LUCI: A Personalization Documentary System Based on the Analysis of the History of the User's Actions

R. AREZKI[1], A.MOKRANE[1], G.DRAY[1] P.PONCELET[1], D.W.PEARSON[2]
[1] Centre LGI2P EMA, Site EERIE Parc Scientifique Georges Besse
30035 Nimes Cedex 1, France
rachid.arezki, abdenour.mokrane, gerard.dray, pascal.poncelet@ema.fr
[2] IUT de Roanne, 20 Avenue de Paris 42334 Roanne, France
david.pearson@univ-st-etienne.fr

Abstract. With the development of Internet and storage devices, online document servers abound with enormous quantities of documents from various themes. The online search of pertinent documents is a fastidious task and "search engine" may overcome this difficulty. However, in such engines, each new document need must be formulated by a new request. Recently, new approaches were proposed to solve this problem by taking into account the user profile. However, these approaches don't consider the evolution in time of the document classes consulted by the user. In this paper, we propose a new approach for learning the user long-term profile for textual document filtering . In this framework, the documents consulted by the user are classified in a dynamic way, then we analyze the distribution in the time of the document classes. The approach aim is to determine, as well as possible, the document classes which interest the user. We also propose a system called *LUCI*, which allows an online document's personalization by using this approach. An empirical study confirms the relevance of our approach.
Keywords. User Modeling, Information Filtering, Long-Term Adaptation and Filtering

1 Introduction

With the development of Internet and storage devices, online document servers abound with enormous quantities of documents from various themes. The online search of pertinent documents is a fastidious task and "search engines" may overcome this difficulty. However, in such engines, each new document need must be formulated by a new request. Recently, considerable efforts have been deployed to overcome this weakness by developing systems able to propose adapted documents to a user without requests formulation. These systems take into account the user profile to propose personalized documents. However, even if documentary filtering systems based on the user profile exist, they do not take into account the evolution in the time of documents classes consulted by the user. In this paper, we propose a new approach for user long-term profile learning for textual documents filtering. It is based on the analysis of the evolution in time of the document classes consulted by the user. Indeed, the documents consulted by the user are classified in a dynamic way, then we analyze the distribution in the time of the documents classes. The approach aim is to determine, as well as possible, the document classes which interest the user. It is achieved by giving more importance to the regularly consulted document classes than those concentrated over short periods. Our approach does not require that the user provides explicitly information to the system.

We have developed a system, called *LUCI*, allowing an online document personalization by using our approach. *LUCI* learns the user profile via the consulted documents and proposes documents dynamically, i.e each time the user modifies its profile (new consulted document), the system proposes to him documents adapted to its new profile.

The article is organized as follows: Section 2 presents the problem of modeling the user long-term interests for a personalized filtering of textual documents. Section 3 describes the functional architecture of the system *LUCI*. Then, Section 4 details our approach for document filtering, based on the distribution in time of the document classes consulted by the user. Section 5 presents the procesus of documentary filtering. The experimental results, on a documentary corpus of reference, are presented in Section 6, we show that our approach allows to detect and to propose documents of regular interest. Section 7 presents briefly the related work on the user modeling. Finally, Section 8 provides some concluding remarks and directions of future research.

2 Problematic

The objective of our proposal is the learning of users long-term profiles for an effective textual document personalization. To achieve this goal, we classify the consulted documents and we analyze the evolution in time of the classes of these documents. The general idea considers that the document classes regularly consulted have more long-term interest than those concentrated over a short period. For example, a user having shares consults everyday one or two documents on finance. However, this same user gave a talk on dinosaurs, and has consulted more than 100 documents in one day on the dinosaurs. In the long-term (one month for example), the finance theme has more interest, to the user, than the dinosaur one, whereas the number of documents consulted on the dinosaurs is much more significant than the number of documents consulted on finances. In fact, the first thematic represents a regular interest whereas the second one is just a momentary need, i.e. concentrated in a short period. In the long-term (one month in this example) we must be able to propose documents of the thematic finances, because it is a regular interest, and not to propose documents of the thematic dinosaur because it is a spontaneous interest.

In our context, the user is connected to an online documentation base which contains a collection of documents on various themes. The documentary corpus is indexed and each document is associated to a vector. The user is modeled by a set of documents (vectors) consulted through time. The problem of long-term document personalization is to find a model able to take into account the evolution in time of the document classes consulted by the user, and as well as possible to detect the classes of regular interests for a better personalization of documents.

3 LUCI Architecture

The goal of the *LUCI* system is to learn dynamically the user profile and to propose documents adapted to his long-term profile. *LUCI* models the user through the consulted documents, and accordingly it proposes to him new ones . As shown in figure 1, *LUCI* is composed of two sub-systems and a structure defining the user model:

1. *User modeling sub-system:* models the user under his actions. Indeed, it supervises the user's actions by updating the model when a document is consulted.
2. *Filtering documentary sub-system:* proposes documents according to the user model, and when this last one is updated (i.e document is consulted), a new document collection is proposed.
3. *The user model:* is a complex structure which models the user long-term interests

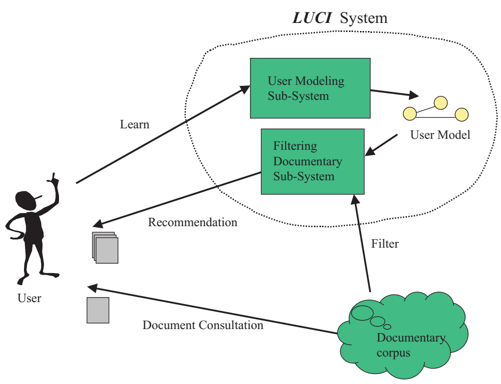

Fig. 1. Functional Architecture

There are many vectorial models to represent documents [Cha90,DDL$^+$90]. In our context, we consider that the documents are represented by the traditional vectorial model [Sal71,SG83].

The extraction of terms (key words) is done by:

– *Identification of terms (lexical segmentation);*

- *Elimination of the terms which are not interesting (by using a preset list);*
- *The reduction of the terms in their root;* [1]
- *The choice of the most frequent terms (10% of the Existing key words).*

To reduce the advantage of the long documents on the short documents, The used keyword vectors are normalized [SSMB95,Sin97]. Let D be a text document, then $\{(t_1, w_1), ..(t_n, w_n)\}$ is the feature vector of D where t_i is a term (word) that occurs in D and w_i is the weight of term t_i. The used keyword vectors are normalized as follow:

$$w_i = \frac{f_i}{\sum_{j=1}^{n} f_j}$$

f_i is the frequency of the term t_i.

We use, to measure the similarity, the *cosine* formula [SG83]: let D_i, D_j be two documents, the similarity of the *cosine* between these two documents is formulated by:

$$SIM(D_i, D_j) = (D_i * D_j)/(\mid D_i \mid^{1/2} * \mid D_j \mid^{1/2})$$

4 Modeling Long-Term Interests

In this section, we describe our approach to model the long-term interest. Our motivation behind this modeling is to capture, as well as possible, the general user interest. Hence, we consider that a class regularly consulted (permanent interest) has more interest than a class concentrated in a short period (spontaneous "or momentary" interest).
The approach is based on:

1. Dynamic classification of the documents consulted by the user;
2. Associating a vector with each class, which corresponds to the sum of its document vectors;
3. Associating a number with each class, which is its document distribution in the time;
4. Using the vectors of classes and their distribution, a vector named LTV (Long-Term Vector), determines the user long term interest is calculated;
5. Proposing documents to the user is done by the calculation of similarity between documents of the corpus and the vector LTV.

In the following of this section, we will start by describing in detail the structure of the long term model. Then, we will describe the method of the dynamic classification of the documents, followed by the calculation method of the document distribution of a class. Finally, we presente the calculation of the LTV vector.

[1] form truncated (stem or lemma)

4.1 Structure of the Long Term-Model

A user is defined by a tuple $X =< id, S >$ where id stands for an unique identifier and S is the set of document classes consulted by the user id, i.e $S = \{C_1, ...C_n\}$, where n stands for number of classes.

A class C_i, i=1 to n, is defined by a tuple $C_i = \{V, Dist_{C_i}, V_{C_i}\}$, such that:

1. V: is a set of tuples which contain the consulted documents of the class C_i (vectors) and their positions (in time).

$$\{(V_1^{C_i}, Pos_1)..(V_{\|C_i\|}^{C_i}, Pos_{\|C_i\|})\}$$

   $\|C_i\|$ is the number of documents of the class C_i
   $V_j^{C_i}$: the J^{th} document of the class C_i,
   The document position Pos_i stands for its appearance in the time order, over all the consulted documents.
2. $Dist_{C_i}$: is the distribution in time of the documents of the class C_i (see section 4.3),
3. V_{C_i}: the vector of the class C_i, with :

$$V_{C_i} = \sum_{j=1}^{\|C_i\|} V_j^{C_i}$$

Now we will examine how the classes are determined and how the documents are associated to these classes.

4.2 Classification of the Consulted Documents by the User

The algorithm below describes the process of classification of the consulted documents. The creation of the classes is done in a dynamic way. Initially, the number of classes is set to zero ($S = \emptyset$). With the first consulted document, the algorithm creates the class C_1. Then for each consulted document, the algorithm calculates the similarity between the document and the whole of the existing classes. If the document is close to one of the existing classes then the document is added to this class. Otherwise, the algorithm creates a new class and it associates this document to him.

The algorithm receives in entry a new document, represented by a vector V_D and the user model $X =< id, S >$. At output, it associates the document V_D and its position to a class.

Algorithm 1: Algorithm of dynamique classification
Input: Document V_D
Pos_{V_D}: position of the document V_D
the user model $X =< id, S >$,
Output: association of V_D to a class of documents

```
begin
  if ||S|| = 0 then
    create a class C_1
    Add (V_D, Pos_{V_D}) to the class C_1
  else
    k ← 1
    for j = 1 to j = ||S|| do
      if SIM(V_{C_j}, V_D) > SIM(V_{C_k}, V_D) then
        k ← j
    if SIM(V_{C_k}, V_D) > α then
      Add (V_D, Pos_{V_D}) to the class C_k
    else
      create a new class C_{||S||+1}
      Add (V_D, Pos_{V_D}) to the class C_{||S||+1}
end
```

$\|S\|$: represent the number of classes contained in the user model,
SIM: similarity function between two vectors,
α : represent the threshold of similarity.

4.3 Calculation of the Document Distribution Classes

To calculate the distribution in time of the consulted document classes, to each class, we associate a group of dots represented in a two dimensional space. Each point represent a document of the class. A document is represented by two cordinates:

1. its on all the documents, i.e. its consultation order in time,
2. its position of the document compared to the documents of its class, i.e. its isertion order in its class.

$$Dist_{C_i} = \frac{Pos_{\|C_i\|} - Pos_1 + 1}{N} * \frac{1}{1 + \sum_{j=1}^{j=\|C_i\|} Distance(D_j, \Delta)}$$

$Dist_{C_i}$: distribution of the class C_i,
$Pos_{\|C_i\|}$: the position of the last consulted document of the class C_i, compared to all the consulted documents,
Pos_1 : the position of the first consulted document of the class C_i, compared to the all the consulted documents,

Δ : the regression line of least squares of the group of dots of the class,
D_j : co-ordinates of the j^{th} document of the class C_i,
N : the number of consulted documents,
$Distance(D_j, \Delta)$: distance between the point D_j and the line Δ.

4.4 Calculation of the Long-Term Vector *LTV*

From the user's long-term model, a vector defining his long-term interest, is calculated.

$$LTV = \sum_{i=1}^{\|S\|} Dist_{Ci} * V_{Ci}$$

$\|S\|$ is the number of classes

The proposal of documents to the user relating to his long-term interest is done by the calculation of similarities between the documents of the corpus and the vector LTV. If the similarity is greater than given threshold, it recommends the document to the user.

4.5 Example

Suppose that a user have consulted a succession of documents: $D_1, D_2, .., D_9$

At the 9^{th} consulted document, the dynamic algorithm of classification will form 3 classes:
$C_1 = \{D_1, D_5, D_9\}$
$C_2 = \{D_2, D_3, D_4\}$
$C_3 = \{D_6, D_7, D_8\}$
The user model is defined by $X =< id, S >$, such as:
$S = \{C_1, C_2, C_3\}$ where:

1. $C_1 = \{V, Rep_{C_1}, V_{C_1}\}$
   $V = \{(D_1, 1), (D_5, 5), (D_9, 9)\}$
   $Rep_{C_1} = \frac{9-1+1}{9} \star \frac{1}{1+0} = 1$
   $V_{C_1} = D_1 + D_5 + D_9$
2. $C_2 = \{V, Rep_{C_2}, V_{C_2}\}$
   $V = \{(D_2, 2), (D_3, 3), (D_4, 4)\}$
   $Rep_{C_2} = \frac{4-2+1}{9} \star \frac{1}{1+0} = \frac{3}{9}$
   $V_{C_2} = D_2 + D_3 + D_4$
3. $C_3 = \{V, Rep_{C_3}, V_{C_3}\}$
   $V = \{(D_6, 6), (D_7, 7), (D_8, 8)\}$
   $Rep_{C_3} = \frac{8-6+1}{9} \star \frac{1}{1+0} = \frac{3}{9}$
   $V_{C_3} = D_6 + D_7 + D_8$

$LTV = (1 \star V_{C_1}) + (\frac{3}{9} \star V_{C_2}) + (\frac{3}{9} \star V_{C_3})$

It is noticed that the number of documents of the classes C_1, C_2, C_3 is identical. However at the calculation of vector LTV the C_1 class is favoured, because it is a regular interest, contrary to the classes C_2 and C_3 which are spontaneous interest.

5 Textual Documents Filtering

The proposal of documents to the user is done by the calculation of similarity between the documents of the documentary corpus and the vector LTV. Each time the user consults a document, its profile is modified, thus the system proposes to him new documents. The algorithm below describes the process of the filtering documentary associated with a new consultation of a document by the user.

Algorithm 2: Algorithm of Textual Documents Filtering
Input:
Document V_D,
Pos_{V_D}: position of document V_D,
user model $X = < id, S >$,
α: constante representing the similarity threshold,
Output: proposal of documents to the user,
begin
1. Associate the document V_D to a class of documents (see Algorithm1);
2. Calculate the vector of the class to which V_D belongs;
3. Calculate the distribution of the whole classes;
4. Calculate the vector LTV;
5. Calculate the similarity between vector LTV and the whole documents of the documentary corpus;
6. Propose to the user the documents whose similarity is higher than the threshold α;

end

6 Experimentation

In order to validate the proposition, experiments were performed. The main objective is to measure the system ability to propose docuemts of regular interests.

6.1 Method

The documents used for our empirical study are press articles, collected from 5 different online newspapers in different periods. Our documentary corpus contains documents on 6 different thematics from 200 documents each one. The chosen thematics are: *Cinema, Data Processing, Economy, Football, International policy, National Policy.*

Starting with an empty profile. The user consults a succession of documents. For each consulted document, the *LUCI* system proposes twenty documents closed to his interest. We evaluate each proposed document.

To show the *LUCI* system ability to propose documents of regular interests, the user consults a set of documents of thematic *Data Processing* (12 documents), then, a set of documents of *Economy* (30 documents), followed by a set of documents of thematic *Football* (24 documents). Added to that, a regular consultation of documents of the thematic *Cinema* (11 documents), which represents a regular interest for the user.

We have chosen to compare *LUCI* with the algorithm *LUCI-NEG*, which is the implementation of *LUCI* without considering the distribution in time of the document classes.

6.2 Results

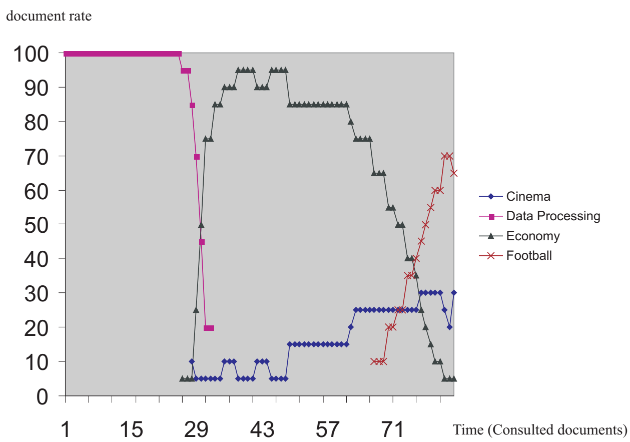

Fig. 2. *LUCI* : Document rate proposed to the user, per thematic

Figures 2 and 3 show the percentages of documents by thematic, proposed to the user, respectively by the *LUCI* system and the algorithm *LUCI-NEG*. It is noticed that starting from iteration 29, the *LUCI* system proposes documents

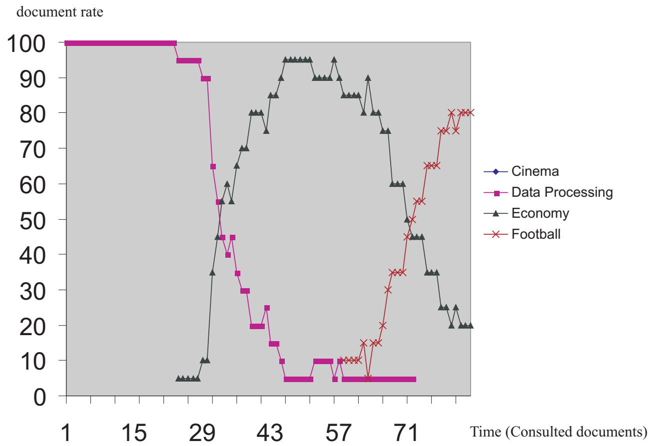

Fig. 3. *LUCI-NEG* : Document rate proposed to the user, per thematic

of thematic *Cinema* (thematic regularly consulted), whereas *LUCI-NEG* does not propose any. That is due to the capacity of the *LUCI* system to detect the classes of documents of regular interest.

The thematic *Data Processing* is just a momentary need. Indeed, the user is no longer interested by this thematic starting from iteration 14. We notice that *LUCI-NEG* continues to propose significantly the documents of this thematic until iteration 72. Whereas, the number of documents of thematic *Data Processing* proposed by *LUCI*, decreases very quickly and it becomes null at iteration 28. The same observations are made for the thematic *Economy*. That is due to the *LUCI's* capacity to detect the classes of documents of spontaneous interests and to decrease constantly and quickly the number of documents proposed which belong to these classes.

Other experiments of this type have been realised, they shows that *LUCI* detects better the classes of documents of regular interests than *LUCI-NEG*.

7 Related Work

Recently many systems have been described for documents and information filtering. In *NewsWeeder*, a NetNews Filtering system, *Lang* describe various alternatives to learn static user profiles [Lan95]. *WebMat*, an agent that helps users to effectively browse and search the web. *WebMat* keeps tracks of user interests in different domains through multiple TF-IDF vectors [SG83]. *Fab* is an adaptive system for Web page recommendation which represents user profiles as a single feature vector [Bal97]. *Syskil& Webert* is an intelligent agent that represents a profile as Boolean features using a Naive Bayesian classifier to determine whether a Web page is relevant or not [PB97]. *Alipes* keeps track of user interest through three descriptors to learn user profiles [WIY99]. In [PP02] a recommendation algorithm that is based on the maintenance of user profiles and their dynamic adjustment according to the users behavior. Neural network have been used to learn user profiles for filtering news articles on the Internet [TT98][MS94]. In [DN02], *Danilowicz* and *Nguyen* uses the user profile for an intelligent information retrieval, they compares the distance between request and user profile and the distance between request and results.

All this research does not take into account the distribution in time of the documents consulted by the user.

8 Conclusion

Having noted that the current systems of document personalization do not take into account the temporal factor when proposing documents to the users. We proposed in this article an approach which takes into account the history of the user's actions and their evolution in time. That by favoring the classes of documents regularly consulted by the user more than those concentrated over a short period. Indeed, we introduced a significant concept into our study, namely the classes of documents of regular interests. The system *LUCI* makes possible to detect these classes and to favor them. The experimental evaluation demonstrated the effectiveness of our approach.

A first prospect for research consists in considering the documents representation and in particular to use approaches like *LSI* [DDL+90] or *TF-IDF* [SG83] for the assignment of the weights. Another prospect for interesting research consists of proposing a variant of our approach taking into account the positive and negative feedback of the user.

References

[Bal97] M. Balbanovic. An adaptative web page recommendation service. *In Proceeding of the First International Conference on Autonomous Agents*, pages 378–385, 1997.

[Cha90] J. Chauch. Dtermination smantique en analyse structurelle : une exprience base sur une definition de distance. *TA Information, vol. 2*, pages 121–167, 1990.

[DDL+90] S. Deerwester, S. Dumais, T. Landauer, G. Furnas, and R. Harsman. Indexing by latent semantic analysis. *Journal of the Society for Information Science, vol. 41*, pages 391–407, 1990.

[DN02] Czeslaw Danilowicz and Huy Cuong Nguyen. Using user profiles in intelligent information retrieval. In *Proceedings of the 13th International Symposium on Foundations of Intelligent Systems*, pages 223–231. Springer-Verlag, 2002.

[Lan95] K. Lang. Newsweeder: Learning to filter netnews. *Proceedings of Machine Learning Conference*, pages 331–339, 1995.

[MS94] M. McElligot and H. Sorensen. An evolutionary connectionist approach to personnal information filtering. *Proceedings of the Fourth Irish Neural Network Conference, Dublin, Ireland*, pages 141–146, 1994.

[PB97] M. Pazzani and D. Billisus. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning Journal*, pages 313–331, 1997.

[PP02] N. Papadopoulos and D. Plexousakis. The role of semantic relevance in dynamic user community management and the formulation of recommendations. *CAISE 2002 Springer-Verlag Berlin Heidelberg*, pages 200–215, 2002.

[Sal71] G. Salton. The smart retrieval system - experiments in automatic document processing. 1971.

[SG83] G. Salton and M.J Mc Gill. Introduction to modern information retrieval. *New York: McGraw-Hill*, 1983.

[Sin97] A. Singhal. Term weighting revisited. *PhD thesis, Departement of Computer Science, Cornell University*, 1997.

[SSMB95] A. Singhal, G. Salton, M. Mitra, and C. Buckley. Document length normalization. *Technical Report, Departement of Computer Science, Cornell University*, 1995.

[TT98] A. Tan and C. Teo. Learning user profiles for personalized information dissemination. *In Proceedings, 1998 IEEE International Joint Conference on Neural Networks, Alaska, pages 183–188, 1998.*, 1998.

[WIY99] D.H. Widyantoro, T.R Ioerger, and J. Yen. An adaptative algorithm for learning changes in user interests. *Proceedings of the Eighth International Conference on Information and Knowledge Management*, pages 405–412, 1999.