
Diviser pour découvrir : une méthode d'analyse du comportement de tous les utilisateurs d'un site web

Florent Masseglia, Doru Tanasa et Brigitte Trousse

*Projet AxIS - INRIA Sophia Antipolis
2004 route des Lucioles - BP 93
06902 Sophia Antipolis, France
email : prénom.nom@sophia.inria.fr*

RÉSUMÉ. Les travaux présentés dans cet article ont pour but d'augmenter de manière significative la pertinence et l'intérêt des motifs découverts par un processus de Web Usage Mining. En effet les motifs séquentiels découverts sur des fichiers logs, sauf si ils sont découverts sous contraintes, sont trop souvent dénués d'intérêt en raison de leur caractère évident. Notre objectif est de découvrir des comportements parfois minoritaires mais dont la cohérence les rend trop intéressants pour ne pas les considérer (comme les attaques pirates sur le site ou les personnes qui consultent la présentation du projet de recherche "alpha"). En nous basant sur une classification des motifs séquentiels obtenus sur le log, nous proposons une division récursive du problème. Cette classification repose sur des résumés des motifs et sur l'exploitation de réseaux de neurones. Nos expérimentations montrent l'obtention des motifs visés, mais aussi que leur découverte par un processus classique est impossible car elle demande de spécifier un support trop faible (jusqu'à 0,006%). En effet l'intégralité des sessions présente une telle diversité de comportements que les plus minoritaires (sortes de "niches") sont à la fois nombreux et particulièrement difficiles à isoler.

ABSTRACT. The goal of this work will be to increase the relevance and the interest of patterns discovered by a Web Usage Mining process. Indeed, the sequential patterns discovered on web log files, unless they are discovered under constraints, often lack interest because of their obvious content. Our goal is to discover minority users behaviors having a coherence which we want to be aware of (like hacking activities on the Web site or a users activity limited to a specific part of the Web site). By means of a clustering method on the discovered sequential patterns, we propose a recursive division of the problem based on pattern summaries and neural networks. Our experiments show that we obtain the targeted patterns and that their discovery by means of a classical process is impossible because of a very weak support (up to 0.006%).

MOTS-CLÉS : web usage mining, motifs séquentiels, classification, résumé des motifs.

KEYWORDS: web usage mining, sequential patterns, clustering, patterns summary, neural networks.

1. Introduction

L'analyse du comportement des utilisateurs d'un site Web, également connue sous le nom de Web Usage Mining, est un domaine de recherche qui consiste à adapter des techniques de fouille de données sur les enregistrements contenus dans les fichiers access logs. Ces fichiers regroupent des informations sur l'adresse IP de la machine, l'URL demandée, la date, et d'autres renseignements concernant la navigation de l'utilisateur. Les techniques de Web Usage Mining s'intéressent à la recherche de motifs ou de connaissances sur les comportements des utilisateurs d'un site Web afin d'extraire des relations entre les données stockées [COO 99, MAS 00, MOB 02, SPI 99]. Parmi les méthodes développées, celles consistant à extraire des motifs séquentiels [AGR 95] s'adaptent particulièrement bien au cas des logs. En théorie, l'extraction de motifs séquentiels sur un fichier access log permet d'extraire le type de relation suivant : *“Sur le site de l'INRIA, 10% des clients ont visité, dans l'ordre, la page d'accueil, la page des opportunités de travail, la page du recrutement des ITA¹, la page des missions des ITA et enfin la page des annales de concours des ITA.”*

Ce type de comportement existe, mais en théorie seulement, car l'extraction de motifs séquentiels à partir d'un fichier de type access log se heurte à de multiples problèmes : la présence du cache (sur la machine de l'utilisateur) et de proxys (qui jouent le rôle de caches de niveau régional, entreprise, etc.)², la grande diversité des pages sur le site³, l'existence de moteurs de recherche extérieurs qui permettent à l'utilisateur d'accéder directement à la partie du site qui le concerne⁴, la représentativité de la partie du site visitée par rapport au site dans sa globalité⁵, la représentativité des utilisateurs de cette partie du site par rapport au nombre total d'utilisateurs sur le site global.

Les problèmes engendrés par la présence de caches peuvent connaître certaines solutions. Par exemple spécifier dans la ressource une fréquence de mise à jour élevée pour éviter une consultation systématique du proxy. Les problèmes engendrés par la représentativité, en revanche, nécessitent une étude plus approfondie. Pour comprendre l'enjeu de ces travaux, reprenons l'exemple de navigation que la théorie prétend obtenir. Bien qu'ils soient les plus représentés, les utilisateurs consultant la partie “opportunité de travail” du site de l'INRIA en Janvier 2003 sont 0,5% des utilisateurs du site global. De la même manière, les utilisateurs consultant la partie “enseignement” du projet de recherche AxIS, ne sont que 0,01% des utilisateurs du site global.

1. Postes d'Ingénieurs, Techniciens, Administratifs.

2. Cela diminue le nombre d'enregistrements dans le log (des éléments de navigation qui sont pris en charge par les caches et qui n'arrivent donc pas jusqu'au site et ne sont alors pas enregistrés).

3. Sur un site comme celui de l'INRIA, on peut compter plus de 70000 ressources filtrées (après l'étape de sélection de données) pour le siège, et plus de 82000 ressources filtrées pour le site de l'unité de Sophia Antipolis.

4. Cela diminue le nombre d'entrées dans le log, mais aussi le nombre de navigations communes à plusieurs utilisateurs (i.e. le préfixe commun aux différentes navigations).

5. une équipe de recherche peut représenter moins de 0,7% de la globalité du site de l'INRIA).

Ainsi l'étude du log pour un tel site Web doit passer par la prise en compte de cette représentativité très particulière pour prétendre à des résultats satisfaisants.

Notre objectif est, dans un premier temps, de montrer qu'un processus d'extraction de motifs séquentiels classique⁶ ne peut pas obtenir les comportements fréquents cohérents qui présentent une aussi faible représentativité. Dans un deuxième temps nous proposons une méthode destinée à la découverte du comportement de tous les utilisateurs d'un site, y compris les plus marginaux. Nous prétendons ainsi répondre à l'un des obstacles majeurs, aujourd'hui, dans le domaine du Web Usage Mining.

Cet article est structuré de la manière suivante : la section 2 donne les définitions des motifs séquentiels et de leur application aux usages du Web, la section 3 fait le point sur les techniques de classification et motifs séquentiels qui sont utilisées en Web Usage Mining. En section 4 nous proposons la méthode de division du log appelée "diviser pour découvrir" et en section 5 nous présentons les méthodes de classification de motifs séquentiels que nous avons mises en place. Enfin la section 6 nous présentons nos principaux résultats et leur interprétations, avant de conclure.

2. Définitions

2.1. Motifs séquentiels

Ce paragraphe expose et illustre la problématique liée à l'extraction de motifs séquentiels dans de grandes bases de données. Il reprend les différentes définitions proposées dans [AGR 93] et [AGR 95].

Dans [AGR 93], le problème de la recherche de règles d'association dans de grandes bases de données est défini de la manière suivante.

Définition 1 Soit $I = \{i_1, i_2, \dots, i_m\}$, un ensemble de m achats (*items*). Soit $D = \{t_1, t_2, \dots, t_n\}$, un ensemble de n transactions ; chacune possède un unique identificateur appelé *TID* et porte sur un ensemble d'items (*itemset*) I . I est appelé un k -*itemset* où k représente le nombre d'éléments de I . Une transaction $t \in D$ contient un itemset I si et seulement si $I \subseteq t$. Le *support* d'un itemset I est le pourcentage de transaction dans D contenant I : $supp(I) = \|\{t \in D \mid I \subseteq t\}\| / \|\{t \in D\}\|$. Une règle d'association est une implication conditionnelle entre les itemsets, $I_1 \Rightarrow I_2$ où les itemsets I_1 , $I_2 \subset I$ et $I_1 \cap I_2 = \emptyset$. La *confi ance* d'une règle d'association $r : I_1 \Rightarrow I_2$ est la probabilité conditionnelle qu'une transaction contienne I_2 étant donné qu'elle contient I_1 . Le support d'une règle d'association est défini par $supp(r) = supp(I_1 \cup I_2)$. Etant donné deux paramètres spécifiés par l'utilisateur, *minsupp* et *minconfi ance*, le problème de la recherche de règles d'association dans une base de données D consiste à rechercher l'ensemble des itemsets fréquents dans D , i.e. tous les itemsets dont le

6. Nous avons écarté de nos études les méthodes d'extraction sous contraintes ou à base d'échantillonnage, pour des raisons expliquées en section 4.

support est supérieur ou égal à $minsupp$. Puis, à partir de cet ensemble, générer toutes les règles d'association dont la confiance est supérieure à $minconfi$ ance.

Pour étendre la problématique précédente à la prise en compte du temps des transactions, les mêmes auteurs ont proposé dans [AGR 95] la notion de séquence définie de la manière suivante :

Définition 2 Une *transaction* constitue, pour un client C , l'ensemble des items achetés par C à une même date. Dans une base de données client, une transaction s'écrit sous forme d'un triplet : $\langle id\text{-client}, id\text{-date}, itemset \rangle$. Un *itemset* est un ensemble non vide d'items noté $(i_1 i_2 \dots i_k)$ où i_j est un *item* (il s'agit de la représentation d'une transaction non datée). Une *séquence* est une liste ordonnée, non vide, d'itemsets notée $\langle s_1 s_2 \dots s_n \rangle$ où s_j est un itemset (une séquence est donc une suite de transactions avec une relation d'ordre entre les transactions). Une *séquence de données* est une séquence représentant les achats d'un client. Soit T_1, T_2, \dots, T_n les transactions d'un client, ordonnées par date d'achat croissante et soit $itemset(T_i)$ l'ensemble des items correspondants à T_i , alors la séquence de données de ce client est $\langle itemset(T_1) itemset(T_2) \dots itemset(T_n) \rangle$.

Exemple 1 Soit C un client et $S = \langle (3) (4\ 5) (8) \rangle$, la séquence de données représentant les achats de ce client. S peut être interprétée par "C a acheté l'item 3, puis en même temps les items 4 et 5 et enfin l'item 8".

Définition 3 Le *support* de s , noté $supp(s)$, est le pourcentage de toutes les séquences dans D qui supportent (contiennent) s . Si $supp(s) \geq minsupp$, avec une valeur de support minimum $minsupp$ fixée par l'utilisateur, la séquence s est dite *fréquente*.

2.2. Adapter la problématique des motifs séquentiels

Ce paragraphe propose de reprendre les concepts essentiels d'un processus de Web Usage Mining, afin de présenter de façon synthétique, les procédés mis en œuvre lors de l'analyse du comportement des utilisateurs d'un site Web. Les principes généraux sont similaires à ceux du processus d'extraction de connaissances exposés dans [FAY 96]. La démarche se décompose en trois phases principales. Tout d'abord, à partir d'un fichier de données brutes, un prétraitement est nécessaire pour éliminer les informations inutiles. Dans la deuxième phase, à partir des données transformées, des algorithmes de data mining sont utilisés pour extraire les itemsets ou les séquences fréquents. Enfin, l'exploitation par l'utilisateur des résultats obtenus est facilitée par un outil de requête et de visualisation.

Les données brutes sont collectées dans des fichiers access log des serveurs Web. Une entrée dans le fichier access log est automatiquement ajoutée chaque fois qu'une requête pour une ressource atteint le serveur Web (*demon http*). Les fichiers access log

peuvent varier selon les systèmes qui hébergent le serveur, mais présentent tous en commun trois champs : l'adresse du demandeur, l'URL demandée et la date à laquelle cette demande a eu lieu. Parmi ces différents types de fichiers, nous avons retenu dans cet article le format CLF spécifié par le CERN et la NCSA [W3C 95] pour les logs HTTP, une entrée contient des enregistrements formés de 7 champs séparés par des espaces :

```
host user authuser [date:time] "request" status bytes
```

La figure 1 illustre un extrait de fichier access log du serveur Web de l'INRIA Sophia Antipolis.

```
138.96.69.8 - - [03/Mar/2003 :18 :42 :14 +0100] "GET /axis/logoToile3.swf HTTP/1.1" - -
138.96.69.8 - - [03/Mar/2003 :18 :48 :00 +0100] "GET /aid/personnel/ HTTP/1.1" - -
138.96.69.8 - - [03/Mar/2003 :19 :03 :14 +0100] "GET /axis/cbrtools/ HTTP/1.0" - -
138.96.69.7 - - [03/Mar/2003 :19 :04 :44 +0100] "GET /axis/cbrtools/manual/ HTTP/1.0" - -
138.96.69.7 - - [03/Mar/2003 :19 :12 :45 +0100] "GET /axis/broadway/ HTTP/1.0" - -
```

Figure 1. Exemple de fichier access log

Deux types de traitements sont effectués sur les entrées du serveur log. Tout d'abord, le fichier access log est trié par adresse et par transaction. Ensuite une étape d'élimination des données "non intéressantes" pour l'analyse (phase de sélection) est réalisée. Au cours de la phase de tri et afin de rendre plus efficace le traitement de l'extraction de données, les URLs et les clients sont codés sous forme d'entiers. Toutes les dates sont également traduites en temps relatif par rapport à la plus petite date du fichier.

Définition 4 Soit Log un ensemble d'entrées dans le fichier access log. Une entrée g , $g \in Log$, est un tuple $g = \langle ip_g, \{(l_1^g.URL, l_1^g.time), \dots, (l_m^g.URL, l_m^g.time)\} \rangle$ tel que pour $1 \leq k \leq m$, $l_k^g.URL$ représente l'objet demandé par le client g à la date $l_k^g.time$, et pour tout $1 \leq j < k$, $l_k^g.time > l_j^g.time$.

La figure 2 illustre un exemple de fichier obtenu après la phase de pré-traitement. A chaque client correspond une suite de "dates" (événements) et la traduction de l'URL demandée par ce client à cette date.

Client	d1	d2	d3	d4	d5
1	10	30	40	20	30
2	10	30	60	20	50
3	10	70	30	20	30

Figure 2. Exemple de fichier résultat issu de la phase de pré-traitement

L'objectif est alors de déterminer, grâce à une phase d'extraction, les séquences de ce jeu de données, qui peuvent être considérées comme fréquentes selon la définition

3. Les résultats obtenus sont du type $\langle (10) (30) (20) (30) \rangle$ (ici avec un support minimum de 66% et en appliquant les algorithmes de fouille de données sur le fichier représenté par la figure 2). Ce dernier résultat, une fois re-traduit en termes d'URLs, confirme la découverte d'un comportement commun à *minSup* utilisateurs et fournit l'enchaînement des pages qui constituent ce comportement fréquent.

3. État de l'art

Dans cette section nous allons nous focaliser sur les principales techniques de classification et d'extraction de motifs séquentiels appliquées au Web Usage Mining (WUM).

3.1. Principales méthodes de classification utilisées pour le WUM

Peu de méthodes existantes de classification ont été appliquées aux données du Web : BIRCH dans [FU 00], CLIQUE dans [PER 98], EM dans [CAD 00], classification neuronale dans [BEN 02]. Une explication consiste dans le fait qu'il est difficile d'adapter certaines méthodes aux particularités des données Web. Par exemple comment définir une fonction moyenne pour les sessions d'une classe ? La quantité de ces données, tant en nombre de transactions (sessions) qu'en nombre d'individus différents (pages Web), pose aussi des problèmes car il est bien connu que la plupart des méthodes de classification retient toutes les données en mémoire.

Mobasher&al, 2002 : dans [MOB 02], les auteurs considèrent deux méthodes de classification, mais ne prennent pas en compte l'ordre des pages dans les sessions. Les sessions (appelées "transactions" dans l'article) sont représentées par des vecteurs binaires ou pondérés par des vues de pages (pages, dans la suite). Une transaction t est de la forme $t = \langle w(p_1, t), w(p_2, t), \dots, w(p_n, t) \rangle$ où $w(p_i, t)$ est le poids de la page p_i dans la transaction t . Le poids d'une page dans une transaction peut dépendre de la présence ou l'absence de cette page dans la transaction (1, respectivement 0). Le poids peut être égal à la valeur d'une fonction sur le temps total de chargement et d'affichage de la page (pouvant exprimer l'intérêt de l'utilisateur dans cette page) ou d'une autre fonction sur le type de page.

Fu&al, 2000 : dans [FU 00] les sessions des utilisateurs sont généralisées en utilisant une induction basée sur les attributs. Cette induction a comme objectif de réduire les dimensions des données. Lors d'une première étape les pages sont organisées dans une hiérarchie. Par exemple la page www-sop.inria.fr/axis/teaching/std-projet2.html est mise dans la hiérarchie suivante : www-sop.inria.fr \rightarrow **axis** \rightarrow **teaching** . Dans l'étape de généralisation la page est ainsi remplacée par une page plus générale : www-sop.inria.fr/axis/teaching/. Cette hiérarchie est construite seulement sur la syntaxe des URLs et pas sur la sémantique des pages. Les données généralisées sont ensuite classées en utilisant un algorithme efficace de classification hiérarchique - BIRCH introduit par [ZHA 96]. Dans les expérimentations décrites, BIRCH a été

capable de proposer une classification des sessions associées à 21 107 utilisateurs. Cependant, d'après les auteurs les performances de l'algorithme BIRCH se dégradent visiblement en augmentant la dimension des données ce qui limite la généralisation des pages à quelques niveaux.

3.2. Les principales méthodes d'extraction de motifs séquentiels et le WUM

Parmi les avantages que présentent les motifs séquentiels sur les règles d'association, dans le contexte du Web Usage Mining, nous soulignerons en particulier la prise en compte de la temporalité. Parmi les travaux visant à appliquer les motifs séquentiels aux logs [MAS 00, TAN 01, SPI 99, BON 01], nous présentons deux travaux [SPI 99, MAS 00] dans cette section.

Spiliopoulou&al, 1999 : l'outil WUM (Web Utilisation Miner) permet la découverte de patrons de navigation qui sont "**intéressants**" du point de vue statistique ou par leur structure. L'extraction de motifs séquentiels proposée par WUM repose sur la fréquence (support minimum) des motifs considérés. On peut aussi spécifier un autre critère subjectif pour les patrons de navigation comme par exemple le fait de passer par des pages ayant certaines propriétés ou celui d'une confiance élevée entre deux ou plusieurs pages du patron de navigation. Les logs Web sont d'abord transformés en logs agrégés en plusieurs étapes. D'abord on obtient les chemins qui sont composés d'une séquence de pages accédées sous la forme : (temps, page source, page destination). Ces chemins sont agrégés dans un "arbre agrégé". Un noeud de l'arbre correspond à une page dans un parcours de navigation. Les parcours de navigation ayant un même préfixe sont groupés ensemble et le support du noeud correspondant au préfixe sera le nombre de ces parcours. Pour découvrir les motifs de navigation, l'utilisateur spécifie des descripteurs de patrons. Ceux-ci sont des séquences de pages et de méta-caractères.

Masseglia&al, 2000 : dans [MAS 00], les auteurs proposent la plateforme Web-Tool. Cette plateforme prend en charge les étapes nécessaires à l'extraction de motifs de navigation fréquents sur un fichier de type access log, à savoir la transformation, la sélection des données, l'extraction de motifs et la visualisation des résultats. L'extraction des motifs dans WebTool repose sur PSP, un algorithme développé par les auteurs, dont l'originalité est de proposer un arbre préfixé pour gérer à la fois les candidats et les fréquents. Comme les autres méthodes d'extraction de motifs séquentiels basées sur le principe générer-élaguer, PSP se heurte au problème de la représentativité, qui devient un obstacle lors d'une recherche poussée (support très faible) de motifs.

4. La méthode diviser pour découvrir

Dans cette section, nous présentons les motivations de notre travail, en termes d'intérêt des motifs visés et de difficultés engendrées, ainsi que le principe général de division que nous proposons.

4.1. Motivations

Considérons des sites Web comme celui du siège de l'INRIA ou celui de l'unité de Sophia Antipolis. Ces sites peuvent être, en partie, représentés comme le suggère la figure 3. Il s'agit de sites riches, dont les thèmes peuvent varier de l'emploi à l'INRIA, au plan stratégique en passant par les pages des membres d'une unité (pages relatives aux activités de recherche, d'enseignement, etc. de chacun).

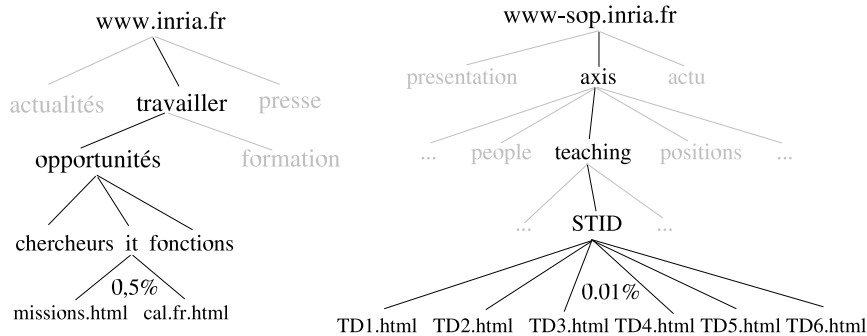


Figure 3. Une partie des sites de l'INRIA

Les leçons que l'on peut tirer d'une activité d'analyse du log correspondant à ces sites sont les suivantes :

- Généralement, les motifs séquentiels issus du fichier log d'un site de cette ampleur sont assez décevants. En effet leur significativité est assez faible et leur évidence les rend peu utiles (par ex. "0,1% des utilisateurs sont passés par la page d'accueil puis la page du sommaire").
- Les comportements intéressants sont contingentés à une partie très précise du log. Par exemple, sur la figure 3, la partie du log correspondant aux activités d'enseignement de D. Tanasa (STID) sera consultée par **0.01%** des utilisateurs enregistrés dans le log. Les utilisateurs concernés par les opportunités de travail à l'INRIA, eux, représentent 0,5% des accès sur le site.
- Si l'on veut extraire des motifs séquentiels intéressants sur ce log, il faut donc spécifier un support extrêmement bas.

Cette dernière réflexion nous permet de mettre en évidence les problèmes suivants. Dans notre cadre de travail, nous avons défini deux voies qui sont volontairement mises de côté. Tout d'abord la recherche de motifs sous contraintes. Sans discuter l'efficacité de cette méthode qui a fait ses preuves, nous prenons cette position pour des arguments relatifs à la seule complétude des résultats. En spécifiant des contraintes, l'utilisateur oriente l'algorithme dans ses recherches. Nous considérons donc, dans le cadre de cet article, que cette technique ne permet pas la découverte de tous les motifs (qui sont par essence à découvrir, donc invisibles de l'utilisateur et de ses contraintes).

La deuxième technique que nous avons écartée est celle de l'échantillonnage. En effet, compte tenu de la très faible représentativité des comportements que nous cherchons, la taille de l'échantillon à spécifier devrait approcher la taille du log. Dans ces conditions, une méthode d'échantillonnage verrait son principal atout remis en cause.

Pour en revenir au problème de la baisse du support, et si l'on garde à l'esprit les arguments de notre éloignement pour la recherche sous contraintes ou l'échantillonnage, imaginons que l'on baisse le support de manière trop importante. Deux conséquences devront alors être prises en compte :

- Le temps de réponse nécessaire au déroulement de l'extraction de motifs séquentiels, correspondant à ce support, devient trop long (la plupart du temps, les résultats ne seront même jamais obtenus, en raison de la complexité du processus).
- Le nombre de fréquents générés par ce processus (dans le cas où il se termine) fait que les résultats seront difficiles à exploiter.

Pourtant, les comportements que nous voulons découvrir ont un support typiquement très faible. En effet ces comportements correspondent à des minorités, mais nous avons pour objectif de les découvrir car nous estimons qu'ils sont révélateurs et utiles. Par exemple, parmi ces comportements, nous pouvons compter les attaques pirates sur le site, ou encore la consultation (certainement par des étudiants) des pages d'exercices correspondants à une séance de travaux pratiques. Notre ambition est alors de mettre en évidence des comportements qui permettraient d'affirmer que :

- 0,04% des utilisateurs ont une activité susceptible d'être une attaque sur le site. Parmi eux, 90% ont navigué selon une séquence typique d'une attaque pirate.
- 0,01% des utilisateurs ont une activité relative à la partie "enseignement" de D. Tanasa. Parmi eux, 15% ont consulté dans l'ordre les 6 pages de son cours sur le data mining.

Le support extrêmement faible de ces motifs est essentiellement dû à la grande diversité des comportements sur le log analysé et au grand nombre d'URLs contenue dans ce site. Pour contourner les problèmes que nous venons de décrire et découvrir tout de même ces motifs, nous avons mis en place la méthode "diviser pour découvrir", décrite plus bas.

4.2. Principe général

Dans les grandes lignes, notre objectif est de découvrir des classes d'utilisateurs (regroupés en fonction de leur comportement sur le site) et d'analyser, ensuite, leurs navigations grâce à une extraction de motifs séquentiels. Notre méthode s'appuie donc sur deux phases. La première phase consiste à diviser le log en sous-logs, censés représenter des activités distinctes. La seconde phase consiste à analyser le comportement des utilisateurs enregistrés dans chacun de ces sous-logs. Le principe général de notre méthode est donc le suivant :

- 1) Extraire des motifs séquentiels sur le log d'origine.

2) Procéder à une classification sur ces motifs séquentiels. Cela nous permet d'obtenir une première classification sur le comportement des utilisateurs.

3) Diviser le log en fonction des classes ainsi obtenues. Chaque sous-log contient les sessions du log original qui correspondent à au moins un comportement de la classe ayant engendré ce sous-log. Un sous-log spécial est alors créé pour recueillir les sessions du log d'origine qui ne correspondent à aucune des classes obtenues dans l'étape précédente.

4) Pour chaque sous-log obtenu, réitérer l'ensemble de ce processus.

La figure 4 illustre cette façon de procéder. Dans le haut de cette figure, on peut observer l'obtention des motifs séquentiels sur le log d'origine, puis leur classification. En bas de cette figure est illustrée la division en sous-logs (SL_1 à SL_n) du log d'origine, en fonction des classes (C_1 à C_n) obtenues précédemment. On peut y constater la création du SL_{n+1} qui contient toutes les sessions n'ayant pas été reconnues comme appartenant aux comportements identifiés sur le log d'origine.

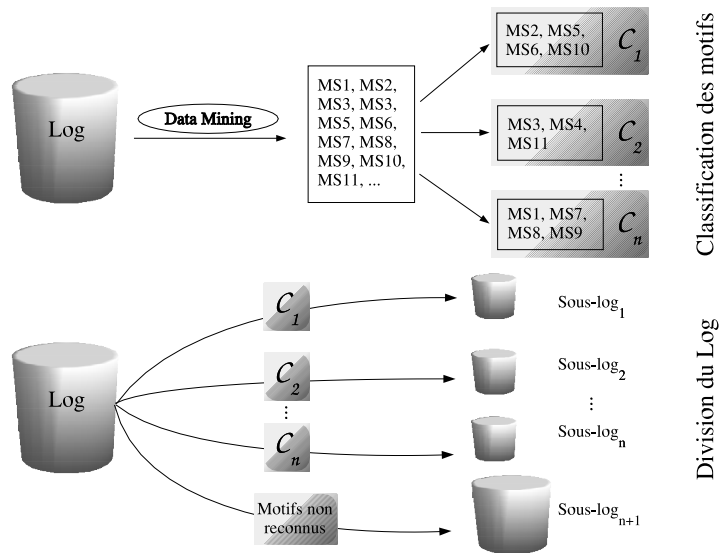


Figure 4. Principe de la méthode diviser pour découvrir.

C'est sur ce dernier sous-log que va précisément reposer la qualité des résultats produits par notre approche. En effet les premiers sous-logs obtenus contiennent les catégories d'utilisateurs les plus représentés. Ils ont donc un intérêt certain, mais l'analyse la plus riche d'enseignements viendra de l'exploitation qui peut être faite du sous-log SL_{n+1} contenant les sessions "inclassables". En considérant désormais ce sous-log comme un log "d'origine", et en réitérant le processus de division du log (tel qu'il est décrit par la figure 4), nous serons en mesure de découvrir des comportements dont la représentation est faible dans le log d'origine.

5. Classification des motifs séquentiels

Nous avons étudié plusieurs méthodes de classification des motifs séquentiels. Leur comparaison repose alors sur la qualité des classes créées, soit la façon dont les motifs seront regroupés. Nous décrivons ici les trois méthodes que nous avons envisagées pour classer les motifs, par ordre d'efficacité. Pour illustrer nos propos nous allons considérer un seul exemple pour toutes ces méthodes.

Exemple 2 *Considérons le groupe d'étudiants venus consulter les pages de cours sur le data mining de D. Tanasa : axis/teaching/STID/. Plusieurs pages peuvent alors être consultées sur cette partie du site : annee02-03.html, TD1.html, TD2.html, TD3.html, exemple-accesslog-complet.html et exemple-errorlog.html. Chacune de ces pages sera identifiée par les lettres a, b, c, d, e et f (i.e. a=axis/teaching/STID/TD1.html, b=axis/teaching/STID/TD2.html, etc.). L'objectif étant bien sûr de détecter que ces six pages doivent être regroupées dans une même classe lors de la classification des motifs séquentiels. En effet, une fois cette classe identifiée, un sous-log contenant les sessions correspondant à cette classe sera créé. Une extraction de motifs séquentiels sur ce sous-log permettra d'obtenir des motifs au support élevé et avec une significativité nettement améliorée.*

Reprenons les six pages (a, b, c, d, e et f) de l'exemple 2. Considérons que, parmi les motifs séquentiels trouvés, nous avons les quatre motifs suivants : 0,11% : $\langle(a)(b)\rangle$, 0,11% : $\langle(b)(c)\rangle$, 0,12% : $\langle(d)(e)\rangle$ et 0,1% : $\langle(d)(f)\rangle$. Envisageons maintenant l'application de chacune des méthodes de classification envisagées sur ce résultat.

5.1. Classification naïve

Dans un premier temps, nous avons implémenté un algorithme de classification hiérarchique, tel qu'on peut le trouver dans [HAN 01] (p. 354). Cet algorithme est basé sur une notion de similitude entre les éléments à regrouper. La notion de similitude que nous avons implémentée est la suivante :

Définition 5 Soient s_1 et s_2 deux motifs séquentiels. soit $PLSC(s_1, s_2)$ la longueur de la plus longue sous-séquence commune à ces deux motifs séquentiels. le degré de similitude d entre s_1 et s_2 est défini comme $d = \frac{PLSC(s_1, s_2)}{\max(\text{longueur}(s_1), \text{longueur}(s_2))}$.

Avec cette méthode, les motifs ne sont regroupés que si ils ont des URLs en commun. Ainsi la distance entre $\langle(a)(b)\rangle$ et $\langle(b)(c)\rangle$ permet de regrouper ces deux motifs dans une classe. De même la distance entre $\langle(d)(e)\rangle$ et $\langle(d)(f)\rangle$ permet de regrouper ces deux motifs dans une nouvelle classe. Finalement le résultat de cette première méthode de classification sera l'obtention de deux classes : $C_1 = \{\langle(a)(b)\rangle, \langle(b)(c)\rangle\}$ et $C_2 = \{\langle(d)(e)\rangle, \langle(d)(f)\rangle\}$.

5.2. Classification par baisse sensible du support

L'idée directrice est ici la suivante : en baissant le support minimum après une première recherche (soit F_k les fréquents de longueur k produits par cette première recherche), on obtient des résultats (soit F_l les fréquents de longueur l ainsi obtenus) qui pour certains peuvent être nouveaux (des séquences dont aucun item n'appartient à F_k) mais aussi qui peuvent être des sur-ensembles des fréquents contenus dans F_k (lemme 1). Ainsi nous pourrions considérer chaque fréquent obtenu dans F_l comme une classe. Ensuite les fréquents de F_k seront classés en fonction de leur inclusion dans les séquences de F_l .

La propriété 1 a pour origine l'article [AGR 95], qui pose les bases de la recherche des motifs séquentiels.

Propriété 1 Soient s_1 et s_2 deux motifs séquentiels tels que $s_1 \subseteq s_2$, alors $support(s_1) \geq support(s_2)$.

En nous basant sur la propriété 1, nous pouvons affirmer le lemme 1, sur lequel repose la validité de cette classification.

Lemme 1 Soit F_k les fréquents de longueur k produits par une recherche de motifs séquentiels avec un support s_1 . Soit F_l les fréquents de longueur l produits par une recherche de motifs séquentiels avec un support $s_2 < s_1$. Tout motif séquentiel de F_k est inclus dans (ou égal à) un motif séquentiel de F_l .

Preuve : F_l contient toutes les séquences de support s_2 . D'après la propriété 1 et la définition du support F_l contient donc aussi toutes les séquences de support s_1 (puisque $s_2 < s_1$). Toujours d'après la définition du support, soit n le nombre d'items appartenant à F_k et m celui de F_l , alors $n \leq m$. Donc $F_k \subseteq F_l$ et $\forall s_k \in F_k, \exists s_l \in F_l/s_k \subseteq s_l$.

Cette méthode de classification des motifs séquentiels est basée sur ce principe, qui veut que la baisse du support permet de spécialiser les motifs séquentiels obtenus. Ce qui permet ensuite de classer chaque motif dans un des motifs qui le spécialise. Reprenons l'exemple 2. En baissant le support, nous pourrions découvrir les motifs suivants : 0,081% : $\langle (a)(b)(c)(d)(e) \rangle$ et 0,08% : $\langle (b)(c)(d)(e)(f) \rangle$. Nous obtenons alors deux classes, correspondant aux deux motifs obtenus, et nous pouvons classer les motifs de la façon suivante : $C_1 = \{ \langle (a)(b) \rangle, \langle (b)(c) \rangle, \langle (d)(e) \rangle \}$ (les séquences incluses dans $\langle (a)(b)(c)(d)(e) \rangle$) et $C_2 = \{ \langle (b)(c) \rangle, \langle (d)(e) \rangle, \langle (d)(f) \rangle \}$ (les séquences incluses dans $\langle (b)(c)(d)(e)(f) \rangle$). Comme avec la classification naïve, présentée plus haut, nous obtenons avec ces deux classes des sous-logs sensiblement identiques. De plus le défaut de cette classification est de reposer sur la notion de "sensibilité" de la baisse du support, qui est laissée à l'appréciation de l'utilisateur.

5.3. Classification neuronale basée sur une généralisation des motifs

Les performances des méthodes de classification précédentes étant souvent insuffisantes, nous avons dirigé nos travaux vers une troisième méthode de classification [BEN 02] plus efficace pour traiter des séquences d'accès sur le Web initialement pour des systèmes de recommandations en-ligne sur le Web. L'efficacité de cette troisième méthode réside dans l'utilisation d'une méthode neuronale et dans l'utilisation d'un résumé pour chaque motif séquentiel à classer basée sur une généralisation des séquences d'accès Web.

5.3.1. Résumés des motifs séquentiels

Pour caractériser un motif séquentiel, nous choisissons d'utiliser quatre attributs basés sur une *généralisation des pages Web* le constituant, d'une part sur l'aspect "multi-sites" (non détaillé dans cet article) et d'autre part sur l'aspect "rubriques visitées de premier niveau tous sites confondus". Actuellement nous nous basons sur la syntaxe des URLs. Ainsi pour la page (a), nous avons *teaching* comme rubrique de deuxième niveau pour le site *www-sop.inria.fr* ainsi que *axis* pour la rubrique de premier niveau. Enfin (a) sera considéré comme un document du site *www-sop.inria.fr* et de la rubrique de premier niveau *axis*. Les quatre attributs sont : 1) le nombre de rubriques de deuxième niveau explorées par site à partir des différentes pages du motif, 2) le nombre des documents du motif par site, 3) le nombre de rubriques de deuxième niveau explorées par rubrique de premier niveau tous sites confondus à partir des différentes pages du motif ; et enfin 4) le nombre de documents du motif par rubrique de premier niveau (tous sites confondus). Cette structuration des motifs correspond à des vecteurs dans un espace de description de dimension égale à $(2 \times \text{nb sites considérés} + 2 \times \text{nb catégories de premier niveau intervenant dans les motifs séquentiels, tous sites confondus i.e. union des rubriques 1})$. Chaque attribut fait l'objet d'une normalisation entre 0..1 et de l'affectation d'un poids d'importance relativement au contexte choisi.

5.3.2. Méthode neuronale

Nous avons adapté au cas des motifs séquentiels une méthode de classification issue d'un travail réalisé en 2000 par [BEN 02] et intégré dans une plateforme objet CBR*Tools⁷ [JAC 98] d'aide à la réutilisation d'expériences. Notre méthode s'appuie sur un modèle hybride de mémoire inspiré de [MAL 96] pour les besoins d'un contrat France Telecom-INRIA (1998-200) composé d'une partie connexionniste inspirée du modèle ARN2 [AZC 91, GIA 92] et constituée d'un réseau neuronal à base de prototypes avec une structure évolutive et d'une partie de mémoire plate qui contient les différents groupes de motifs.

Définition 6 Un réseau à base de prototypes est caractérisé par 1) une *mémoire* de prototypes, les vecteurs représentant un motif ou un prototype sont mémorisés dans les poids de certaines connexions du réseau 2) un mécanisme d'*apprentissage* qui

7. CBR*Tools URL=<http://www-sop.inria.fr/axis/software.html#tools>

permet de *mémoriser* de nouveaux motifs ou de modifier les prototypes déjà existants, ce mécanisme agit normalement sur les valeurs des poids de certaines connexions; 3) un mécanisme d'*utilisation* qui permet d'utiliser le réseau pour la *remémoration* des motifs les plus "similaires" à un nouveau problème présenté au réseau, le réseau fournit ensuite en sortie la classe des prototypes remémoré(s) comme classification pour le nouveau problème.

La structure d'un réseau à base de prototypes de type ARN2 est évolutive dans le sens où le nombre d'unités cachées (prototypes) dans la couche cachée n'est pas fixé au départ et il peut croître au cours de l'apprentissage. Un prototype est caractérisé par son vecteur de référence, d'une région d'influence et d'un ensemble de motifs qu'il représente.

Architecture d'un réseau à base de prototypes Un réseau à base de prototypes contient trois couches [GIA 92] comme le montre la figure 5) :

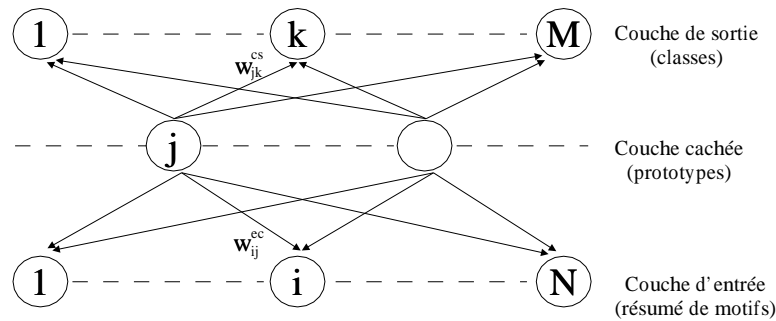


Figure 5. Architecture générale d'un réseau à base de prototypes

1) Une couche d'entrée qui comprend N unités, N correspond à la dimension de l'espace de description d'un motif séquentiel (cf. section 5.3.1).

2) Une couche cachée qui comprend les prototypes décrits dans le même espace de description que les motifs : par exemple, le prototype j sera représenté par le vecteur de pondération : $W_j^{ec} = (w_{1j}^{ec} \dots w_{Nj}^{ec})$, poids des connexions qui lient cette unité aux unités de la couche d'entrée.

3) Une couche de sortie qui comprend M unités correspondant aux M classes ; $w_{jk}^{cs} = 1$ si le prototype j appartient à la classe k ; $w_{jk}^{cs} = 0$ sinon

À chaque prototype, on associe un seuil s_i qui sera modifié pendant l'apprentissage. Ce seuil détermine une région dans l'espace des entrées appelée *région d'influence*. Si un motif introduit dans le réseau tombe dans une région d'influence d'un prototype, alors ce prototype sera activé. Cette région est définie par l'ensemble des vecteurs d'entrée ayant une mesure de distance inférieure à un seuil donné S_e .

Dynamique d'activation Soit m un motif à classer et U l'ensemble des prototypes contenant m dans leur région d'influence. L'activation des prototypes est donnée par l'équation :

$$A_c = 1 \text{ si } D(m, W_c^{ec}) = \min_{j \in U} D(m, W_j^{ec}) : A_c = 0 \text{ sinon}$$

où c est le prototype gagnant, $W_c^{ec} = (w_{1c}^{ec} \dots w_{Nc}^{ec})$ et D une distance euclidienne (avec normalisation). Les poids w_{jk}^{cs} entre la couche cachée et la couche de sortie sont donnés par les formules :

$$w_{jk}^{cs} = 1 \text{ si le prototype } j \text{ appartient à la classe } k; w_{jk}^{cs} = 0 \text{ sinon.}$$

L'activation de la classe k (cas où $O_k=1$) est donnée par : $O_k = \sum_{j \in U} A_j \times w_{jk}^{cs}$

Mode d'apprentissage et de mémorisation Pendant le mode d'apprentissage, un motif m ayant le vecteur de description (m_1, \dots, m_N) est présenté au réseau :

1) Le motif tombe dans *une, plusieurs* ou *aucune* région(s) d'influence, le prototype le plus proche à m est activé, appelons le prototype gagnant C . Par conséquent, une seule classe *au plus* sera activée.

2) Si m n'est tombé dans aucune région d'influence, un nouveau prototype représentant ce motif est ajouté à la couche cachée avec un seuil d'influence.

3) Les poids du prototype gagnant sont modifiés de la façon suivante : $W_c^{ec} = W_c^{ec} + \alpha(t) \times (m - W_c^{ec})$ où $\alpha(t)$ est une suite décroissante avec le temps, elle est donnée par la relation : $\alpha(0) = 1; \alpha(1) = c_1; \alpha(t+1) = \frac{\alpha(t)}{1+c_2 \times \alpha(t)}$ où c_1 et c_2 sont deux constantes. Cette opération permet d'obtenir des prototypes représentatifs pour chaque classe.

Un traitement particulier proposé dans [GIA 92] et basé sur la notion de région d'incertitude a été implanté pour les motifs frontières, évitant la création d'un nombre considérable d'unités cachées dans le réseau.

Illustration sur notre exemple Supposons que 1) nous fusionnions les logs de deux sites www.inria.fr et www-sop.inria.fr pour le log d'origine, 2) structurions en sessions et qu'enfin les motifs extraits ne mettent en jeu que, par exemple, six catégories de niveau un (dont *axis*) tous sites confondus. Soient les quatre motifs séquentiels $\langle(a)(b)\rangle, \langle(b)(c)\rangle, \langle(d)(e)\rangle$ et $\langle(d)(f)\rangle$ utilisés précédemment. Les pages Web "a,...f" intervenant dans ces motifs sont toutes du même site www-sop.inria.fr, ayant *axis* comme rubrique de premier niveau et *teaching* comme rubrique de deuxième niveau. Ainsi, ils seront résumés de la même manière à l'aide des quatre attributs ci-dessous.

oRubrique2ParSite	0, 2	oDocsParSite	0, 2
oRubrique2ParRubrique1	0,0,0,0,2,0	oDocsParRubrique1	0,0,0,0,2,0

Le réseau de prototypes ainsi constitué sur cet exemple comprend une couche d'entrée de dimension 16 (chaque motif donnant lieu à un vecteur de description de di-

mension 16). Comme les motifs choisis ont le même résumé, nous obtenons une seule classe comme le montre la figure 6.

Motifs d'origine	Classification		
	Naive	Baisse du support	Reseaux de neurones
<(a)(b)>	<(a)(b)>, <(b)(c)>	<(a)(b)>, <(b)(c)>, <(d)(e)>	<(a)(b)>, <(b)(c)>, <(d)(e)>, <(d)(f)>
<(b)(c)>			
<(d)(e)>	<(d)(e)>, <(d)(f)>	<(b)(c)>, <(d)(e)>, <(d)(f)>	
<(d)(f)>			

Figure 6. Résultats des méthodes de classification envisagées.

6. Expérimentations

Les logs sur lesquels nous avons effectué nos expérimentations ont les caractéristiques décrites par les tableaux de la figure 7. Ils portent sur une période d'un mois (Février 2003) pour le site du siège, deux mois (Février et Mars 2003) pour le site de Sophia Antipolis et représentent 2,1 Go et 3 Go. Les programmes d'extraction sont réalisés en C++ sur une machine de type PC équipée de processeur pentium 2,1 Ghz et exploitée par un système Linux (2.4).

Caractéristique	www.inria.fr	www-sop.inria.fr
Nombre de lignes	11 637 62	15 158 076
Nombre de sessions	432 396	564 870
Nombre d'URLs (filtrées)	68 732	82 372
Longueur moyenne des sessions	6.3	4.4
Nombre moyen d'URLs dans les sessions	7.2	6.3

Figure 7. Caractéristiques des fichiers logs

Lors de nos expérimentations sur ces logs, nous avons pu mettre en évidence des comportements fréquents, dont la représentativité relative (support du comportement par rapport au nombre total de sessions du log) était de plus en plus faible. Cette baisse de la représentativité étant proportionnelle au nombre de divisions effectuées pour isoler ce comportement. Les tableaux de la figure 8 recensent quelques-uns des motifs découverts. Voici la description de ces comportements :

C1 : avec le préfixe commun **travailler/opportunites/** :
 <(it.fr.html) (ita/missions.fr.html) (ita/concoursit.fr.html)
 (ita/annales2001/index.fr.html)>. Ce comportement est relatif aux offres de postes d'ITA.

C2 : <(travailler/opportunites/chercheurs.fr.html)
 (travailler/opportunites/chercheurs/concoursr2.fr.html)
 (recherche/equipes/index.fr.html) (recherche/equipes/listes/index.fr.html)>. Ce com-

www.inria.fr

Id	P	Sessions	S1	S2	T1	T2	R1	R2
C1	1	28740	10%	0.6%	34	23	27	70
C2	2	4473	28%	0.28%	58	1364	188	23035
C3	3	280	85%	0.04%	73	?	14	?
C4	4	198	13%	0.006%	97	?	6	?

www-sop.inria.fr

Id	P	Sessions	S1	S2	T1	T2	R1	R2
C5	1	19686	10%	0.34%	24	24	1	35
C6	2	3252	4%	0.02%	51	?	138	?
C7	2	1551	29%	0.08%	74	16681	20	2482
C8	3	381	23%	0.01%	99	?	6	?

Description des paramètres

Id	Identifiant du comportement
P	Niveau de profondeur (nombre de divisions nécessaires pour l'obtenir)
Sessions	Nombre de sessions du sous-log dont ce comportement est extrait
S1	Support relatif du comportement (support sur le sous-log)
S2	Support absolu du comportement (support sur le log d'origine)
T1	Temps (s) relatif nécessaire pour obtenir ce comportement (sur le sous-log)
T2	Temps (s) absolu pour obtenir ce comportement avec le support S2 (sur le log d'origine)
R1	Taille relative du résultat (nombre de comportements sur le sous-log)
R2	Taille absolue du résultat avec le support S2 (nb de comportement sur le log d'origine)

Figure 8. Caractéristiques des motifs découverts

portement est relatif aux offres de postes de chercheurs à l'INRIA. Les utilisateurs consultent la page des offres, celle des concours puis celles décrivant les équipes.

C3 : <(scripts/root.exe) (c/winnt/system32/cmd.exe) (..%255c../..%255c../winnt/system32/cmd.exe) (..%255c../..%255c../c1%1c../..%c1%1c../..%c1%1c../winnt/system32/cmd.exe) (winnt/system32/cmd.exe) (winnt/system32/cmd.exe) (winnt/system32/cmd.exe)>. Ce comportement est typique d'une attaque pirate. Après l'avoir isolé, nous avons consulté le responsable de la sécurité du réseau de l'unité de Sophia Antipolis, qui nous a confirmé qu'un tel enchaînement d'appels aux scripts ne pouvait être qu'une attaque y recherchant une faille. Généralement ces attaques sont préprogrammées et les pirates utilisent les mêmes programmes d'attaques, ce qui confère à ce comportement un support relatif très élevé (plus de 80%).

C4 : avec le préfixe commun **rapportsactivite/RA95/omega/** : <(node10.html) (node11.html) (node12.html) (node13.html)>. Ce comportement reflète l'activité des utilisateurs qui se sont intéressés au rapport d'activité du projet

omega. Cependant, cette activité porte sur le rapport de 1995. Cela pourrait s'expliquer par le fait que les moteurs de recherche disponibles sur Internet (extérieurs à l'INRIA) renvoient sur les rapports d'*omega* des années 95, 97 et 98 quand l'objet de la recherche correspond au thème de ce projet. Cela illustre nos propos, en introduction, disant que les moteurs de recherche extérieurs sont un facteur incontournable.

C5 : <(koala/colas/mouse-wheel-scroll) (koala/colas/mouse-wheel-scroll)>. Parmi les comportements extraits, tous ne sont pas d'un intérêt flagrant. Celui-là montre simplement une répétition de l'URL pointant sur un logiciel développé par un membre de l'unité de Sophia Antipolis. Ce logiciel étant fortement demandé, il ressort parmi les premiers comportements.

C6 : avec le préfixe commun **robotvis/personnel/zhang/Publis/Tutorial-Estim/** : <(Main.html) (node3.html) (node4.html) (node5.html) (node6.html) (node7.html)>. Cet exemple, comme le suivant, reflète le comportement d'utilisateurs venus consulter des pages de cours ou de tutoriaux réalisés par des membres du personnel de l'unité.

C7 : avec le préfixe **mascotte/personnel/Sebastien.Choplin/cours/iut-infocom/excel/** : <(exercices.html) (exo1.xls) (exo2.xls) (exo3.xls) (exo4.xls) (exo5.xls)>.

C8 : avec le préfixe **epidaure/Demonstrations/foie3d/** : <(endo4.html) (endo5.html) (endo6.html) (endo8.html) (endo9.html) (endo10.html) (endo11.html) (endo12.html)>.

Des comportements significatifs avec un support vraiment très faible La liste des comportements ainsi découverts couvre plus de 50 objectifs de navigation distincts sur le site du siège, et plus de 100 sur celui de l'unité de Sophia Antipolis. Nous avons reporté ici 8 objectifs différents, qui vont des offres de postes aux consultations de pages de cours, en passant par les activités de recherche et les tentatives de piratage. Les comportements reportés ci-dessus ont donc pour but d'illustrer le type de comportements obtenus, mais aussi le succès de notre méthode pour découvrir les sortes de "niches" décrites dans cet article. À savoir des comportements homogènes pour une minorité d'utilisateurs, mais que l'on ne pourrait pas découvrir sans tenir compte de leur représentativité très faible sur le log global.

Une méthode performante Pour confirmer nos propos, nous avons reporté dans la table de la figure 8 d'un côté le temps $T1$ nécessaire pour l'extraction du motif avec notre méthode et un support $S1$ (temps cumulé de l'extraction de motifs séquentiels à chaque étape de la division) et d'un autre côté le temps $T2$ nécessaire pour obtenir ce même motif avec une méthode classique, un support $S2$ correspondant à la représentativité de ce motif. Par exemple, si $S1$ est de 10% pour un sous-log contenant 100 sessions et que le log original contient 100 000 sessions alors $S2$ vaudra 0.1%. Nous avons comparé les temps de réponse avec $S1$ sur le sous-log et $S2$ sur le log d'origine. Très souvent, le temps $T2$ nécessaire pour obtenir les résultats avec le support $S2$ est tel que nous n'avons pas pu obtenir les résultats. Le '?' traduit un échec de la méthode d'extraction classique dû à la faiblesse du support et donc à la complexité du processus d'extraction. Cela traduit également l'incapacité d'une méthode d'extraction de motifs séquentiels classique à obtenir ces motifs.

Des résultats plus faciles à exploiter De plus, avec un support S^2 si faible les résultats peuvent atteindre une taille si grande qu'ils en deviendraient difficiles à exploiter. Grâce à notre méthode de division, les résultats sont classés au fil de leur découverte, en fonction de l'objectif de navigation du sous-log qui leur correspond.

7. Conclusion et perspectives

Dans cet article, nous avons présenté une méthode destinée à découvrir les comportements fréquents (jusqu'au plus minoritaires) des utilisateurs d'un site web, sous forme de motifs séquentiels. L'originalité de notre approche se situe dans la procédure de divisions successives du log, afin d'isoler les comportements sous forme de classes. Pour cela nous avons développé des méthodes de classification spécifiques aux motifs séquentiels. La plus pertinente se base sur un algorithme exploitant des réseaux de neurones. Nous avons ensuite proposé une étude des possibilités offertes par notre approche, ses limites et les frontières qu'elle permet de franchir. Parmi ces frontières, nous avons pu souligner la capacité de notre approche à extraire des comportements relatifs à des minorités d'internautes. Ces comportements ont des caractéristiques typiques des enjeux du data mining comme leur cohérence avec la communauté à laquelle ils sont associés mais aussi leur grande significativité. Cependant, notre approche permet de s'affranchir des difficultés relatives au très faible support de ces comportements. Enfin notre réflexion se porte sur d'autres types de problèmes que l'on pourrait aborder de cette manière, afin de préciser la situation du data mining à la frontière entre la quantité de données et la qualité des résultats.

8. Bibliographie

- [AGR 93] AGRAWAL R., IMIELINSKI T., SWAMI A., « Mining Association Rules between Sets of Items in Large Databases », *Proceedings of the 1993 ACM SIGMOD Conference*, Washington DC, USA, May 1993, p. 207-216.
- [AGR 95] AGRAWAL R., SRIKANT R., « Mining Sequential Patterns », YU P., CHEN A. P., Eds., *Proceedings of the Eleventh International Conference on Data Engineering, March 6-10, 1995, Taipei, Taiwan*, IEEE Computer Society, 1995, p. 3-14.
- [AZC 91] AZCARRAZA A., GIACOMETTI A., « A Prototype-Based Incremental Network Model for Classification Task », *4th International Conference on Neural Networks and Their Applications*, Nimes, France, 1991.
- [BEN 02] BENEDEK A., TROUSSE B., « Adaptation of Self-Organizing Maps for CBR case indexing », *Proceedings of the 4th International Workshop on Symbolic and Numeric Algorithms for Scientific Computing*, Timisoara, Romania, October 2002, p. 31-45, and also 27th Annual Conference of the Gesellschaft für Klassifikation, 12-14 March 2003, Cottbus, Germany.
- [BON 01] BONCHI F., GIANNOTTI F., GOZZI C., MANCO G., NANNI M., PEDRESCHI D., RENSO C., RUGGIERI S., « Web Log Data Warehousing and Mining for Intelligent Web Caching », *Data Knowledge Engineering*, vol. 39, n° 2, 2001, p. 165-189.

- [CAD 00] CADEZ I. V., HECKERMAN D., MEEK C., SMYTH P., WHITE S., « Visualization of Navigation Patterns on a Web Site Using Model-based Clustering », *In Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, Massachusetts, 2000, p. 280-284.
- [COO 99] COOLEY R., MOBASHER B., SRIVASTAVA J., « Data Preparation for Mining World Wide Web Browsing Patterns », *Knowledge and Information Systems*, vol. 1, n° 1, 1999, p. 5-32.
- [FAY 96] FAYAD U., PIATETSKY-SHAPIRO G., SMYTH P., UTHURUSAMY R., Eds., *Advances in Knowledge Discovery and Data Mining*, AAAI Press, Menlo Park, CA, 1996.
- [FU 00] FU Y., SANDHU K., SHIH M., « A Generalization-Based Approach to Clustering of Web Usage Sessions », *Proceedings of the 1999 KDD Workshop on Web Mining*, San Diego, CA. Springer-Verlag, vol. 1836 de LNAI, Springer, 2000, p. 21-38.
- [GIA 92] GIACOMETTI A., « Modèles Hybrides de l'Expertise », novembre 1992, Thèse de doctorat, ENST Paris.
- [HAN 01] HAN J., KAMBER M., *Data Mining, Concepts and Techniques*, Morgan Kaufmann, 2001.
- [JAC 98] JACZYNSKI M., « Modèle et plate-forme à objets pour l'indexation des cas par situation comportementales : application à l'assistance à la navigation sur le Web », décembre 1998, Thèse de doctorat, Université de Nice Sophia-Antipolis.
- [MAL 96] MALEK M., « Un modèle hybride de mémoire pour le raisonnement à partir de cas », octobre 1996, Thèse de doctorat, Université Joseph Fourier.
- [MAS 00] MASSEGLIA F., PONCELET P., CICHETTI R., « An efficient algorithm for Web usage mining », *Networking and Information Systems Journal (NIS)*, , April 2000.
- [MOB 02] MOBASHER B., DAI H., LUO T., NAKAGAWA M., « Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization », *Data Mining and Knowledge Discovery*, vol. 6, n° 1, 2002, p. 61-82.
- [PER 98] PERKOWITZ M., ETZIONI O., « Adaptive Web Sites : Automatically Synthesizing Web Pages », *AAAI/IAAI*, 1998, p. 727-732.
- [SPI 99] SPILIOPOULOU M., FAULSTICH L. C., WINKLER K., « A Data Miner analyzing the Navigational Behaviour of Web Users », *Proceedings of the Workshop on Machine Learning in User Modelling of the ACAI'99 Int. Conf.*, Crete, Greece, July 1999.
- [TAN 01] TANASA D., TROUSSE B., « Web Access Pattern Discovery and Analysis based on Page Classification and on Indexing Sessions with a Generalised Suffix Tree », *Proceedings of the 3rd International Workshop on Symbolic and Numeric Algorithms for Scientific Computing*, Timisoara, Romania, October 2001, p. 62-72.
- [W3C 95] W3C, « httpd-log files », <http://www.w3.org/Daemon/User/Config/Logging.html>, 1995.
- [ZHA 96] ZHANG T., RAMAKRISHNAN R., LIVNY M., « BIRCH : An Efficient Data Clustering Method for Very Large Databases », JAGADISH H. V., MUMICK I. S., Eds., *Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, Montreal, Quebec, Canada, June 4-6, 1996, ACM Press, 1996, p. 103-114.