Des séquences aux tendances

Céline Fiot, Florent Masseglia, Anne Laurent, Maguelonne Teisseire

Des séquences aux tendances

Céline Fiot* — Florent Masseglia* — Anne Laurent — Maguelonne Teisseire****

** INRIA Sophia Antipolis Méditerranée*
2004 route des Lucioles – B.P.93, F-06902 Sophia Antipolis Cedex
{*celine.fiot, florent.masseglia*}*@sophia.inria.fr*

*** LIRMM - Univ. Montpellier II - CNRS*
161 rue Ada, F-34392 Montpellier Cedex 5
{*laurent, teisseire*}*@lirmm.fr*

RÉSUMÉ. *Les données temporelles peuvent être traitées de nombreuses façons afin d'en extraire des connaissances. La découverte de motifs séquentiels met en évidence des sous-séquences fréquentes contenues dans des séquences d'enregistrements annotés temporellement. L'analyse des accès à un site web permet par exemple de découvrir que* "5% des utilisateurs accèdent à la page register.php puis à la page help.html". *Cependant, les motifs séquentiels ne permettent pas d'extraire des tendances temporelles, du type* "une augmentation du nombre de requêtes au formulaire d'inscription précède souvent une augmentation des requêtes à la page d'aide quelques secondes plus tard". *Dans cet article, nous proposons d'extraire des motifs caractérisant ces évolutions fréquentes grâce à deux algorithmes,* TED *et* EVA. *Nous présentons notre approche, implémentée et testée sur des données réelles.*

ABSTRACT. *Temporal data can be handled by different techniques for discovering specific knowledge. Sequential pattern mining allows discovering frequent sequences embedded in temporally annotated records. In the access data of a Web site, one may, for instance, discover that* "5% of the users request the page register.php and then request the page help.html". *However, sequential patterns do not allow extracting temporal tendencies. By means of temporal tendency mining, one may discover in the same access data that* "An increasing number of requests to registration.php during a short period preceeds an increasing number of requests to faq.html, after a very short period". *In this paper, we define evolution patterns that allow discovering such knowledge. We define evolution patterns and introduce our algorithms* TED *and* EVA.

MOTS-CLÉS : *Fouille de données, motifs séquentiels, tendances, évolution, sous-ensembles flous.*

KEYWORDS: *Data mining, Sequential Patterns, Trends, Evolution, Tendencies, Fuzzy Sets Theory.*

1. Introduction

De nombreuses applications (surveillance réseaux, analyse de consultation de site web, gestion de clientèle) collectent des données annotées temporellement. Ces données peuvent être exploitées par des techniques de fouille de données spécifiques utilisant ces annotations temporelles, comme, par exemple, la recherche de motifs séquentiels. Les motifs séquentiels sont des séquences fréquentes, contenues dans des bases de données dont les enregistrements, associés à des objets, ont plusieurs valeurs et sont ordonnés grâce à une estampille temporelle. Un exemple de telles données se trouve dans les séquences d'accès successifs d'internautes à des pages webs.

Etant donné que la plupart des bases de données contiennent à la fois des attributs symboliques et de nombreuses informations numériques quantitatives, telles les paramètres de fonctionnement (température, vitesse, ...) de machines ou de véhicules ou encore la durée ou le débit de connexion à un site web, des travaux ont généralisé l'extraction des motifs séquentiels. Ainsi, (Hong *et al.,* 2001, Chen *et al.,* 2001, Fiot *et al.,* 2007b) utilisent une discrétisation des attributs numériques en sous-ensembles flous afin d'extraire des *motifs séquentiels flous*. De tels motifs permettent de découvrir des connaissances complémentaires, liées aux valeurs numériques fréquemment observées dans les données et à leurs corrélations. Alors qu'un motif classique indiquerait que "*Dans 80% des cas, l'allumage du moteur précède une vitesse non nulle*", un motif séquentiel flou pourrait décrire plus explicitement ce schéma par "*Dans 80% des cas, un nombre de tours élevé du moteur s'accompagne d'une température élevée, et précède une vitesse moyenne*".

Toutefois, ces motifs ne permettent pas d'étudier l'évolution des valeurs numériques des attributs ni leurs corrélations. De plus un motif séquentiel flou décrivant qu'"*un nombre de tours élevé du moteur précède un nombre de tours faible*" n'apporte pas d'information concernant la durée de cette diminution ni sur la dynamique ou l'intensité du changement de régime.

Dans cet article, nous proposons donc une méthode d'extraction des tendances temporelles sous la forme de *motifs d'évolution*. En reprenant l'exemple précédent concernant le comportement de véhicules, un motif d'évolution pourrait être par exemple *"Lorsque le régime moteur augmente fortement, après une très courte période, la vitesse du véhicule augmente lentement pendant une courte période*". L'intérêt se trouve également dans le fait qu'une augmentation de la vitesse peut se produire dans des tranches de vitesses basses ou hautes (de 10 à 20 km/h ou de 150 à 190 km/h).

Cette approche peut être vue comme une extension des principes utilisés pour l'extraction de motifs séquentiels flous. Cependant, la modélisation de telles connaissances requiert la gestion d'un grand nombre d'éléments, à la fois en terme d'attributs et d'enregistrements. Notre objectif est donc de proposer un outil permettant d'extraire des tendances existant dans des bases de séquences quantitatives.

Dans la suite de cet article, nous définissons les concepts liés à l'extraction des motifs séquentiels flous dans la section 2. Puis dans la section 3, nous introduisons notre

approche, en commençant par définir les motifs d'évolution. La section 4 s'intéresse ensuite aux algorithmes développés et la section 5 décrit quelques résultats obtenus lors d'expérimentations sur des logs d'accès web. Nous concluons dans la section 6.

2. Motifs et tendances

2.1. *Motifs séquentiels*

Les motifs séquentiels ont initialement été proposés par (Agrawal *et al.*, 1995) et reposent sur la notion de *séquence fréquente maximale*.

Considérons une base de données décrivant un ensemble d'objets. Chaque enregistrement correspond à un triplet *(id-objet, id-date,* $\{$*items*$\}$*)* qui caractérise l'objet auquel est rattaché l'enregistrement, ainsi que la date et les *items* correspondants. Soit $\mathcal{I} = \{i_1, i_2, \cdots, i_m\}$ l'ensemble des *items* de la base. Un *itemset* est un ensemble non vide et non ordonné d'items i_j, noté $(i_1, i_2, \ldots, i_k)$. Une *séquence* s se définit alors comme une liste ordonnée non vide d'itemsets s_j qui sera notée $\langle s_1 s_2 \cdots s_p \rangle$. Une n-séquence est une séquence de taille n, c'est-à-dire composée de n items.

Une séquence S'=$\langle s'_1 s'_2 \cdots s'_m \rangle$ est une *sous-séquence* de S=$\langle s_1 s_2 \cdots s_p \rangle$ s'il existe des entiers $a_1 < a_2 < \cdots < a_m$ tels que $s'_1 \subseteq s_{a_1}, s'_2 \subseteq s_{a_2}, \cdots, s'_m \subseteq s_{a_m}$.

Exemple 1 *Si un client achète les produits* e*,* a*,* k*,* u *et* f *selon la séquence* $S = \langle (e)\ (a\ k)\ (u)\ (f) \rangle$*, cela signifie qu'il a d'abord acheté le produit* e*, puis les produits* a *et* k *ensemble, ensuite le produit* u *et finalement* f*.* S *est une 5-séquence. De plus,* $S' = \langle (a)(f) \rangle$ *est une sous-séquence de* S *car* $(a) \subseteq (a\ k)$ *et* $(f) \subseteq (f)$*. Par contre* $\langle (a)(k) \rangle$ *n'est pas une sous-séquence de* $< (a\ k) >$*, ni l'inverse.*

Les enregistrements de la base sont regroupés par objet et ordonnés chronologiquement, définissant ainsi des *séquences de données*. Un objet o *supporte* une séquence S si elle est une sous-séquence de la séquence de données de o. La *fréquence* d'une séquence est définie comme le pourcentage d'objets de la base qui supporte S. Une séquence est *fréquente* si sa fréquence est au moins égale à une valeur $minFreq$ spécifiée par l'utilisateur. La recherche de motifs séquentiels dans une base de séquences consiste alors à trouver toutes les séquences fréquentes et maximales, i.e. non incluses dans d'autres, (Agrawal *et al.*, 1995). Chacune de ces séquences fréquentes maximales est un *motif séquentiel*.

Des extensions ont été proposées pour intégrer les valeurs numériques associées aux items (Hong *et al.*, 2001, Chen *et al.*, 2001, Fiot *et al.*, 2007b), la prise en compte de contraintes temporelles (espacement des différents évènements d'une séquence, rapprochement d'évènements proches en une même date...) (Srikant *et al.*, 1996, Masseglia *et al.*, 2004), ou encore la présence de valeurs manquantes dans la base de données (Fiot *et al.*, 2007a).

2.2. *Théorie des sous-ensembles flous*

La théorie des sous-ensembles flous, introduite par (Zadeh, 1965), autorise l'appartenance partielle à une classe, et donc la gradualité de passage d'une situation à une autre. Dans ce cadre, un objet peut appartenir à un ensemble et en même temps à son complément.

On considère par exemple l'univers X des tailles possibles d'un individu. Un *sous-ensemble flou* A (par exemple *Petit* ou *Grand*) est défini par une fonction d'appartenance μ_A qui décrit le degré avec lequel chaque élément $x \in X$ appartient à A. Cette fonction est décrite sur le domaine des valeurs de X et associe chaque valeur à un degré compris entre 0 et 1. Ainsi, un individu de 1m63 pourra à la fois être grand et petit avec, par exemple, un degré $\mu_{Petit}(x = 1m63)$=0.7 pour le sous-ensemble flou *Petit* et de 0.3 pour le sous-ensemble flou *Grand*.

Les *opérateurs* en logique floue sont une généralisation des opérateurs classiques. On considère notamment la négation, l'intersection et l'union. L'opérateur $\top$ ou t-norme (norme triangulaire) est l'opérateur binaire d'intersection : $\mu_{A\cap B}(x) = \top(\mu_A(x), \mu_B(x))$. L'opérateur $\perp$ ou t-conorme (conorme triangulaire) est l'opérateur binaire d'union : $\mu_{A\cup B}(x) = \perp(\mu_A(x), \mu_B(x))$. Il existe différents opérateurs pouvant être considérés comme t-normes ou t-conormes. Le choix de ces opérateurs dépend de leurs propriétés et de l'application.

2.3. *Motifs séquentiels flous*

Afin de prendre en compte les informations numériques quantitatives, plusieurs travaux ont proposé de partitionner chaque attribut numérique en plusieurs sous-ensembles flous (Hong *et al.,* 2001, Chen *et al.,* 2001, Fiot *et al.,* 2007b). La base de données quantitatives est convertie en une base de degrés d'appartenance, qui est ensuite fouillée pour extraire des motifs fréquents.

Les concepts d'item et d'itemset ont été redéfinis. Un *item flou* est un couple $[x, a]$ composé d'un item/attribut x et d'un sous-ensemble flou a, décrit sur l'univers des quantités associées à x. Un *itemset flou* est un ensemble non vide et non ordonné d'items flous. L'itemset flou (X, A) décrit un ensemble X d'items x_i, chacun associé à un sous-ensemble flou a_i, regroupés dans l'ensemble A. Enfin, une *g-k-séquence* floue $S = \langle s_1 \cdots s_g \rangle$ est une séquence composée de g itemsets flous $s_i = (X, A)$ regroupant au total k items flous $[x, a]$.

Exemple 2 *$[bonbon, peu]$ est un item flou où peu est un sous-ensemble flou défini par une fonction d'appartenance sur l'univers des quantités possibles de l'item $bonbon$. $([bonbon, peu][soda, beaucoup])$ est un itemset flou, noté $((bonbon, soda)(peu, beaucoup))$. La séquence $\langle([soda, peu]\ [bonbon, peu])\ ([chips, peu])\rangle$ regroupe 3 items flous dans 2 itemsets, c'est une 2-3-séquence floue.*

Dans la suite de cet article, nous utiliserons les notations suivantes : $\mathcal{O}$ représente l'ensemble des objets de la base et $\mathcal{R}_o$ l'ensemble des enregistrements d'un objet o. $\varrho[x]$ dénote la valeur numérique associée à l'attribut x pour l'enregistrement ϱ. Un enregistrement ϱ dans une base de séquences floues (ou base de degrés d'appartenance) est constitué des degrés d'appartenance des valeurs numériques des différents attributs à chaque sous-ensemble flou, par exemple $r(x,a) = \mu_a(\varrho[x])$ donne la valeur de l'enregistrement r pour l'item flou (x,a). Il représente le degré d'appartenance de la quantité $\varrho[x]$ de l'item/attribut x au sous-ensemble flou a.

La fréquence d'une séquence floue S se calcule alors par la formule [1] :

$$FFreq(S) = \frac{\sum_{o \in \mathcal{O}} \varphi(S,o)}{|\mathcal{O}|} \quad [1]$$

où $\varphi(S,o)$ est le degré d'inclusion de S dans la séquence de données de l'objet o.

Ce degré est calculé en considérant la meilleure occurrence, i.e. l'occurrence correspondant au meilleur degré, de la liste ordonnée des itemsets de S. Il est obtenu par le calcul suivant [2] :

$$\varphi(S,o) = \underline{\perp}_{\varsigma \subseteq \zeta_o | S = \varsigma = \langle s_1 \ldots s_i \ldots s_k \rangle} \overline{\top}_{s_1 \ldots s_k} (\overline{\top}_{j \in s_i} \mu(j)) \quad [2]$$

où k est le nombre d'itemsets dans S, ζ_o l'ensemble des séquences incluses dans la séquence de données de l'objet o et $\overline{\top}$ et $\underline{\perp}$ sont des opérateurs de t-norme et t-conorme généralisés aux cas n-aires. En pratique, nous utilisons les t-normes et t-conormes de Zadeh, $\min$ et $\max$.

2.4. *Tendances dans les séries temporelles*

De nombreux travaux se sont intéressés à l'analyse de séries temporelles, proposant des techniques de segmentation, de description ou de représentation de ces données (Sklansky *et al.,* 1980, Hugueney *et al.,* 2001). Plus particulièrement, (Kacprzyk *et al.,* 2006, Kacprzyk *et al.,* 2007) proposent de caractériser de façon intelligible des tendances dans des séries temporelles univariées, grâce à l'utilisation de sous-ensembles flous en tant que descripteurs. Dans (Keogh *et al.,* 2005), il s'agit de détecter des anomalies au sein de séries temporelles.

Considérant des séries temporelles multivariées, la plupart des approches ont pour but d'analyser les évolutions parallèles, similaires ou induites, de plusieurs séries temporelles, chacune réduite à un attribut numérique, en utilisant des techniques de segmentation ou d'alignements multiples (Keogh *et al.,* 1999).

Cependant toutes les données ne peuvent être considérées comme des séries temporelles, même multivariées. En effet, d'une part, tous les enregistrements ne contiennent pas nécessairement des valeurs pour l'ensemble des attributs, d'autre part les relevés peuvent être réalisés de façon discontinue, à des intervalles de temps

irréguliers, ou n'ont pas lieu sur les mêmes périodes ni les mêmes attributs selon les séquences de données.

C'est pourquoi nous nous sommes intéressés plus spécifiquement aux propositions concernant la découverte de tendances dans des bases de séquences. Or, exceptés (Dong *et al.*, 1999, Lent *et al.*, 1997) approchant cette problématique, il n'existe pas de méthode générique permettant de découvrir des évolutions fréquentes ou encore d'analyser des durées dans des données séquentielles.

3. TED : modélisation de tendances dans des attributs quantitatifs

L'objectif de ce travail est de permettre l'expression des évolutions des valeurs d'attributs quantitatifs, communes à plusieurs objets d'une base de séquences. Par exemple, un *motif d'évolution* pourrait être *l'augmentation forte du régime moteur est suivie après un court instant par l'augmentation rapide de la vitesse du véhicule*.

Nous utilisons également le formalisme de séquences floues afin d'extraire une information additionnelle, permettant d'exprimer les durées liées à ces évolutions. Afin d'extraire de tels motifs, nous proposons de transformer les séquences de données quantitatives originales en séquences de variations, grâce à l'algorithme TED. Ces séquences seront ensuite fouillées par notre algorithme EVA. Dans cette section, nous décrivons les concepts de séquences d'évolution et détaillons notre approche pour extraire des motifs d'évolution.

3.1. *Principe général*

Le principe de notre approche est décrit par la Figure 1. Tout d'abord, la base de données quantitatives (Qdb, Tableau 1) est convertie en une *base de variations*.

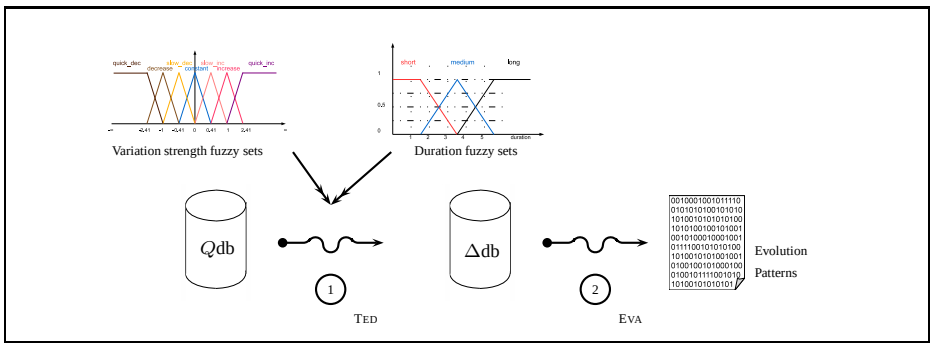


Figure 1. *Principe de l'approche*

Ensuite, sur le même principe que le prétraitement réalisé lors de la découverte de motifs séquentiels flous, cet ensemble de données est converti en une base de degrés

d'appartenance (Δdb, Tableau 2). Les sous-ensembles flous utilisés pour réaliser ce prétraitement sont prédéfinis, soit automatiquement, par clustering par exemple, soit à partir d'une connaissance experte. Deux types de sous-ensembles flous sont nécessaires : ceux donnant les termes linguistiques qui décrivent la force des variations, les autres décrivant la durée des tendances. Ces étapes de conversion sont présentées plus en détails dans les sous-sections 3.3 et 3.4. La base de degrés d'appartenance Δdb est la *base de tendances*. C'est ce jeu de données qui est ensuite fouillé en vue de l'extraction des motifs d'évolution (étape 2 sur la Figure 1), comme cela est décrit dans la sous-section 4.1.

Exemple 3 *Soit la séquence* $\langle([x,4])([x,3][y,5][z,8])([x,2][y,4][z,10])\ ([y,6])\rangle$ *caractérisant le nombre de connexions aux URL* x*,* y *et* z*, durant plusieurs sessions d'une même IP. Le tableau 1 représente cette séquence sous la forme d'une base quantitative, Qdb. Cette base contient quatre enregistrements ordonnés pour l'IP considérée.*

Tableau 1. *Une base de données quantitatives.*

	date	**x**	**y**	**z**	
d1	4	4			r^1
d2	7	3	5	8	r^2
d3	8	2	4	10	r^3
d4	10		6		r^4

Le tableau 2, sous-section 3.3, décrit la base de tendances Δ*db issues de la base quantitative 1, après l'exécution de l'algorithme* TED.

3.2. *Motifs d'évolution*

Les séquences de données sont modélisées de telle sorte que chaque item exprime une variation, par exemple *"un nombre croissant de requêtes"*.

Chaque enregistrement de cette base de données de tendances représente l'évolution entre deux enregistrements rattachés au même objet dans le jeu de données original. Les items flous d'un enregistrement de cette base de tendances ont été créés à partir d'un même couple d'enregistrements de la base de données initiale. Nous définissons une *base de données de tendances* comme un ensemble de séquences de données composées d'*items d'évolution*.

Un *item d'évolution* représente une tendance, augmentation, baisse ou stagnation, d'un attribut quantitatif. L'utilisation des sous-ensembles flous permet également de considérer l'intensité de cette variation. Ainsi, un item d'évolution est défini comme un item flou spécifique $[x, v]$ dans lequel x est un attribut quantitatif de la base originale et v un sous-ensemble flou représentant à la fois la tendance et l'intensité de la variation de la valeur de x. Par exemple, avec l'utilisation des granules

de tendances décrite par la Figure 2(a), tirées de (Kacprzyk *et al.,* 2006), un item d'évolution *[nb_faq.html, quick_inc]* pourrait signifier que "*le nombre de requêtes à la page faq.html augmente rapidement*". Chaque item d'évolution est associé à un degré d'appartenance qui décrit plus précisément l'intensité de la variation.

Un itemset d'évolution peut alors être défini comme un ensemble non vide et non ordonné d'items d'évolution. Il représente l'évolution conjointe (*co-évolution*) de plusieurs attributs, i.e. les variations de plusieurs attributs sur une même période. Une séquence d'évolutions, liste ordonnée d'itemsets d'évolutions, décrit alors des tendances successives dans la base de données quantitatives.
Un intemset d'évolution sera noté avec des parenthèses $([x, inc][y, dec])$ et une séquence par des angles $\langle([x, inc][y, dec])\,([z, q\_inc])\rangle$.

Ainsi une base de données de tendances est une base de données de degrés d'appartenance dans laquelle les items flous décrivent l'intensité des variations des valeurs d'attributs quantitatifs. Cependant, ce jeu de données ne peut être considéré comme une base de séquences floues qui pourrait être analysée par des algorithmes d'extraction de motifs séquentiels flous, comme décrit dans (Hong *et al.,* 2001, Chen *et al.,* 2001, Fiot *et al.,* 2007b).

En effet, chaque enregistrement de cette base est identifié par un objet mais correspond à deux estampilles temporelles. L'une correspond à la date de début de la variation observée, la seconde à sa date de fin. De plus, il peut y avoir pour un même objet plusieurs enregistrements ayant soit la même date de début soit la même date de fin, puisque le jeu de données de variations contient les évolutions existant entre chaque paire d'enregistrements de la base de données quantitative Qdb ayant des attributs en commun. C'est pourquoi l'extraction de motifs d'évolution n'est pas une simple application de la recherche de motifs séquentiels flous à la suite d'un prétraitement spécifique.

Dans la sous-section suivante, nous décrivons comment la base de données d'évolution est construite grâce à l'algorithme TED, qui génère les enregistrements de variations. Dans la section 4, nous détaillons l'algorithme EVA qui permet l'extraction des motifs d'évolution grâce à une structure de graphe de séquence.

3.3. *Base de données de tendances*

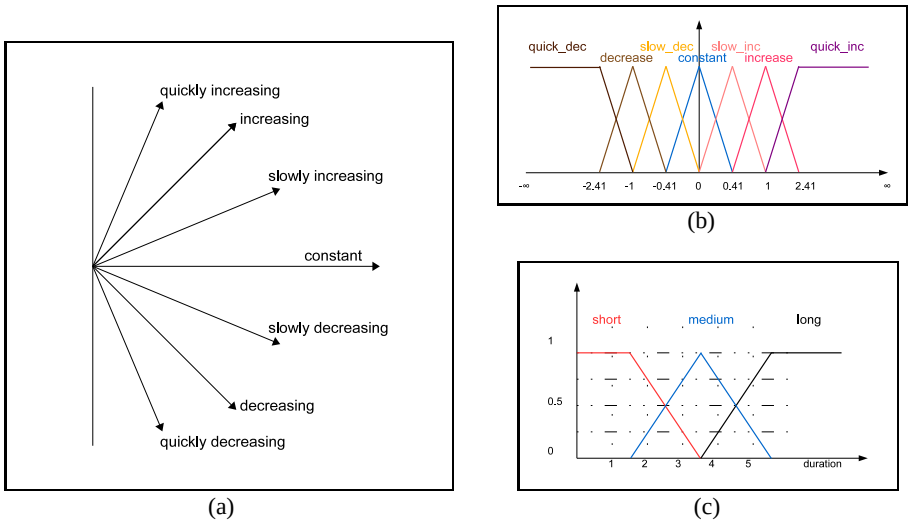
Chaque item d'évolution $[x, v]$ de la base de tendances Δdb représente la variation d'un attribut quantitatif entre deux enregistrements successifs r^1 et r^2, rattachés au même objet o dans la base quantitative Qdb. Chaque enregistrement de Δdb est construit par combinaison de deux enregistrements de Qdb. Plus précisément, pour chaque paire ordonnée d'enregistrements r^i et r^j d'une séquence de données, tels que $r^i(x^i)$ et $r^j(x^j)$ sont renseignés et r^i précède r^j, un *enregistrement de variation* $\Delta r_{r^i r^j}$ est créé dans Δdb et contient l'item d'évolution $[x, TrendGran]$, où *TrendGran* est le sous-ensemble flou caractérisant la variation.

Par ailleurs, l'ordre de la base de données quantitative est conservé pour chaque

séquence de données : les enregistrements de variation $\Delta r_{r^i r^j}$ sont ordonnés chronologiquement en fonction des estampilles temporelles de r^i et r^j.

Exemple 4 *A partir des enregistrements r^1 et r^2 de la Table 1, un enregistrement de variation $\Delta r_{r^1 r^2}$ est créé, contenant l'item d'évolution $[x, decreasing]$ car la quantité pour x dans l'enregistrement r^1 est plus grande que celle enregistrée pour r^2.*

Afin de définir la tendance existant entre deux enregistrements de la base de données quantitative, les enregistrements r^i et r^j sont assimilés à deux points. Nous estimons alors la pente de la droite reliant r^i, de coordonnées ($r^i[x^i]$, d_i), à r^j, de coordonnées ($r^j[x^j]$, d_j). Cette pente nous permet de choisir les termes linguistiques représentant la tendance parmi les granules représentées sur la Figure 2(a). Enfin, le degré d'appartenance à ces tendances est obtenu grâce à des sous-ensembles flous prédéfinis, ceux par exemple décrits par la Figure 2(b). (Batyrshin, 2002) décrit plusieurs façons de caractériser des tendances grâce aux sous-ensembles flous.



(a) (b) (c)

Figure 2. *2(a) : Granules de tendances ; 2(b) : Sous-ensembles flous de tendances ; 2(c) : Termes linguistiques pour les durées.*

Exemple 5 *A partir des enregistrements r^1 et r^2 de la Table 1, l'item d'évolution créé dans l'Exemple 4 est en fait associé à la tendance* slowly decreasing*, étant donné que la pente entre r^1 et r^2 pour l'attribut x est égale à $\frac{3-4}{7-4} = -1/3$. De plus, la Figure 2(b) nous indique que cette pente correspond à un degré d'appartenance de 0.8 pour la tendance "slowly decreasing" et de 0.2 pour la tendance "constant".*
A partir des enregistrements r^3 et r^4 de la Table 1, l'item d'évolution créé pour l'attribut y correspond à une pente $\frac{6-4}{10-8} = 1$. L'enregistrement de variation correspondant créé dans Δdb est alors $\Delta r_{r^3 r^4}([x, increasing]) = 1$.

Il s'agit maintenant d'intégrer aux enregistrements de variation l'information de durée qui peut être obtenue à partir des estampilles temporelles des enregistrements originaux. Un ou deux items flous additionnels, *items de durée*, sont donc créés lors de l'ajout de l'enregistrement de variations $\Delta r_{r^i r^j}$, décrivant la durée écoulée entre les enregistrements initiaux r^i et r^j.

Exemple 6 *En considérant les sous-ensembles flous de durée Figure 2(c), la base de données de tendances issue de la Table 1 est celle décrite par le tableau 2. On*

			x			**y**				**z**		duration		
	r^i	r^j	dec	s.dec	cst	dec	cst	s.inc	inc	inc	q.inc	sh.	m.	lg.
δ^1	d1	d2		0.8	0.2							0.25	0.75	
δ^2	d1	d3	0.15	0.85									0.25	0.75
δ^3	d2	d3	1			1				0.3	0.7	1		
δ^4	d2	d4					0.2	0.8				0.25	0.75	
δ^5	d3	d4							1				0.75	0.25

Tableau 2. *Séquence de tendances issue de la Table 1.*

peut lire par exemple qu'entre les dates d1 et d2, le nombre d'accès à l'URL x a lentement diminué, pouvant être considéré comme presque constante, et que cette variation s'est déroulée sur une période moyenne. Alors que sur une période plus longue, allant de d1 à d3, le nombre de connections à l'URL x diminue plus fortement, la tendance étant caractérisée par "slowly decreasing" *et* "decreasing" *au lieu de* "constant" *précédemment.*

La base de tendances est donc constituée d'un ensemble d'enregistrement $\Delta r_{r^i r^j}$, contenant chacun : un id-objet o, correspondant à l'identifiant o de r^i et r^j, deux estampilles temporelles $t(r^i)$ et $t(r^j)$, respectivement estampilles temporelles de r^i et r^j, un ensemble d'items d'évolution $[x, v]$, et de durée $\Delta r_{r^i r^j}(\delta) = \mu_d(t(r^j) - t(r^i))$.

3.4. *L'algorithme* TED

Le processus décrit dans les paragraphes précédents (étape 1 de la figure 1) est réalisé par l'algorithme TED, décrit par la figure 3.

Cet algorithme parcourt la base de données quantitative et pour chaque enregistrement r, la suite de la séquence de données est parsée afin de trouver les enregistrements suivants qui contiennent des attributs renseignés pour r. Pour chaque paire d'enregistrements ayant des attributs en commun, un enregistrement est ajouté dans la base de tendances, contenant les items d'évolution et de durée correspondant, ainsi que les degrés d'appartenance.

L'algorithme TED crée la base de tendances (TrEnd Database) avec une complexité temporelle d'ordre $O(n^2)$, où n est le nombre d'enregistrements de la base initiale Qdb. Dans le pire cas, la base de tendances contiendra $\sum_{o \in \mathcal{O}} \frac{|\mathcal{R}_o|(|\mathcal{R}_o| - 1)}{2}$ enregistrements.

TED ***Main*** **- Input :** Qdb ; **Ouput :** Δbd
Δdb.*initialize*() ;
For each data sequence $o \in Q$db **do**
  For each record $r \in \mathcal{R}_o$ **do**
    For each record $r' \in \mathcal{R}_o$ / $t(r') > t(r)$ **do**
      Δr.*initialize*() ;
      For each attribute a **do**
        If (($r[a]$!= NULL) **AND** ($r'[a]$!= NULL)) **Then**
          Δr.*add*(a, v, $\mu_v(r[a] - r'[a])$) ; *[where v is the variation strength]*
        End If
      End For
      Δdb.*add*(o,Δr, $t(r)$, $t(r')$, d, $\mu_d(t(r') - t(r))$) ; *[with d, duration fuzzy set]*
    End For
  End For
End For
return Δdb ;

Figure 3. *Algorithme* TED

4. EVA : un algorithme pour la découverte de motifs d'évolution

Nous avons choisi d'implémenter notre approche sur le principe des approches d'extraction de motifs par niveau. Ce type d'algorithmes utilise les séquences fréquentes de taille k afin de générer les séquences candidates (potentiellement fréquentes) de taille $k + 1$. Ensuite, la fréquence de ces séquences candidates est calculée, les non fréquentes étant finalement supprimées. La fouille de la base de tendances est réalisées sur le principe de l'extraction de motifs séquentiels flous conduite par TOTALLYFUZZY, décrite dans (Fiot *et al.*, 2007b).

Cependant, l'application directe de cet algorithme n'est pas possible. Dans cette section, nous détaillons tout d'abord en quoi le format spécifique de la base de tendances interdit l'utilisation des algorithmes existants, puis nous introduisons notre solution permettant de gérer la chronologie des enregistrements. Enfin, nous décrivons brièvement l'algorithme EVA conçu dans ce sens.

4.1. *A propos des durées*

Puisque chaque enregistrement de variation $\Delta r_{r^i r^j}$ est construit à partir de deux enregistrements du jeu de données initial, il comporte deux estampilles temporelles $t(r^i)$ et $t(r^j)$, qui décrivent une durée. La recherche de séquences nécessite donc d'établir un ordre et/ou des contraintes entre les enregistrements de la base de tendances, tenant compte des estampilles initiales $t(r^i)$ et $t(r^j)$.

Considérons plusieurs enregistrements de variations qui auraient la même date de début $t(r^1)$, il ne peuvent être inclus dans une même séquence d'évolution, même

s'ils correspondent au même objet et à la même séquence de données. En effet, un enregistrement couvrant la période de $t(r^1)$ à $t(r^3)$ ne précède ni ne suit un second enregistrement contenant r^2 ayant eu lieu entre r^1 et r^3 dans la base originale Qdb (Allen, 1990). L'exemple 7 illustre ces cas grâce aux données de la Table 2.

Exemple 7 *La Figure 4(a) représente les enregistrements de variations de la Table 2. Le second enregistrement recouvre le premier et le troisième, et le quatrième recouvre le troisième et le cinquième enregistrement.*

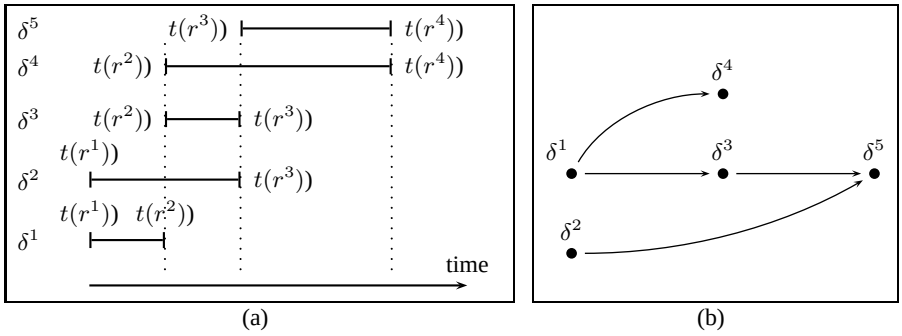


Figure 4. *4(a) : Enregistrements couvrants ; 4(b) : Graphe de séquence de la séquence de données de la Table 2.*

Afin de prendre en compte ces recouvrements possibles, la base Δdb devrait être scannée en réalisant de nombreux aller-retour lors de la recherche des séquences candidates. Pour éviter un traitement aussi coûteux, nous avons donc conçu une méthode qui permet de sauter les enregistrements couvrant pour chaque séquence candidate.

4.2. *Gestion de la chronologie des enregistrements*

Notre approche utilise une structure de graphe afin de représenter chaque séquence de données de Δdb. Les principes de ce modèle sont proches de ceux développés dans (Masseglia *et al.*, 2004) pour inclure des contraintes de temps lors de l'extraction de motifs séquentiels. Les sommets du graphe de séquence sont en fait les enregistrements de variations et les arcs représentent les séquences possibles.

Ainsi, avant l'extraction des motifs d'évolution, EVA transforme chaque séquence de données de la base de tendances en un graphe de séquences. Ce sont ces graphes de séquences qui sont parcourus pour découvrir les motifs d'évolution.

Pour chaque objet o de Δdb, le graphe de séquence est construit par la fonction *createGraph*, décrite dans (Fiot *et al.*, 2008). Cette fonction parcourt les séquences de données Δdb. Tout d'abord, un sommet est créé pour chaque enregistrement d'une

séquence de données. Puis, les arcs sont créés. Pour chaque sommet, *createGraph* créé les arcs des séquences autorisées, i.e. pour deux sommets v_i et v_j, un arc est construit de v_i vers v_j si et seulement si v_j.startDate() $\geq v_i$.endDate().

La Figure 4(b) représente le graphe de séquence obtenu à partir de la séquence de données donnée par le tableau 2. A partir de cette séquence de données, on peut construire trois séquences maximales dans lesquels rechercher des séquences candidates : $\langle \delta^1 \delta^3 \delta^5 \rangle$, $\langle \delta^2 \delta^5 \rangle$ et $\langle \delta^1 \delta^4 \rangle$.

4.3. *L'algorithme* EVA

L'algorithme général permettant l'extraction des motifs d'évolution (EVolution pAtterns) est décrit par la Figure 5.

EVA ***Main*** **- Input :** $minFreq$, Δdb ; **Ouput :** F, frequent evolution sequences
$F_0 \leftarrow \varnothing$; $k \leftarrow 1$; $F_1 \leftarrow \{\{< i >\}/i \in \mathcal{I} \& freq(i) > minFreq\}$;
For each trend data sequence $\delta S \in \Delta$db **do**
  $graphDB \leftarrow$ createGraph(δS) ;
End For
While ($Candidate(k) \neq \varnothing$) **do**
  For each sequence graph $g \in graphDB$ **do**
    [countFrequency is the TotallyFuzzy algorithm adapted to sequence graph parsing]
    countFrequency($Candidate(k), minFreq, g$) ;
  End For
  $F_k \leftarrow \{s \in Candidate(k)/freq(s) > minFreq\}$;
  $Candidate(k+1) \leftarrow$ generate(F_k) ; k++ ;
End While
return $F \leftarrow \bigcup_{j=0}^{k} F_k$

Figure 5. EVA *: algorithme principal*

Une fois la base de tendances générées par TED, la fonction *createGraph* est appelée afin de transformer chaque séquence de données en un graphe de séquences, intégrant ainsi les contraintes temporelles liées à la gestion des durées de variation.

Ensuite, les graphes de séquences sont parcourus afin de découvrir les items d'évolution fréquents, en fonction du seuil de fréquence minimale $minFreq$ spécifié par l'utilisateur. Après cette étape, les fréquents de taille k sont combinés en séquences candidates de taille $k+1$. Ces séquences sont ensuite recherchées dans les graphes de séquences et EVA détermine leur fréquence en utilisant les principes de l'algorithme TOTALLYFUZZY (Fiot *et al.*, 2007b), implémentant le calcul de fréquence 1, section 2. L'algorithme EVA s'arrête lorsqu'il n'y a plus de séquences candidates trouvées fréquentes.

La complétude de ces approches par graphe de séquences a été démontrée (Fiot, 2007). A la fin du processus, nous obtenons donc bien tous les motifs d'évolution supportés par la base de tendances.

5. Expérimentations

Nous avons appliqué l'extraction de motifs d'évolution à l'analyse d'utilisation d'un site Web et plus particulièrement afin d'identifier les pages visitées de façon récurrentes et répétées, ainsi que l'évolution du nombre d'accès à chaque page.

5.1. *Données*

Les logs d'accès d'un site Web de laboratoire ont été nettoyés et préparés. Les enregistrements contiennent le nombre d'accès à une page, la même demi-journée par un utilisateur. Par exemple, l'enregistrement *"1500 5067 10 6"* signifie que *"le visiteur 1500"* a visité *6* fois l'URL codé par *10* lors de la demi-journée *5067*. Cette base de données contient 27209 pages Web visitées par 79756 IPs différentes durant 3 mois (91 demi-journées). La conversion des données au formalisme d'extraction détaillé dans les sections 2 et 3 est donnée dans le tableau 3.

Objet	$\leftrightarrow$	IP
Date	$\leftrightarrow$	demi-journée
Items quantitatifs	$\leftrightarrow$	nombre d'accès à chaque page
Items d'évolution	$\leftrightarrow$	variation du nombre d'accès à chaque page web
Durée	$\leftrightarrow$	période séparant deux accès à une même page

Tableau 3. *Format pour l'extraction de motifs dans des logs d'accès Web*

Comme détaillé dans la section 3.1, les quantités sont comparées par notre algorithme TED, qui convertit le jeu de données en une base de données de tendance contenant des items d'évolution et les degrés d'appartenance aux sous-ensembles flous de variations. Ensuite ces données sont fouillées grâce à l'algorithme EVA.

5.2. *Résultats*

En ce qui concerne les performances d'exécution, le comportement global de notre algorithme d'extraction de motifs d'évolution est similaire à celui observé lors d'extraction de motifs séquentiels flous : à mesure que la fréquence minimale diminue, le nombre et la taille des séquences fréquentes augmentent, ainsi que le nombre de candidats à tester, impliquant une augmentation du nombre de passes sur la base de données et donc du temps d'exécution.

Quant à l'analyse qualitative, les motifs découverts semblent pertinents. Un motif d'évolution que nous avons découvert dans des logs d'accès au site de l'INRIA Sophia

concerne le projet de Koala et exprime la tendance temporelle suivante *"L'augmentation lente du nombre de connexions à la page KBM précède d'une courte période, l'augmentation, pendant une longue période, des connexions à la page de KOML. Ces augmentations sont suivies d'une augmentation lente des connexions à DJAVA"*.

Ces résultats présentent l'avantage d'être expressifs, lisibles, plus informatifs que les motifs séquentiels classiques et complémentaires des motifs séquentiels flous extraits sur cette même base.

6. Conclusion

Dans cet article, nous avons présenté une approche permettant la découverte d'évolutions et de durées typiques séparant les événements de bases de données de séquences numériques. Cette approche est basée sur les motifs séquentiels flous, utilisés pour la fouille de séquences de variations.

Contrairement à l'analyse de séries temporelles, dans le contexte des séquences de données, certains attributs peuvent ne pas être remplis sur certaines périodes ou par certains enregistrements. De plus la durée entre deux enregistrements consécutifs n'est pas nécessairement régulière. C'est pourquoi, la description de tendances dans de telles données numériques requiert une approche spécifique.

Nous avons donc proposé un processus basé sur deux algorithmes. Tout d'abord, TED convertit une base de données de séquences quantitatives en une base de données de tendances, décrivant l'évolution de valeurs d'attribut numériques pour plusieurs objets. Ces évolutions sont représentées sous la forme de séquences de tendances. Ensuite, EVA extrait les séquences d'évolutions fréquentes dans cet ensemble de données. Les motifs d'évolution ainsi découverts dans des logs d'accès Web informent par exemple qu'*un nombre croissant de demandes à registration.php pendant une période courte précède un nombre croissant de demandes à faq.html, après une période très courte*. Ces relations temporelles dans des navigations Web pourraient alors être utilisées pour modifier l'architecture du site selon l'utilisation qui en est faite, ainsi que la qualité de services, en facilitant l'accès à certaines pages fréquemment visionner ou en évitant d'archiver des articles accédés de façon répétées sur de longues périodes.

Les extensions de ce travail pourraient mener à la définition d'implications temporelles, décrivant des relations causales entre l'évolution d'attributs. Celles-ci incluraient alors quelques calculs statistiques dont la recherche de dépendances basées sur des régressions linéaires. La connaissance ainsi découverte pourrait alors être utilisée pour l'explication de certains évènements telles que des défaillances.

7. Bibliographie

Agrawal R., Srikant R., « Mining sequential patterns », *11th Int. Conf. on Data Engineering*, p. 3-14, 1995.

Allen J. F., « Maintaining knowledge about temporal intervals », *Readings in qualitative reasoning about physical systems*p. 361-372, 1990.

Batyrshin I., « On granular derivatives and the solution of a granular initial value problem », , vol. 12, n° 3, p. 403-410, 2002.

Chen R.-S., Tzeng G.-H., Chen C.-C., Hu Y.-C., « Discovery of Fuzzy Sequential Patterns for Fuzzy Partitions in Quantitative Attributes », *ACS/IEEE Int. Conf. on Computer Systems and Applications*, p. 144-150, 2001.

Dong G., Li J., « Efficient Mining of Emerging Patterns : Discovering Trends and Differences », *5th Int. Conf. on Knowledge Discovery and Data Mining (KDD'99)*, p. 43-52, 1999.

Fiot C., « Extension des contraintes de temps : précision et flexibilité pour les motifs séquentiels généralisés », *Revue I3*, 2007.

Fiot C., Laurent A., Teisseire M., « Approximate Sequential Patterns for Incomplete Sequence Database Mining », *16th IEEE Int. Conf. on Fuzzy Systems (FuzzIEEE'07)*, 2007a.

Fiot C., Laurent A., Teisseire M., « From Crispness to Fuzziness : Three Algorithms for Soft Sequential Pattern Mining », *IEEE Transactions on Fuzzy Systems*, vol. 15, n° 6, p. 1263-1277, 2007b.

Fiot C., Masseglia F., Laurent A., Teisseire M., Ted and Eva : Expressing Temporal Tendencies Among Quantitative Variables Using Fuzzy Sequential Patterns, Technical Report n° RR-08002, LIRMM, 2008.

Hong T., Lin K., Wang S., « Mining Fuzzy Sequential Patterns from Multiple-Items Transactions », *Joint 9th IFSA World Congress and 20th NAFIPS Int. Conf.*, p. 1317-1321, 2001.

Hugueney B., Bouchon-Meunier B., « Time-Series Segmentation and Symbolic Representation, from Process-Monitoring to Data-Mining », *Computational Intelligence. Theory and Applications : Int. Conf., 7th Fuzzy Days*, p. 1611-3349, 2001.

Kacprzyk J., Wilbik A., Zadrozny S., « Capturing the Essence of a Dynamic Behavior of Sequences of Numerical Data Using Elements of a Quasi-natural Language », *IEEE Int. Conf. on Systems, Man and Cybernetics (SMC'06)*, p. 3365-3370, 2006.

Kacprzyk J., Wilbik A., Zadrozny S., « Linguistic Summaries of Time Series via an OWA Operator Based Aggregation of Partial Trends », *IEEE Int. Conf. on Fuzzy Systems (FuzzIEEE'07)*, p. 1-6, 2007.

Keogh E. J., Pazzani M. J., « Scaling up Dynamic Time Warping to Massive Datasets », *Principles of Data Mining and Knowledge Discovery*, p. 1-11, 1999.

Keogh E., Lin J., Fu A., « HOT SAX : Efficiently Finding the Most Unusual Time Series Subsequence », *5th IEEE Int. Conf. on Data Mining (ICDM '05)*, p. 226-233, 2005.

Lent B., Agrawal R., Srikant R., « Discovering Trends in Text Databases », *3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97)*, p. 227-230, 1997.

Masseglia F., Poncelet P., Teisseire M., « Pre-Processing Time Constraints for Efficiently Mining Generalized Sequential Patterns », *11th Int. Symposium on Temporal Representation and Reasoning*, p. 87-95, 2004.

Sklansky J., Gonzalez V., « Fast polygonal approximation of digitized curves », *Pattern Recognition*, vol. 12, n° 5, p. 327-331, 1980.

Srikant R., Agrawal R., « Mining Sequential Patterns : Generalizations and Performance Improvements », *5th Int. Conf. on Extending Database Technology*, p. 3-17, 1996.

Zadeh L., « Fuzzy Sets », *Information and Control*, vol. 3, n° 8, p. 338-353, 1965.