Similarity measure to identify users' profiles in web usage mining

Firas Abou Latif, Nicolas Delestre, Nicolas Malandain, Jean-Pierre Pécuchet

▶ **To cite this version:**

Firas Abou Latif, Nicolas Delestre, Nicolas Malandain, Jean-Pierre Pécuchet. Similarity measure to identify users' profiles in web usage mining. INFORSID XXVIII°, May 2010, Marseille, France. pp.77-92. hal-00560096

HAL Id: hal-00560096
https://hal.science/hal-00560096

Submitted on 27 Jan 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Similarity measure to identify users' profiles in web usage mining

Firas Abou Latif — Nicolas Delestre— Nicolas Malandain — Jean-Pierre Pécuchet

Laboratoire LITIS, EA 4108, INSA de Rouen
BP 08 Avenue de l'université
76801 Saint Étienne du Rouvray, France
{prénom.nom}@insa-rouen.fr

ABSTRACT. *Nowadays, content available on the Internet is continuously growing. Websites are gathering more and more information. It makes the website browsing process even harder. This paper addresses the web usage mining problem. We try to use a characteristic patterns algorithm to identify users' profiles. Our experiment shows that this algorithm is not suitable for our encyclopedic hypermedia. Therefore, we present an alternative approach through casting a user trace (set of transitions) as a graph. Then we suggest a similarity measure among different browsing traces. To validate this measure we used the t-SNE algorithm (t-Distributed Stochastic Neighbor Embedding) which allows us to project our data in a two dimensional space. Then we apply SVM classification algorithm (Support Vector Machine) and compare our results with the results of characteristic patterns algorithm.*

RÉSUMÉ. *L'identification du profil d'un utilisateur d'un hypermédia est une problématique récurrente. Le Web Usage Mining, qui tente de résoudre ce problème, propose des techniques basées sur le principe des motifs caractéristiques. Nous montrons dans cet article que ces algorithmes ne fonctionnent pas dans notre contexte d'hypermédia encyclopédique. Comme alternative, nous proposons d'utiliser des algorithmes de classification. Ceci nous amène à définir une mesure de similarité entre traces de navigation, représentées sous forme d'ensembles de transitions. Cette mesure utilise une similarité entre documents basée sur leur contenu et leurs liens dans l'hypermédia. Afin de valider notre approche, nous avons construit une base d'apprentissage, constituée de traces issues de sept classes (à l'image de nos données réelles). Nous avons alors utilisé l'algorithme du t-SNE en faisant varier deux paramètres : le pourcentage de transitions discriminantes dans une trace et la similarité entre transitions discriminantes. Les bons résultats obtenus sur ces données jouets nous permettront certainement par la suite de valider notre approche sur des données réelles.*

KEYWORDS: *Web usage mining, Classification, SVM, t-SNE.*

MOTS-CLÉS : *Web usage mining, Algorithme de classification, SVM (Séparateurs à Vaste Marge), t-SNE.*

1. Introduction

The increase in website content causes navigation problems, for example more than three millions English articles composed the Wikipedia[1] in January 2010. This mass of information makes the browsing process of a website harder and harder. Adaptive hypermedia tries to discover a solution to this problem by dynamically changing website internal connections (Rheaume, 1993). This adaptation is built out of the information that we guess from the hypermedia browsing behaviour of the user (knowledge, performance, etc.) (Chen et Magoulas, 2005). The adaptive hypermedia needs knowledge about the user. Such knowledge can be obtained by asking user, or by studying his behavior. Adaptive hypermedia can use web usage mining to find users' profiles. In the web mining domain, data mining tools are often used to extract knowledge. Web mining includes three sub-domains (Éric Guichard, 2004): web content mining, web structure mining, web usage mining (WUM).

WUM is used to discover unknown useful information from data (Kosala et Blockeel, 2000). The goal is to capture, to model and to analyze patterns and profiles of website users. The models discovered are usually represented as collections of pages, of transactions or resources that are reached frequently by one user or a particular user group (Mobasher, 2007). The classical algorithm for solving this problem is divided into three dependant steps (Tanasa, 2005): **pre-processing**, which structures the data from navigation, **pattern discovery**, which may characterize the profile, **pattern analysis**, attempting to associate patterns with one or more profiles.

In our work we aim at adapting automatically a geographical encyclopedia website called Hypergeo[2]. So we need to identify the user's profile for best website adaptation. Therefore, we use the latter algorithm to obtain the profile of a user browsing Hypergeo. We describe, in the first part of this article, the Hypergeo hypermedia. Then we use pattern discovery methods based on the identification of characteristic patterns. We show that this method is not adapted to our data. This leads us in the fourth part to propose a similarity measure among users' traces. Then we validate our proposal by using data generated according to real observation. We find criteria allowing us to classify a new users using the SVM algorithm. Next, we compare our approach with characteristic patterns over the same data. We conclude by proposing a measure of similarity for real data as future works.

2. Context: Hypergeo

Hypergeo is a free public geographical encyclopedia website available in three languages (French, English and Spanish). The project began in 2001, it aims at publishing the main concepts and theories of geography (Elissalde et Kosmopoulos, 2007). It is hierarchically organized into three main sections to allow language selection. For each

1. www.wikipedia.org
2. www.hypergeo.eu

language, every topic is divided into sub-topics or articles. Figure 1 shows this organization, topics and articles are respectively represented using diamonds and ovals.

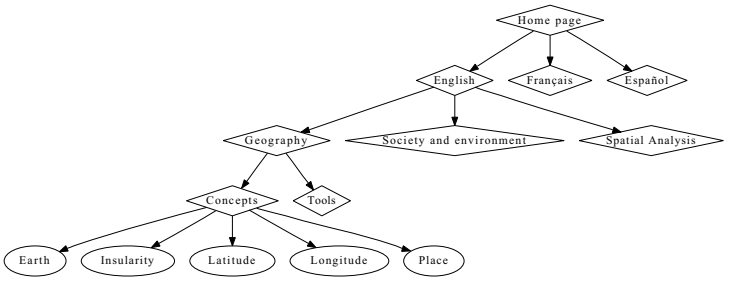

Figure 1. *The organization of the Hypergeo website.*

The Hypergeo website contains 398 articles and 78 topics written by 59 experts in geography (217 articles and 33 topics are in French). In order to experiment web data mining techniques, we have collected information about users navigations, i.e. HTTP transactions, for more than three months (from 14 February 2008 to 30 May 2008). At the end of this period, we obtained the navigational data of 17902 users who carried out 89921 transactions (a user is identified by a cookie). Among them, 8466 users carried out 48056 transactions on French documents. To avoid any service disruption on the Hypergeo website, search engine robots were allowed to hit the original website, while users' requests were reoriented to a copy of the website with full transparency. HTTP dated transactions were stored in a database. During this period, on the first visit of a user, he was asked to select one of seven profiles determined by an expert, *cf.* table 1 (because, essentially Hypergeo is used by teachers and students).

1	student in geography	2	student in another field
3	teacher in geography	4	teacher in another field
5	professor in geography	6	professor in another field
7	other		

Table 1. *Hypergeo users' profiles.*

Even though our final goal is to determine users' profile among these seven profiles, also we try to find out users' profile among three groups of profiles: student (1, 2), teacher (3, 4, 5, and 6) and other (7) by using the technique of characteristic patterns.

3. Applying Web Usage Mining (WUM) on Hypergeo

WUM can help to optimize the functionality of web applications, and provide more personalized content to users. The process of WUM can be divided into three

correlated steps: data gathering and preprocessing, model discovery and model analysis (Cooley, 2000) (*cf.* figure 2). The model analysis is the final step of the WUM process. It depends of the goal of the research. Our work focuses only on data gathering and preprocessing and model discovery (for more details about WUM see (Liu, 2007)).

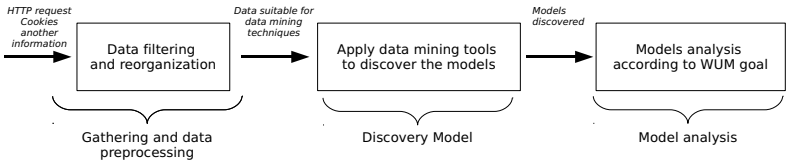

Figure 2. *Web usage mining process.*

3.1. *Gathering and data preprocessing*

Typically, this step is an important task in all WUM applications. It uses the information of HTTP requests, cookies and other information like log files. This information is filtered and reorganized into database to be suitable for the next step. This process is often the longest stage and is computationaly demanding. Data collection and preprocessing is a critical step for the extraction of useful models.

3.2. *Discovery Model*

When the first phase is done, the techniques of data mining can be applied. The techniques most frequently used in WUM are: cluster mining, association rules mining, and characteristic patterns.

3.2.1. *Cluster mining*

Clustering is a non supervised classification technique (i.e. we have no label on the data). It is used in data mining to group a collection into unlabelled significant groups (Jain et Dubes, 1988). Unsupervised learning does not need predefined classes to characterize objects. When cluster mining is applied to web, the objects can be web documents, references to these documents, or even users' visits (Koutri *et al.*, 2005).

3.2.2. *Association rules mining*

Searching for association rules refers to the identification of all co-occurrence of elements in the subset of data, so that the presence of an element in a set implies the presence of other elements (Mobasher *et al.*, 1996). Agrawal et al. define the problem of searching association rules in massive datasets (Agrawal *et al.*, 1993).

Profile	Student	Teacher	Other
Number of tested users	2150	178	239
Number of tested users having characteristic pattern	1160	120	164
Number of correctly identified users	413	43	58
Percentage of all users	19,21%	24,16%	24,27%
Percentage of all users having characteristic patterns	35,60%	35,83%	35,37%

Table 2. *Result for the discovered characteristic pattern method ($S_{min} = 0.1$, $C_{min} = 0.1$).*

3.2.3. *Characteristic patterns*

Characteristic patterns is a common algorithms used in web usage mining. This method computes two coefficient for all patterns appearing in users' browsing. Every pattern is characterized by the support S and by the confidence C. The support S is the conditional probability that this pattern exists in the set of the pages visited by the user u. The confidence C is the conditional probability that the set of the pages having this pattern is included in the set of the pages visited by the user u.

A characteristic pattern for a user u is a pattern which has a support S larger than the threshold S_{min}, and a confidence C higher than the threshold C_{min}. The larger values of S_{min} and C_{min} make the determination of the profile for new users more accurate. Gao et al. (Gao et Sheng, 2004) present the characteristic pattern mining algorithm that uses the FP-Growth (Han *et al.*, 2004) and naive Bayesian multi-net method to identification users.

3.3. *Application to Hypergeo*

First we cast our data to the three profiles seen previously, *i.e.* student, teacher and other. During the period of information gathering, we saved the data in the following form: (User ID, source page, destination page). Every user ID is linked with one of the seven profiles in the table 1. The user IDs are anonymous (no IP address are stored) and are automatically generated numbers. To test characteristic patterns algorithm, we only select French users who perform more than certain transitions number, because we can not guess user's profile from only one transition. In the literature, a pattern is characteristic when its parameter S is higher than $S_{min} = 0.5$, we accept in this case that it has a low confidence C. Or, it is necessary that its confidence C is higher than $C_{min} = 0.5$, for accepting that its support S is low.

Unfortunately, in our data we do not have any of these two conditions. Indeed, when we set S_{min} larger than 0.1, no pattern satisfies the first condition. And, when we set S_{min} equals to 0.1, and C_{min} larger than 0.5, we have characteristics patterns only for the student profile. The best results were obtained with $C_{min} = 0.1$ and $S_{min} = 0.1$ (*cf.* table 2). In this case, we note that the recognition rate is very poor, equivalent to random.

These results can be explained by the encyclopedic nature of Hypergeo. Indeed when the user enters the site Hypergeo he has a goal, even though he finds his goal maybe he continues browsing the website. Thus no characteristic patterns were found. On the other hand, Hypergeo is an encyclopedic website, so the pages are thematic and organized hierarchically. Thus there are similarities among different pages could be used to identify user's profile.

4. Similarity measure

As we saw in the previous section, the algorithm based on characteristic patterns does not reach our goal to identify users' profiles. Thus, we propose to experiment an identification method based on classification. This leads us to define a similarity measure between the information collected during users' navigation. The "trace" is defined as the data collected during the navigation of one user. In our case, we extracted this data from information registered in the server (i.e. we lose some information like the use of back button). The structure of the trace is similar to an oriented chronological graph where the pages are the vertices and the edges are the dated transitions, *cf.* figure 3.

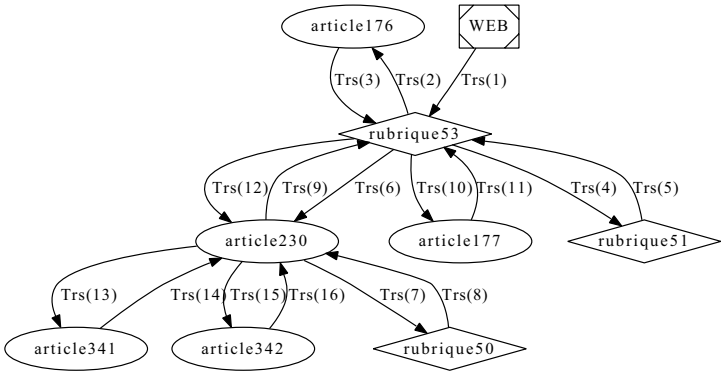

Figure 3. *Hypergeo user's trace as graph.*

The traces may be seen from several points of view:

– a set of vertices, as seen in the previous section with characteristic patterns,

– a set of transitions ($\{vertexX \rightarrow vertexY, ...\}$),

– a list of transitions (time-ordered).

We have to work with methods insensitive to the order of the visited pages. Therefore, we choose the trace as a set of transitions. To calculate a similarity between two users, we define a similarity between two traces. Then, it is necessary to define a similarity between the elements of the traces, *i.e.* the transitions (succession of two documents/nodes)

Similarity between traces Sim_{trace}

To apply a classification algorithm, we need to define a similaity between users. Gaussian kernel is often used (equation 1) to measure a similarity between attributes.

$$k(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{\| \mathbf{x} - \mathbf{y} \|^2}{2\sigma^2}\right) \quad [1]$$

This kernel is applied when users X and Y have numeric attributes, unluckily it is not the case for our data. Therefore, we propose to calculate the similarity between two traces from the set of the transitions representing the user's path in the website. For example, in Figure 4, the trace of user U_A is the set of his transitions A,
$A = \{(2,3),(3,8),(8,3),(3,6),(6,2)\}$. And for user U_B, the transitions of his trace is the set B, $B = \{(1,2),(2,5),(5,8)\}$. Figure 4 shows that the transition $(2,3)$ has two appearances in path of user A, but only one in the set A (because we use a set to present the elements of trace, and the set can not have two identical elements).

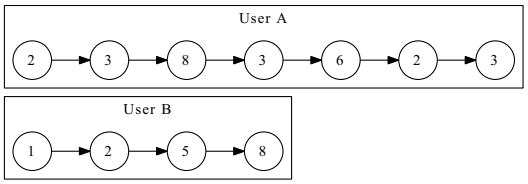

Figure 4. *An example of users' paths.*

A possible similarity between the trace of user U_A and the trace of user U_B is:

$$Sim_{trace} = \exp\left(-D_{trace}(U_A, U_B)\right) \quad [2]$$

As shown previously, a trace is like a graph. So, $D_{trace}(U_A, U_B)$ is defined as the average of dissimilarities between all transitions of users U_A and U_B (Suard et Rakotomamonjy, 2007). In other words, the formula to calculate the dissimilarity between two traces can be written as:

$$D_{trace}(U_A, U_B) = \frac{1}{|A||B|} \sum_{i:t_i \in A} \sum_{j:t_j \in B} D_{transition}(t_i, t_j) \quad [3]$$

Knowing that $D_{transition}(t_i, t_j)$ is the dissimilarity between two transitions. However this dissimilarity is very sensitive to different trace lengths. For example, let A,B,C three sets of transitions, $A = \{1,2,4,8,9\}$, $B = \{1,2\}$, $C = \{5,7,11,13,15,17,18,20\}$. With the previous formula, we find that the dissimilarity between A and B is greater than the dissimilarity between A and C because of the cardinality of C (in spite of A and B having two transitions in common, while A and C have none). Therefore we must improve the dissimilarity satisfying the following constraints:

– the dissimilarity between two identical traces should be zero,

– the dissimilarity between a trace and a sub-trace included in it, must be smaller than dissimilarity between two traces that do not satisfy this condition,

– the number of transitions in common should prevail over the length of traces.

Hence, we define the dissimilarity between traces in the following manner. Let:

– Set_A and Set_B be the set of user's transitions respectively U_A and U_B,

– $C = Set\_A \bigcap Set\_B$,

– $D = Set\_A \setminus C$,

– $E = Set\_B \setminus C$.

Figure 5 exhibits the relations between Set_A and Set_B.

Then we define $D_{trace}(U_A, U_B)$ as follows:

$$\begin{aligned} D_{trace}(U_A, U_B) \quad = \quad & \frac{|C||D|}{fac} \sum_{i:t_i \in C} \sum_{j:t_j \in D} D_{transition}(t_i, t_j) + \\ & \frac{|C||E|}{fac} \sum_{i:t_i \in C} \sum_{j:t_j \in E} D_{transition}(t_i, t_j) + \\ & \frac{|D||E|}{fac} \sum_{i:t_i \in D} \sum_{j:t_j \in E} D_{transition}(t_i, t_j) \end{aligned} \qquad [4]$$

With: $fac = (|C||D| + |C||E| + |D||E|)$.

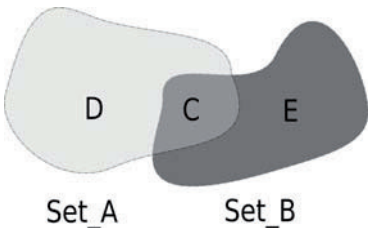

Figure 5. *Users' transitions sets Set_A, Set_B.*

This dissimilarity validates the three previous criteria. If $U_A = U_B$, then D and E are empty set ($|D| = |E| = 0$) and $D_{trace}(U_A, U_B) = 0$. If we have trace and sub-trace, E is the empty set, so only first part of equation 4 is not null. For two different traces, if they do not have shared transitions we go back to equation 3, in other case we apply the equation 4 with it's three parts. In the next section we validate this similarity measure, then we propose a definition for $D_{transition}$.

5. Validation

Identifying $D_{transition}$ characteristics using synthetic Data

The analysis of the dissimilarity defined above shows that two traces are considered close if they share identical transitions or if $D_{transition}(t_i, t_j)$ is low, so the transitions are similar (although it might be different).

Therefore we must determine which of these two factors is the most important. This is why, we test this dissimilarity on a synthetic database built like the actual outcome of Hypergeo:

– same number of classes (seven),

– same proportion to the traces number in each class (first class is predominant over six other classes),

– same transitions possible number in a trace,

– same length traces average,

– same average of the trace length.

Then we generate several traces sets by varying two parameters:

1) the discriminatory transitions percentage in a trace $DiscNB$, the discriminatory page of profile is the page that presents in this profile more than in the other profiles

$$p(trans_i \in TransDisc_i | profil_i) \geq DiscNB \quad [5]$$

2) similarity among discriminatory transitions of the same profile $SimDisc$.

We will evaluate the quality of the similarity measure based on these two parameters using an algorithm for data projection and SVM classification algorithm.

5.1. *Validation using t-SNE (t-Distributed Stochastic Neighbor Embedding)*

Projecting methods are used to visualize the data for better interpretation. These methods reduce the data dimension for displaying them in a projected space. A good projection method must perform two criteria:

1) two far data in the original space will be far in the projected space;

2) two close data in the original space are close in the projected space.

As indicated by Duda et al. (Duda *et al.*, 2000) and Hastie et al. (Hastie *et al.*, 2001), there are several projection methods: Principal Component Analysis (PCA), Multidimensional Scaling, Kohonen map (self-organizing map), SNE (Stochastic Neighbor Embedding) and t - SNE. The t-SNE algorithm is one of the last proposed algorithm. The rapidity and quality of the projection is particularly good (van der Maaten et Hinton, 2008). So we used it to project the traces in a two dimensional Euclidean space.

Figures 6 and 7 show the t-SNE results for $DiscNB = 0.1$ and $SimDisc = 0.3/0.5$. We see in the two figures that the seventh profile is discriminated from the other profiles. These two figures do not give any clue about the influence of the similarity among discriminatory transitions.

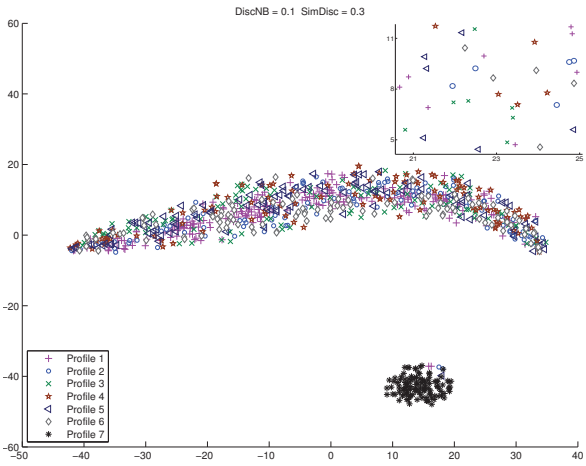

Figure 6. *t-SNE projection,* $DiscNB = 0.1$, $SimDisc = 0.3$.

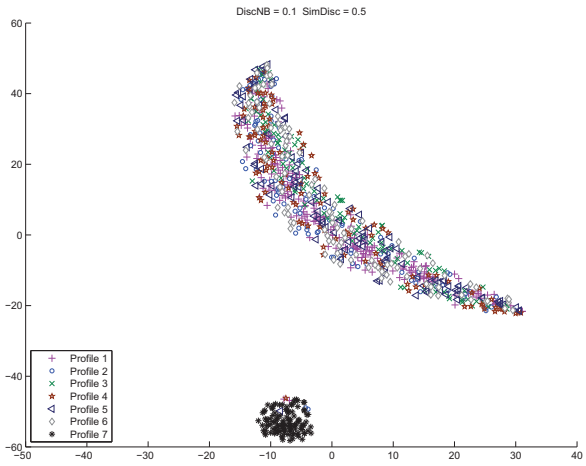

Figure 7. *t-SNE projection,* $DiscNB = 0.1$, $SimDisc = 0.5$.

Then we take other examples in Figure 8 ($DiscNB = 0.2$ and $SimDisc = 0.3$). We can discriminate the seven profiles with some errors. The seventh profile move closer to the other profiles. In Figure 9, we raise $DiscNB$ for a fixed $SimDisc$

($DiscNB = 0.3$ and $SimDisc = 0.3$). We see that the first six profiles are distinguished and the seventh profile (profile "other") is discriminated with errors.

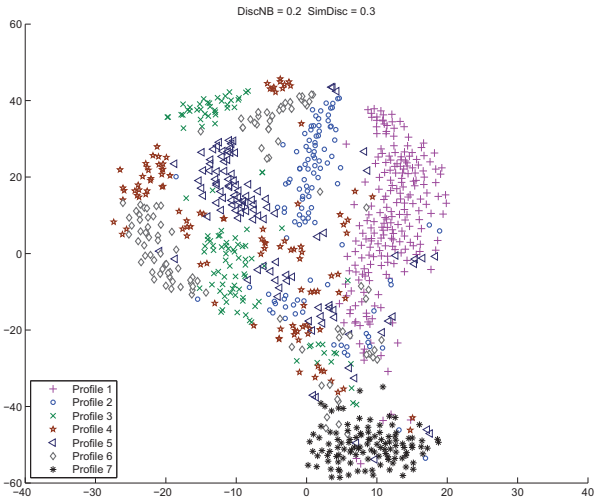

Figure 8. *t-SNE projection,* $DiscNB = 0.2$*,* $SimDisc = 0.3$*.*

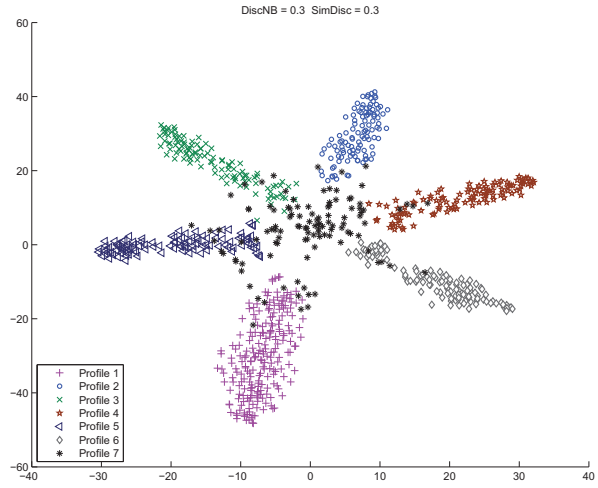

Figure 9. *t-SNE projection,* $DiscNB = 0.3$*,* $SimDisc = 0.3$*.*

These figures led us to study the relationship among the recognition rate and the values of $DiscNB$, $SimDisc$. To study this relationship, we use the SVM classifications algorithm with several values of $DiscNB$, $SimDisc$.

5.2. *Validation using SVM*

The SVM is a supervised classification method. It is based on the existence of a linear separator on the data. The linear separator proposed by this method can be computed in a high dimensional space induced by a kernel. Thus, SVM is suitable for nonlinearly separable data (Canu, 2007). There are several algorithms to perform the SVM method as incremental SVM (Cauwenberghs et Poggio, 2001), DirectSVM (Roobaert, 2002) and SimpleSVM (Vishwanathan et Narasimha Murty, 2002). In our research, we use SVM and Kernel Methods Matlab Toolbox (Canu *et al.*, 2005) that applies SimpleSVM. Our experimental protocol consists in generating several databases using the two parameters $DiscNB$, $SimDisc$ by tacking at each time a part of this database for the learning phase (60%) and the other for the test phase (40%). We vary $DiscNB$ from 0.1 to 0.7 by step 0.1 (we do the same for $SimDisc$). For each synthetic database, we build the confusion matrix to study the classification quality. Table 3 presents an example of the confusion matrix.

		Decision		
		1	i	k
Profile	1	a_{11}	a_{1i}	a_{1k}
	i	a_{i1}	a_{ii}	a_{ik}
	k	a_{k1}	a_{ki}	a_{kk}

Table 3. *The confusion matrix*

Each a_{ij} is the number of traces that belong to the profile i and have been classified as belonging to profile j. Table 4 presents the confusion matrix by applying the SVM with $DiscNB = 0.3$ and $SimDisc = 0.3$. We can measure the classification quality by the following criteria:

$$\text{criterion } 1 = \frac{\sum\limits_{i} a_{ii}}{\sum\limits_{i,j} a_{ij}} \quad (i, j = 1, 2, \ldots, \text{nb profiles}) \qquad [6]$$

There is a strong impact of the profile having a big number of users on the classification quality measure. Therefore we propose to use the following criterion:

$$\text{criterion } 2 = \frac{1}{\text{nb profiles}} \sum_{i} \frac{a_{ii}}{a_{ii} + \sum\limits_{j,j\neq i} a_{ij} + \sum\limits_{j,j\neq i} a_{ji}}$$
$$(i, j = 1, 2, \ldots, \text{nb profiles}) \qquad [7]$$

Figure 10 presents the results of applying SVM and characteristic patterns algorithms on our synthetic data according to criterion 1. Figure 11 show the results according to criterion 2. For characteristic patterns algorithm, $S_{min} = 0.5$ and $C_{min} = 0.5$. We test several values for $DiscNB$. The characteristic patterns algorithm performs well when $DiscNB \geq 0.4$. In the Hypergeo users' traces, the discriminatory transitions/pages percentage is lower than 0.4. But the experiments on synthetic data show

		Decision						
		1	2	3	4	5	6	7
Profil	1	147	0	0	0	0	0	0
	2	0	76	0	0	0	0	0
	3	0	1	75	0	0	0	0
	4	2	0	0	74	0	0	0
	5	2	0	0	0	73	1	0
	6	0	0	0	0	0	76	0
	7	18	6	10	12	10	12	8

Table 4. *Confusion matrix for* $DiscNB = 0.3$, $SimDisc = 0.3$

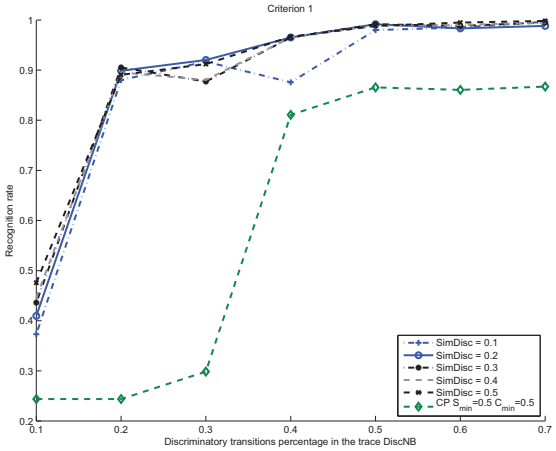

Figure 10. *Characteristic patterns CP and SVM results according to criterion 1.*

that we need to have $DiscNB \geq 0.4$. This result explains why characteristic patterns method is not suitable for the Hypergeo case.

Conversely, the figure 10 and 11 show that our method gives good recognition rates for $DiscNB \geq 0.2$ even though $SimDisc = 0.1$. When $SimDisc$ raises, the recognition rate raises. So we can say that the discriminatory transitions percentage play a role more important than the similarity among discriminatory transitions, but the improvement of similarity among discriminatory transitions will improve the recognition rate.

Moreover, the figures 10 and 11 show that the recognition rate of our approach is better than the recognition rate of characteristic patterns for all $DiscNB$ values. For example in Figure 11, for $DiscNB = 0.5$, the recognition rate of characteristic patterns and our approach are respectively 0.74 , 0.96 even for $SimDisc = 0.1$.

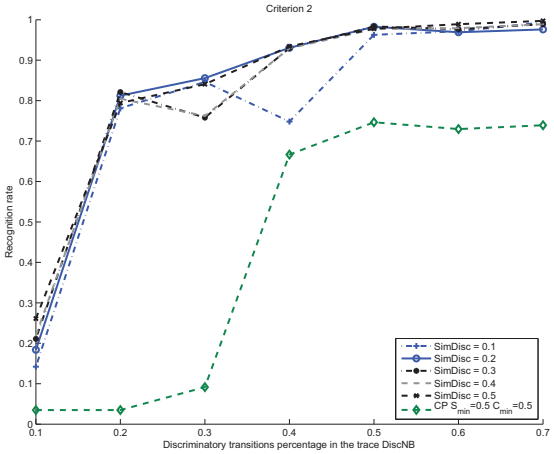

Figure 11. *Characteristic patterns CP and SVM results according to criterion 2.*

6. Future work

Our experiment shows that the similarity among discriminatory transitions improve the recognition rate. Therefore we plan to find a definition to $D_{transition}$ and to apply our similarity on Hypergeo users' traces. So we propose $D_{transition}(t_i, t_j) = (D_{doc}(d1_i, d1_j) + D_{doc}(d2_i, d2_j))/2$, where $d1_i$ is the source node and $d2_i$ is the destination node of the transition t_i. In Hypergeo, the nodes are articles or topics. So we must define dissimilarity among topics, among articles and between topic and article. An article differs from a topic by the fact that the article has a text. We can therefore obtain dissimilarity among articles using methods such as tf-idf or LSA. Moreover, Hypergeo is an encyclopedic website, topics and articles are organized hierarchically, a dissimilarity among them based on topological dissimilarity is possible. Table 5 summarizes our proposal.

d_1	d_2	$D_{doc}(d_1, d_2)$
article	article	a dissimilarity such as *tf-idf, LSA or other*
article	topic	Hypergeo hierarchical dissimilarity
topic	topic	Hypergeo hierarchical dissimilarity

Table 5. *Proposition of dissimilarity between documents.*

7. Conclusion

In this paper we look for Hypergeo users' profiles in order to adapt this encyclopedic website according to the traces. We have shown that the web usage mining algorithm based on characteristic patterns is not suitable to our data. However we

proposed to use classification algorithms to identify users' profile. The use of these algorithms rely on the definition of a similarity measure. Hence we have proposed a dissimilarity among traces D_{trace}, we cast the trace as a set of transitions, then we propose a similarity measure between them (equation 4).

We verify that this measure avoids the impact of long trace. Next, we test this measure on synthetic data to find the criteria of discriminatory transitions percentage and similarity among them for good recognition rate. We show that our method gives better recognition rate than characteristic patterns method for all value of discriminatory transitions percentage. Finally, we propose a $D_{transition}$ for future work taking into account the dissimilarity among articles and topics.

References

AGRAWAL, R., IMIELINSKI, T. et SWAMI, A. N. (1993). Mining association rules between sets of items in large databases. *In SIGMOD Conference*, pages 207–216.

CANU, S. (2007). Machines à noyaux pour l'apprentissage statistique. *Techniques de l'ingénieur*, TE5255.

CANU, S., GRANDVALET, Y., GUIGUE, V. et RAKOTOMAMONJY, A. (2005). Svm and kernel methods matlab toolbox. Perception Systèmes et Information, INSA de Rouen, Rouen, France.

CAUWENBERGHS, G. et POGGIO, T. (2001). Incremental and decremental support vector machine learning. *In Advances in neural information processing systems 13: proceedings of the 2000 conference*, pages 409–415. The MIT Press.

CHEN, S. Y. et MAGOULAS, G. D. (2005). *Adaptable and Adaptive Hypermedia Systems*. IRM Press.

COOLEY, R. (2000). *Web Usage Mining: Discovery and Application of Interesting Patterns from Web Data*. Thèse de doctorat, University of Minnesota.

DUDA, R. O., HART, P. E. et STORK, D. G. (2000). *Pattern Classification (2nd Edition)*. Wiley-Interscience.

ELISSALDE, B. et KOSMOPOULOS, C. (2007). De la navigation au savoir. réflexions sur les nouveaux comportements de diffusion et d'acquisition des connaissances à travers l'expérience d'hypergeo. *ametist*, Numéro 1 AMETIST.

GAO, W. et SHENG, O. R. L. (2004). Mining characteristic patterns to identify web users. *In WIST*.

HAN, J., PEI, J., YIN, Y. et MAO, R. (2004). Mining frequent patterns without candidate generation: A frequent-pattern tree approach. *Data Mining Knowledge Discovery*, 8(1):53–87.

HASTIE, T., TIBSHIRANI, R. et FRIEDMAN, J. (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer.

JAIN, A. K. et DUBES, R. C. (1988). *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, N.J.

KOSALA, R. et BLOCKEEL, H. (2000). Web mining research: A survey. *SIGKDD Explorations*, 2:1–15.

KOUTRI, M., AVOURIS, N. et DASKALAKI, S. (2005). A survey on web usage mining techniques for web-based adaptive hypermedia systems. *Adaptable and adaptive hypermedia systems*, pages 125–149.

LIU, B. (2007). *Web Data Mining*. Springer Berlin Heidelberg.

MOBASHER, B. (2007). Web usage mining. *In Web Data Mining, Exploring Hyperlinks, Contents, and Usage Data*, pages 449–483. Springer Berlin Heidelberg.

MOBASHER, B., JAIN, N., HAN, E. et SRIVASTAVA, J. (1996). Web mining: Pattern discovery from world wide web transactions. Rapport technique, Technical Report TR-96050, Department of Computer Science, University of Minnesota.

RHEAUME, J. (1993). Les hypertextes et les hypermédia. *Revue EducaTechnologie*, 1(2).

ROOBAERT, D. (2002). Directsvm: A simple support vector machine perceptron. *The Journal of VLSI Signal Processing*, 32(1):147–156.

SUARD, F. et RAKOTOMAMONJY, A. (2007). Mesure de similarité de graphes par noyau de sacs de chemins. *In 21 colloque GRETSI sur le traitement du signal et des images*.

TANASA, D. (2005). *Web Usage Mining: Contributions to Intersites Logs Preprocessing and Sequential Pattern Extraction with Low Support*. Thèse de doctorat, UNIVERSITÉ DE NICE SOPHIA ANTIPOLIS.

van der MAATEN, L. et HINTON, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

VISHWANATHAN, S. et NARASIMHA MURTY, M. (2002). Ssvm: a simple svm algorithm. *IJCNN '02. Proceedings of the 2002 International Joint Conference on Neural Networks*, 3:2393 – 2398.

Éric GUICHARD (2004). *Mesure de l'internet*. Les Canadiens en Europe.