71}	a_{72}	a_{73}	a_{74}	…	a_{7k}		
t_8		a_8	a_{81}	a_{82}	a_{83}	a_{84}	…	a_{8k}		
t_9		a_9	a_{91}	a_{92}	a_{93}	a_{94}	…	a_{9k}		
…		…	…	…	…	…	…			
t_m		a_m	a_{m1}	a_{m2}	a_{m3}	a_{m4}	…	a_{mk}		

Fig. 3. Example 'term-to-sentences' matrix, with term to sentences analysis of a specific document. Values a_{ij} satisfy equation: $a_i = \sum_{j=1}^{k} a_{ij}, \forall 1 \le i \le n$.

5 Scalable Classification Algorithm

The most useful characteristic of the proposed classification algorithm is its scalability feature. A text document can be classified into many different categories depending on the similarity of the semantic representation of its sentences with the categories. Exploiting user's level of expertise in a specific area, we can relax or tighten a similarity threshold of the distance between a specific number of sentences of an article and some categories, in order to allow classification of the article in many categories. Formal definition of the Training Phase of the Scalable classification algorithm is shown in Figure 4:

Training Phase

1) Decompose labeled text documents into their sentences
2) Compute term to sentences matrix of every category using some indexing method
3) Compute category vectors by combining the columns of the corresponding term to sentences matrix
4) Estimate categories similarity threshold, by computing the cosines of the angles between the different category vectors of step 3
5) For each category, estimate sentences similarity threshold by computing the cosines of the angles between all sentence vectors with the corresponding category vector

Fig. 4. Training Phase of the Scalable Classification Algorithm

Main characteristics of the classification phase (Figure 5) include (a) the ability to adjust the number of sentences k that must much a sentences similarity threshold in order to classify the corresponding document to a category and (b) the feedback that the algorithm implicitly takes in order to re-compute categories vectors and therefore capture semantic changes of the meaning of a topic as time (arrival of new text documents) passes.

```
Classification Phase
1) Decompose unlabeled text document into its sentences
2) Compute term to sentences matrix of the document
3) Compute document vector by combining the columns of the term to sentences matrix
4) Estimate similarity (cosine) of the document vector with the category vectors computed at step 3 of Training Phase. If cosine matches a similarity threshold computed at step 4 of Training Phase classify the  document to the corresponding category
5) Estimate similarity (cosines) of each sentence with the category vectors computed at step 3 of Training Phase
6) If a cosine matches a similarity threshold computed at step 5 of Training Phase classify the document to the corresponding category (allowing scalable multi-category document classification)
7) The category vector computed during step 3 of Training Phase is re-computed based on the newly acquired data after the classification of the unlabeled text document to categories matching the threshold criterion
```

Fig. 5. Classification Phase of the Scalable Classification Algorithm

It is important to mention that the procedure of 'estimation of similarity' involved in many steps of our algorithm, can be implemented using a variety of techniques such as (a) simple cosine computation, (b) latent semantic analysis of the matrix so as to produce its low rank approximation and then compute the similarity [3, 4, 8, 13, 15] (c) other low rank approximation of the matrix that either use randomized techniques to approximate the SVD of the matrix [1, 2, 8, 9] or use partial SVD on cluster blocks of the matrix and then recombine it to achieve fast and accurate matrix approximation.

6 Scalability as Personalization

Users of the system select the level of their expertise on different categories. Using this information, the core mechanism of the system that implements the Scalable Classification Algorithm changes the number k of sentences (according to Table 1) that should match the threshold criterion of a category in order to be classified.

Table 1. Configuration of number of sentences that much the threshold criterion vs user expertise

k (number of sentences)	User expertise
1	low
2	medium
3	high

7 Experimental Evaluation

Experimental evaluation involves two main steps. Firstly, we analyze the performance of the Scalable Classification algorithm, using a well known dataset [7]. Using data gathered during this procedure, we also specify different criterion thresholds and apply them to the core mechanism of the presented system. At last, experimental results of the real articles' classification are presented.

In order to evaluate our scalable classification technique we used the 20 newsgroup dataset [7], which is a widely used dataset in the evaluation process of many classification algorithms (both supervised and unsupervised).

The 20 newsgroup dataset is a collection of articles of 20 newsgroups. Each category contains 1000 articles. We preprocessed the documents so as to use only the main text (as Subject section may contain many keywords of the corresponding category). In order to evaluate the similarity values between different category vectors we used the standard metric [12] that computes the cosine of the corresponding vectors aj and q using Formula 2.

$$\cos\theta_j = \frac{a_j^T q}{\| a_j \|_2 \| q \|_2} = \frac{\sum_{i=1}^{t} a_{ij} q_i}{\sqrt{\sum_{i=1}^{t} a_{ij}^2}\sqrt{\sum_{i=1}^{t} q_i^2}} \qquad (1)$$

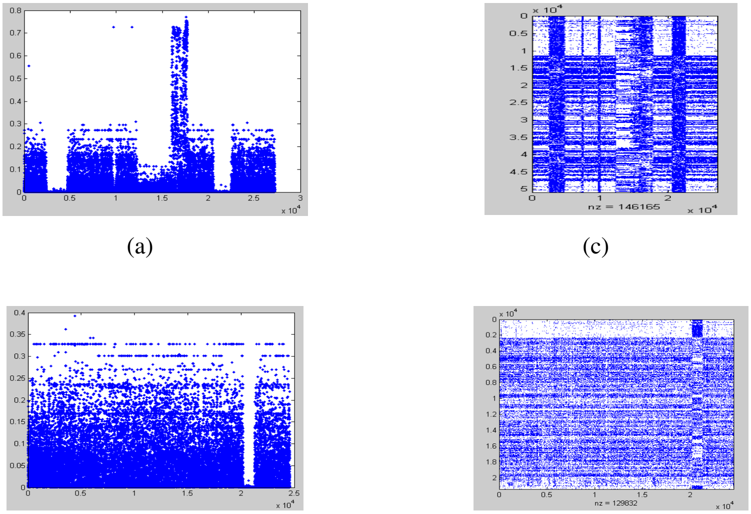
Below, we present evaluation of the similarity thresholds obtained for the 'sentence vs. category' using the 20 newsgroup dataset. All experiments were conducted using data collected using both the Rainbow tool [16] for statistical analysis and separation procedures of the datasets, as well as using the TMG [17] a recently developed MATLAB toolbox for the construction of term document matrices from text collections.

Comparing the twenty category vectors it turns up that different category vectors create a minimum angle of 19.43 degrees and a maximum angle of 53.80 degrees. It is also easily seen that semantically different categories create large enough angles (e.g. alt.atheism and comp.os.ms-windows.misc create and angle of 42.71 deggres) while semantically close categories create smaller angles (e.g. talk.religion.misc and alt.atheism create an angle of 19.44 degrees). That means that a 'category vs. category' threshold can be estimated to an angle 19.43 degrees with a corresponding similarity value of 0.94.

Figure 6 presents the sentence vs. category vectors similarities for different categories of the 20 newsgroup dataset. The basic results can be summarized as:

- General categories (like alt.atheism or soc.religion.christian) have a dense uniform allocation of similarities in the range [0-0.1] and a sparse uniform allocation in the range [0.1 – 0.5]
- Well structured categories seem to be indicated from a uniform sentence vs. category similarity chart

Trying to investigate on an easy way to identify general categories and proceed on further separation, non-well structured categories seem to reside on 'term to sentence'



(a)

(c)

(b)

(d)

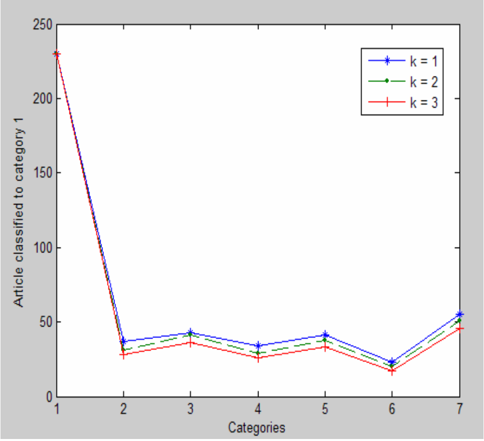
Fig. 6. Sentence vs category vectors for different categories of the 20-newsgroup dataset (first line) with the corresponding 'term-to-sentences' matrix using function spy of MATLAB (second line) (a) comp.os.ms-windows.misc (b) comp.windows.x

matrices that have a blocked structure. Figure 6 provides a visualization of the matrix elements of the 'term to sentence' matrix where large values are identified by intense color. Figures of categories that were identified as not well structured in the previous Section are shown to have a matrix with blocked structure (e.g. (c) or (d) matrices).

8 System Evaluation

Using the similarity threshold of 19.43 degrees that we computed using the 20 newsgroup dataset, we tuned the core mechanism of the system that uses the Scalable Classification Algorithm so as to classify an article into a category if k sentences of the article much this criterion. Figure 7 shows how many business articles are also classified to other categories for three values of k. As value of k increases, the amount of multi-labeled articles decreases.

We also, tested the classification feedback that our Scalable Classification Algorithm provides. Figure 8, reports the maximum and the minimum angle between the different category vectors, as time passes and newly classified articles affect the category vectors. We run the system for a period of 15 days and we computed the angles between the re-computed category vectors at the end of every day. It is easily seen that minimum angles vary close to 20 degrees, while maximum angles are close to 40 degrees.



	Category
1	Business
2	Education
3	Entertainment
4	Health
5	Politics
6	Sports
7	Technology

Fig. 7. Multi-labeled business articles for different values of k (number of sentences to much the threshold criterion)

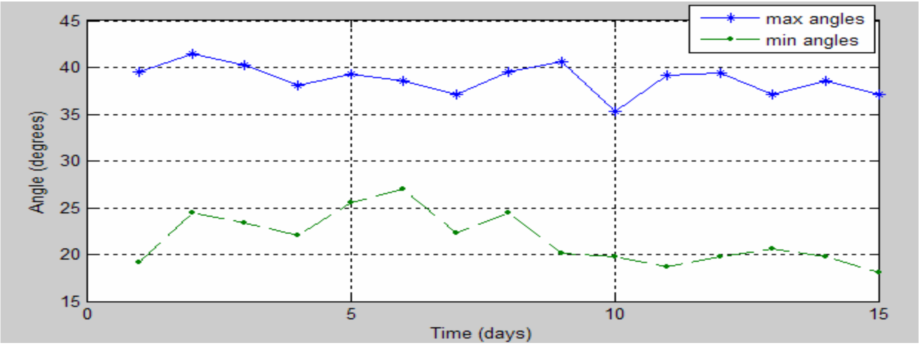


Fig. 8. Maximum and Minimum angles between category vectors, for a period of 15 days. Classification feedback of our algorithm results in small variances of the vectors that represent each category.

9 Conclusions and Future Work

In this paper, we propose a new technique for personalized article classification that exploits user's awareness of a topic in order to classify articles in a 'per-user' manner. Furthermore, the architecture of the backend of a portal that uses this technique is presented and analyzed. Unlike standard techniques for personalization, user only specifies his level of expertise on different categories. The core of the system relies on a new text analysis and classification method that decomposes text documents on their sentences in order to capture more topic concepts of every document.

For future work, we will further explore the classification of real articles using our system. It will be interesting to apply data mining techniques on data deriving from the amount of multi-labeled documents and try to identify the behavior and impact of major 'alarm news'. The scalable classification algorithm is also of independent interest and we intend to study theoretically its performance.

Acknowledgements

Ioannis Antonellis's work was partially supported by a Zosima Foundation Scholarship under grant 1039944/ 891/ B0011 /21-04-2003 (joint decision of Ministry of Education and Ministry of Economics)

References

1. D. Achlioptas, F. McSherry, Fast Computation of Low Rank Matrix Approximations, STOC '01 ACM.
2. Y. Azar, A. Fiat, A. Karlin, F. McSherry, J. Saia, Data mining through spectral analysis, STOC '01 ACM.
3. M.W. Berry, S.T. Dumais & G.W. O' Brien, Using Linear Algebra for Intelligent Information Retrieval, UT-CS-94-270, Technical Report.
4. M. W. Berry, Z. Drmac, E. R. Jessup, Matrices, Vector Spaces, and Information Retrieval, SIAM Review Vol. 41, No 2 pp 335-362.
5. C. Bouras, V. Kapoulas, I. Misedakis, A Web - page fragmentation technique for personalized browsing, 19th ACM Symposium on Applied Computing - Track on Internet Data Management, Nicosia, Cyprus, March 14 - 17 2004, pp. 1146 – 1147.
6. C. Bouras and A. Konidaris, Web Components: A Concept for Improving Personalization and Reducing User Perceived Latency on the World Wide Web, Proceedings of the 2nd International Conference on Internet Computing (IC2001), Las Vegas, Nevada, USA, June 2001, Vol. 2, pp.238-244.
7. CMU Text Learning Group Data Archives, 20 newsgroup dataset, http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/news20.html.
8. P. Drineas, R. Kannan, A. Frieze, S. Vempala, V. Vinay, Clustering of large graphs via the singular value decomposition, Machine Learning 56 (2004), 9-33.
9. P. Drineas, R. Kannan, M. Mahoney, Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix, Tech.Report TR-1270, Yale University, Department of Computer Science, February 2004.
10. S. Dumais, G. Furnas, T. Landauer, Indexing by Semantic Analysis, SIAM.
11. Google News Service, http://news.google.com
12. W. Jones and G. Furnas, Pictuers of relevance: A geometric analysis of similarity measures, J. American Society for Information Science, 38 (1987), pp. 420-442.
13. T. K. Landauer, P. W. Foltz, D. Laham (1998). Introduction to Latent Semantic Analysis. Discourse Processes, 25, pp. 259-284.
14. My Yahoo!, http://my.yahoo.com
15. C. H. Papadimitriou, P. Raghavan, H. Tamaki, S. Vempala, Latent semantic indexing: A probabilistic analysis, 17th Annual Symposium on Principles of Database Systems (Seattle, WA, 1998), 1998, PP 159-168.
16. Rainbow, statistical text classifier, http://www-2.cs.cmu.edu/~mccallum/bow/rainbow/.
17. D. Zeimpekis, E. Gallopoulos, Design of a MATLAB toolbox for term-document matrix generation, Proceedings of the Workshop on Clustering High Dimensional Data, SIAM 2005 (to appear).