Using event sequence alignment to automatically segment web users for prediction and recommendation

Vinh Trung Luu

École Doctorale 269 Mathématiques, Sciences de l'Information et de l'Ingénieur (MSII)
Laboratoire Modélisation Intelligence Processus et Systèmes (MIPS)

Thèse présentée pour obtenir le grade de

Docteur de l'Université de Haute Alsace

Discipline : Informatique

Using event sequence alignment to automatically segment web users for prediction and recommendation

---

Par : Vinh-Trung Luu

Soutenue publiquement le 12/16/2016

Membres du jury :

Rapporteur : Mustapha Lebbah, Maître de Conférences HDR, Université Paris 13
Rapporteur : Fabrice Bouquet, Professeur, Université de Franche-Comté
Examinateur : Abderrafiaa Koukam, Professeur, Université de Technologie de Belfort-Montbéliard
Directeur de thèse : Pierre-Alain Muller, Professeur, Université de Haute Alsace
Examinateur : Germain Forestier, Maître de Conférences, Université de Haute Alsace
Examinateur : Frédéric Fondement, Maître de Conférences, Université de Haute Alsace

Contents

List of Figures

List of Tables

Acknowledgment

As a representation of lessons learnt in Using event sequences alignment for automatic web users segmentation, this thesis represents a milestone after 3 years of work at Univesite de Haute Alsace and particularly at the MIPS-ENSISA, from December 2013 to December 2016.

I would like to express my sincere appreciation to my thesis director, Professor Dr. Pierre-Alain Muller, Vice-President of Innovation of Université de Haute Alsace for your patience, motivation and constant support of my research. You have been encouraging me and guiding me to grow as a researcher as well as finish this thesis.

I would also thank my enthusiastic advisors very much, Dr. Germain Forestier and Dr. Frederic Fondement. You have helped me to build up the research in depth and your advise on both my Ph.D study and career path have been valuable.

Besides, I would also like to thank professor Mustapha Lebbah, professor Fabrice Bouquet, professor Abderrafiaa Koukam for being my committee members and for their perceptive comments and consolidation. My sincere thanks also go to my friends Mathis Ripken, Florent Bourgeois, Mariem Mahfoudh, Houda Chanti and Paul Bourgeois for all of your support during my Ph.D study.

I am deeply grateful to be funded by Vietnam International Education Development and Campus France, and assisted by BeamPulse, thanks to you.

After all, I dedicate this thesis to my family for spiritually supporting me all over thesis writing, and being with me all the time.

Chapter 1

Résumé en Français

Introduction

Une masse de données importante est collectée chaque jour par les gestionnaires de site internet sur les visiteurs qui accèdent à leurs services. La collecte de ces données a pour objectif de mieux comprendre les usages et d'acquérir des connaissances sur le comportement des visiteurs. A partir de ces connaissances, les gestionnaires de site peuvent décider de modifier leur site ou proposer aux visiteurs du contenu personnalisé. Cependant, le volume de données collectés ainsi que la complexité de représentation des interactions entre le visiteur et le site internet nécessitent le développement de nouveaux outils de fouille de données. Dans cette thèse, nous avons exploré l'utilisation des méthodes d'alignement de séquences pour l'extraction de connaissances sur l'utilisation de site Web (web mining). Ces méthodes sont la base du regroupement automatique d'internautes en segments, ce qui permet de découvrir des groupes de comportements similaires. De plus, nous avons également étudié comment ces groupes pouvaient servir à effectuer de la prédiction et la recommandation de pages. Ces thèmes sont particulièrement importants avec le développement très rapide du commerce en ligne qui produit un grand volume de données (big data) qu'il est impossible de traiter manuellement. L'utilisation de l'alignement de séquences dans ce domaine a cependant été encore peu étudié. Nous proposons ainsi dans cette thèse d'étudier l'utilisation de traces de navigation afin de mieux comprendre et de prédire le comportement des internautes lors de leur navigation sur des sites internet. Notre objectif principal est la construction automatique de segments qui regroupent de nombreux internautes ayant un comportement similaire. Ces segments peuvent par la suite être utilisés afin de mener des campagne de marketing ciblé. Ces travaux

ont été réalisé en collaboration avec la société Beampulse qui a été notre fournisseur de données.

Contexte et motivation

Nous travaillons sur le marketing comportemental sur internet. D'une part, nous observons le comportement des visiteurs, et d'autre part, nous déclenchons (en temps-réel) des stimulations destinées à modifier ce comportement. Le fonctionnement en temps-réel et la personnalisation de masse sont les deux défis que nous devons relever. L'analyse des usages sur internet a été largement utilisé pour transformer les données de navigation bas-niveau (tels que click sur les pages) en connaissances exploitables par les gestionnaires sites. Une session contient toutes les interactions (click, changement de pages, etc.) qu'un utilisateur a effectué avec un site lors d'une visite. Afin de pouvoir détecter des comportements similaires dans un ensemble de sessions, il est nécessaire de pouvoir évaluer la similarité entre deux sessions. La granularité de ces événements peut être affinée, de pages chargées jusqu'au niveau des événements Javascript. Dans cette thèse, nous considérons les sessions comme des séquences d'événements, et nous nous intéressons plus particulièrement aux séquences de pages visités par l'internaute lors d'une visite. Comme cadre applicatif, nous avons un accès quotidien à des centaines de milliers de ces séquences, qui sont enregistrées par notre partenaire industriel BeamPulse. Ces séquences proviennent principalement de sites de commerce électronique. La Figure 1.1 résume l'approche adoptée dans nos travaux.

Afin de mesurer la similarité entre des séquences d'événements, de nombreuses méthodes d'alignement de séquences ont été envisagées et appliquées afin de trouver l'approche appropriée pour l'exploitation des comportements des internautes. Nous avons proposé un état de l'art des mesures existantes afin de présenter les avantages et les inconvénients de chacune des approches. Nous nous sommes intéressés plus particulièrement à deux méthodes d'alignements issues des techniques d'alignement utilisées en bio-informatique. L'alignement global, qui aligne l'intégralité de deux séquences avec l'algorithme de Needleman-Wunsch; et l'alignement local, qui aligne une partie de deux séquences, avec l'algorithme de Smith-Waterman.

Dans nos travaux, nous avons proposé une nouvelle méthode permettant de com-

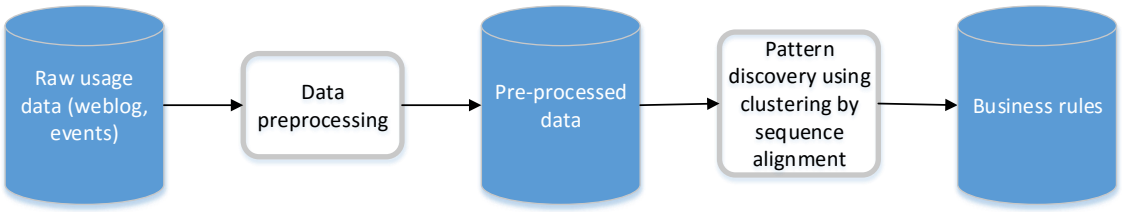


Figure 1.1: La vue d'ensemble des différentes étapes de l'acquisition, à leur traitement puis à leur exploitation par les gestionnaires de sites internet.

biner ces deux algorithmes. Cette combinaison de méthodes permet de prendre en compte la similarité locale et globale des séquences, menant à une meilleure évaluation de leur similarité.

Un problème qui a nous particulièrement mobilisé, est la comparaison de séquences ayant des différences importantes en termes de longueur. En effet, l'évaluation de la similarité de ces séquences est difficile, et les méthodes classiques échouent souvent à l'évaluer correctement. Afin de résoudre de problème, nous avons proposé de combiner similarité locale et similarité globale en pondérant relativement ces deux mesures. Ainsi, peut importe la différence en longueur des séquences, la similarité est correctement évaluée.

La méthode de Needleman-Wunsch est basée sur la programmation dynamique pour l'alignement de deux séquences. La programmation dynamique rends possible l'alignement optimal de deux séquences et est très populaire dans le domaine de l'alignement. Dans l'algorithme de Needleman-Wunsch, l'alignement est effectué du début à la fin de chaque séquence, ce qui produit un alignement global. L'alignement global de séquences fonctionne bien avec des séquences de même longueur afin de trouver le meilleur alignment possible. Cependant, l'alignement global donne souvent de mauvais résultats avec des séquences de longueurs différentes, par exemple si l'une d'elle contient plusieurs sous séquences similaires. Dans ce cas, un alignement local est préférable. L'algorithme de Smith-Waterman, permet lui, de trouver un alignement local en recherchant les régions similaires dans les deux séquences.

Ces deux méthodes de programmation dynamique sont très souvent utilisées pour aligner des protéines ou des séquences de nucléotides. Comme les algorithmes de Needleman-Wunsch et Smith-Waterman ont des objectifs différents, chacune de ces méthodes a des avantages et des inconvénients. Needleman-Wunsch trouve l'alignement

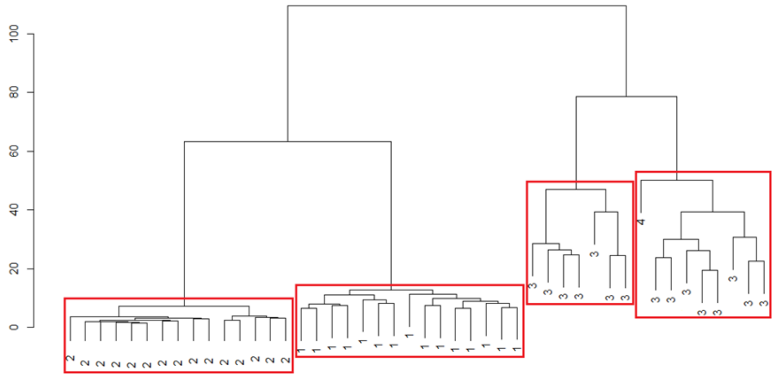


Figure 1.2: Exemple de clustering hiérarchique ascendant obtenu avec la mesure combinant les algirthmes de Needleman-Wunsch et de Smith-Waterman.

optimal des séquences entières alors que Smith-Waterman détecte des régions de similarité entre les deux séquences. Afin de lever ce verrou, nous avons proposé d'utiliser une combinaison de ces deux méthodes permettant ainsi d'évaluer au mieux la similarité entre les séquences (localement et globalement) :

$$S(s_i, s_j) = \left[\frac{NW(s_i, s_j)}{l}\right] + \left[\frac{SW(s_i, s_j)}{(2 * l)}\right] \tag{1.1}$$

Avec $NW(s_i, s_j)$ et $SW(s_i, s_j)$ les scores respectifs des algorithmes de Needleman-Wunsch et de Smith-Waterman entre deux séquence s_i et s_j, et l la distance de la plus longue séquence entre s_i et s_j. L'avantage de cette combinaison est que celle-ci fonctionne mieux quand la différence de longueur entre les deux séquences est importante. En effet, plus la différences de longueur est importante, plus notre métrique prendra en compte le score de Smith-Waterman, et se concentrera ainsi à trouver un alignment local pertinent.

Une fois cette mesure de similarité définie, celle-ci a été utilisée afin de construire des segments de visites d'internaute similaires. Nous avons utilisé un algorithme de clustering hiérarchique ascendant avec le critère de Ward pour sa capacité à construire des groupes équilibrés. La Figure1.2 présente un exemple de clustering hiérarchique obtenu.

Les travaux sur la définition d'une métrique combinant Needleman-Wunsch Smith-Waterman ont donné lieu à deux publications scientifiques dans un atelier d'une conférence internationale (*Luu et al.*, 2015a) et dans une conférence internationale (*Luu*

et al., 2016a). Dans *Luu et al.* (2015a) nous présentons nos premiers travaux avec des techniques basées sur le filtrage de la similarité. Nous avons défini des règles permettant de prendre en compte les scores produits par les algorithmes de Needleman-Wunsch et de Smith-Waterman. Dans la suite de ces travaux (*Luu et al.*, 2016a), nous avons proposé une nouvelle métrique combinant ces deux scores afin d'éviter à l'utilisateur d'avoir à définir des valeurs de seuils.

Dans la continuité de ces travaux, nous nous sommes intéressés à l'utilisation des résultats précédents pour la prédiction et la recommandation lors de la visite d'internautes. L'idée principale est de réussir à prédire lors d'une visite, quelles sont les prochaines pages possibles que l'utilisateur va visiter, afin d'émettre des recommandations. Ceci permet notamment de guider l'internaute dans sa visite sur un site internet. Dans ce cadre, nous avons proposé une méthode permettant de construire de façon automatique des groupes de comportement. Nous avons modifié la méthodologie précédente (*Luu et al.*, 2015a, 2016a) afin de s'assurer que chaque groupe ne contienne que des séquences ayant un préfixe similaire. Ainsi, lors d'une nouvelle visite, le groupe de comportement le plus proche de la visite en cours est trouvé. Ce groupe est ensuite utilisé afin d'émettre une prédiction sur la prochaine page que l'utilisateur est susceptible de visiter. Avec cette information, le gestionnaire de site peut décider de déclencher des actions particulières afin d'effectuer une recommandation ou de guider l'internaute vers une page spécifique. Ces travaux ont été validés dans une conférence internationale (*Luu et al.*, 2016b).

Les travaux réalisés dans cette thèse on ouvert de nombreuses perspectives de recherche. En effet, nous nous sommes limités jusqu'à présent à l'utilisation de séquences représentant la suite des pages visitées par un internaute lors d'une session. Cependant, de nombreux autres évènements pourraient être pris en compte comme les clics, les scrolls, etc. Enfin, les choix méthodologiques actuels ne permettent pas un passage à l'échelle immédiat, les algorithmes d'alignement étant encore trop coûteux en temps de calcul. Des adaptations sont actuellement à l'étude afin de pouvoir passer du traitement de plusieurs milliers de séquences à plusieurs dizaines de milliers voir des millions.

Chapter 2

Introduction

Thesis abstract

This thesis explored the application of sequence alignment in web usage mining, including user clustering and web prediction and recommendation. This topic was chosen as the online business has rapidly developed and gathered a huge volume of information and the use of sequence alignment in the field is still limited. In this context, researchers are required to build up models that rely on sequence alignment methods and to empirically assess their relevance in user behavioral mining. This thesis presents a novel methodological point of view in the area and show applicable approaches in our quest to improve previous related work.

Context and motivations

Web usage behavior analysis has been central in a large number of investigations in order to maintain the relation between users and web services. Useful information extraction has been addressed by web content providers to understand users' need, so that their content can be correspondingly adapted. One of the promising approaches to reach this target is pattern discovery using clustering, which groups users who show similar behavioral characteristics. Our research goal is to perform users clustering, in real time, based on their session similarity. Since web sessions can be regarded as sequences of symbol, we propose a similarity evaluation method which relies on sequence alignment techniques. The mechanism of this approach is described in Figure 2.1. Our main contribution is the combination of global and local sequence alignment techniques such as Needleman-Wunsch and Smith-Waterman to evaluate the

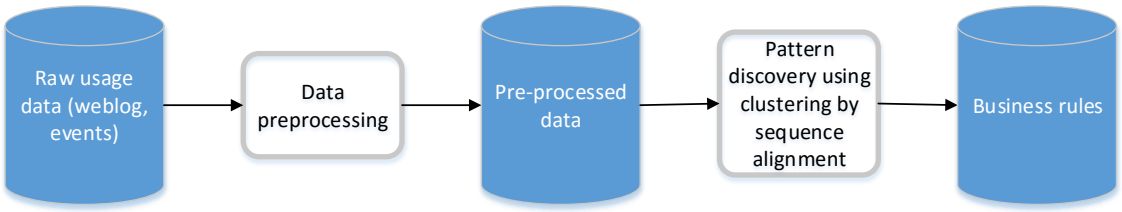


Figure 2.1: The overview of web usage mining that applies clustering based on sequence alignment similarity.

similarity between web sessions.

In order to enhance previous efforts to apply sequence alignment in session similarity evaluation, we targeted their disadvantages to propose a novel sequence similarity measure which meets the specificities of web sessions, such as the variable of length. In addition, a time-series sequence alignment technique like Dynamic Time Warping was also taken into consideration as competitor to prove its improperness in aligning user sessions. Furthermore, we used clustering algorithms based on our proposed measure to build up personalized web application which is applicable in marketing strategies. Furthermore, the result of web user behavioral clustering was used to draw prediction, and subsequently we show that these predictions provide useful knowledge for web user recommendation.

Outline

This thesis is organized as follows: Chapter 2 reviews current approaches referring to research area. Correspondingly, the state-of-the-art is categorized into five classes which are related to sequence length difference, element order and succession. Chapter 3 presents our contributions that define web session alignment application, describe the combination of local and global alignment for computing similarity of sessions, examine the performance of our approach and competitor measures through clustering experiments. In addition, this chapter provides web prediction method which takes advantage of our clustering, and web recommendation based on prediction, that are valuable in behavioral marketing. Chapter 4 concludes the thesis by giving a short statement of the main results, our contribution summary and the perspective of research.

Publications:

- Luu, V.T., Forestier, G., Fondement, F. and Muller, P.A., 2015. Web site audience segmentation using hybrid alignment techniques. In Trends and Applications in Knowledge Discovery and Data Mining (pp. 29-40). Springer International Publishing. (*Luu et al.*, 2015a)

- Luu, V.T., Ripken, M., Forestier, G., Fondement, F. and Muller, P.A., 2016. Using Glocal Event Alignment for Comparing Sequences of Significantly Different Lengths. In Machine Learning and Data Mining in Pattern Recognition (pp. 58-72). Springer International Publishing. (*Luu et al.*, 2016a)

- Luu, V.T., Forestier, G., Ripken, M., Fondement, F. and Muller, P.A. Web usage prediction and recommendation using web session clustering, The Eleventh International Conference on Digital Information Management (IEEE - 2016) (*Luu et al.*, 2016b)

- Luu, V.T., Forestier, G., Ripken, M., Fondement, F. and Muller, P.A. A review of web session similarity measures for web usage mining (**Submitted** to Information Sciences, Elsevier)

Chapter 3

State of the art

Introduction

In recent years, web usage mining has received a growing interest motivated by the unprecedented increase of web traffic. The goal of web usage mining is to identify hidden pattern from visitor browsing data. The aim is to adapt website features to visitor interest, create recommender systems, personalize information, etc. This involves clustering of different visits having similar navigational patterns. One of the most popular approaches to discover these clusters is web session clustering. In this context, new techniques have been proposed to track web user navigation and record their interaction with web sites (*e.g.* sequence of page visits, clicks, etc.) by event listener or server log file. Visitor segmentation consists in using these interactions to create groups of users having similar behaviors in the way they interact with the web site. Creating these groups is of major interest because they can later be targeted with specific actions in order to recommend them content. An efficient session grouping system not only relies upon the nature of the clustering algorithm but also on the performance of session similarity measure. Therefore, one of the most important challenges to create the groups, is the definition of similarity measures allowing to consistently compare web sessions. For a measure to be consistent, it has to provide a graduated evaluation of how similar two web sessions are. Only computing statistics on the number of visited pages or the mean duration of the visit is not satisfactory. Indeed, if the exact same activities were performed in a random order, the evaluation would have been the same. Consequently, a strong interest has been given to alignment techniques which allows to take into account the specificity of the data. Alignment is a well-known approach which write one sequence on top of the other

and insert necessary spaces to equalize their length, then align one-to-one opposite elements. This way of sequence comparison finds it utilization in various domains, and web session alignment which regards sessions as sequences, is among them. In different web applications, there has been a variety of measures respecting alignment-based sequence comparison, including similarity and dissimilarity ones, adopted for evaluating the similarity of web sessions. In the objective of similarity evaluation, the distance scores should be minimized to make the alignment more appropriate. On the contrary, dissimilarity measures seek the highest score which is obtained by mismatches or gaps to decide which alignment is the best. In other words, sessions are clustered according to their low intra-group similarity and separated by their high intra-group similarity. Concerning prior sequence alignment and primarily known as dissimilarity measures, edit-distance type measures compute the distance between sequences through the amount of insertions, deletions, substitutions and even transpositions to make them identical.

A variation of algorithms, particularly from bioinformatic, have been adopted or improved to deal with web session alignment. Despite the long investigation of web session alignment methods, recent researches have indicated considerable progress in scalability, correctness or robustness to noise. In this paper, we review the most important similarity measures that have been proposed for web session similarity evaluation as well as other applicable approaches and restrictions of measures. We propose five categories of existing measures: (1) Not taking sequences with different lengths into account, (2) Taking sequences with different lengths into account but not the order of element, (3) Taking sequences with different lengths and order of elements into account but not their succession, (4) Taking sequences with different lengths, order of elements and locally their succession into account, and (5) Taking sequences with different lengths, order of elements and their global and local matching into account. Each category is composed of a summary table and corresponding description which provides more details for each measure. After reviewing measure categories, we draw readers' attention to other approaches which are not sequence alignment based. We continue to discuss challenges (in reference to the limitation of sequence alignment application) which the measures have to encounter in web session similarity evaluation. The organization of the paper is as follows: in Section 2, we introduce the concept of sequence alignment and its corresponding formalization. In Section 3, we

present the most important methods and we categorize them into five main groups depending of their features. In Section 4, with present other related approaches not based on alignment. Finally, Section 5 presents a discussion about the measure and Section 6 concludes the paper.

Sequence, alignment and score

In this section, we introduce the notations that will be used later to present and compare sequence alignment approaches.

Sequence

Let Σ be a finite alphabet set that consists of symbols or characters. Let Σ^+ be the set of all strings over the alphabet Σ. Any possible string $s \in \Sigma^+$ formed by characters drawn from Σ is defined as $s[1 \dots n] = s[1], s[2] \dots s[n]$ where $s[i] \in \Sigma$ for $1 \leq i \leq n$, and $n = |s|$ denotes the length of s. Assume we are given $\Sigma = \{\mathsf{ABCDE}\}$, a set of k strings $S = \{s_1, s_2, \dots, s_k\}$ possibly contains $s_1 = \mathsf{AB}, s_2 = \mathsf{ABCD}, s_3 = \mathsf{BDCA}, \dots, s_k = \mathsf{ACDEB}$.

Web Access Sequences (WAS) are ordered sequences of pages accessed in a session. They can be extracted either from log files or recorded from low-level events, in an online or offline manner. If we consider each page-visit to be an alphabetical character, then each WAS can be treated as a string in the example above, representing the order in which visitors have accessed website pages. Accordingly, a sequence such as $s_i = \mathsf{ACDEB}$ shows that a visitor entered the website by page A, accessed C, D, E afterwards, and finally ended up with B. Consequently, S is a set of all WAS taken place in a website.

A *subsequence* is a common sequence-related notion, let s_i be a subsequence of s_j if ones can find s_i by removing none or some characters from s_j. For instance, $s_j = \mathsf{ADB}$ is a subsequence of $s_i = \mathsf{ACDEB}$, and s_i is said to be the *supersequence* of s_j.

Alignment

The sequence alignment over a set $S = \{s_1, s_2, \dots, s_k\}$ is another set $S_a = \{s_1^a, s_2^a, \dots, s_k^a\}$ of identical length sequences where each s_i^a can be built from s_i through gap "-" insertions for $1 \leq i \leq k$. S_a is called *aligned* sequence set (*Della Vedova*, 2000). Given

a set of two or more sequences $S = \{s_1, s_2, \ldots, s_k\}$, an alignment of such set can be defined as a matrix **A** of k-rows where the i-th row is sequence s_i^a, each cell holds an element of Σ or "-"and there is no column of the matrix which holds only gaps. Under an alignment **A** of two sequences s_1 and s_2, given n is the length of aligned sequences, elements $s_1^a[i]$ and $s_2^a[i]$ are opposite for $1 \leq i \leq n$. Figure 3.1 illustrates three among all possible alignments of two sequences.

ABCDC- ACDCEC	ABCDC-- A-CDCEC	——ABCDC ACDCEC-
(a) First alignment	(b) Second alignment	(c) Third alignment

Figure 3.1: Three among all possible alignments of two sequences.

If $s_1^a[i]$ is identical to $s_2^a[i]$, a *match* appears. Otherwise, if $s_1^a[i]$ and $s_2^a[i]$ are different and neither of them is a gap, it is a *mismatch*. Otherwise, a *gap* occurs. A *mismatch* can be considered as a substitution of $s_1^a[i]$ with $s_2^a[i]$ or vice versa. Alternatively, if $s_1^a[i]$ is "-"and $s_2^a[i]$ is not, a *gap* is a insertion of $s_2^a[i]$ in the position of $s_1^a[i]$ or a deletion of $s_2^a[i]$. These edit operations, also called *indels*, are applied to two sequences to convert one to another. It is pairwise alignment if there are such two sequences involved in the alignment. In case the number of sequences is higher than two, the alignment is called multiple alignment.

Score

A distance between sequences is defined as a reflection of the amount of work required to make sequences duplicate as mentioned previously. In general, this distance can be counted by a *score*. This score, high or low, indicates the cost of equalizing sequences. The cost for each substitution or indel can be defined as a function d: $(|\ \Sigma \cup \{-\}\ |) \times (|\ \Sigma \cup \{-\}\ |) \rightarrow \mathbb{R}$. For every $a, b \in \Sigma \cup \{-\}$,

$$d(\mathsf{A}, \mathsf{B}) = \begin{cases} 0, & \text{if } \mathsf{A} = \mathsf{B} \\ d(\mathsf{B}, \mathsf{A}), & \text{otherwise} \end{cases} \tag{3.1}$$

Depending on the defined scoring scheme, score of a substitution or indel may be various and have an effect on the *best scoring alignment* since the final alignment score is the sum of all *pair-score* through the sequences (*Bose and van der Aalst*, 2012).

For example, there are 1 match, 1 indel and 4 substitutions in the first alignment in Figure 3.1, the second one has 4 matches and 3 indels, the last one has 11 indels. If a scoring scheme of 0 for match,-1 for indel and -2 for substitution are applied, the final score of the first, second and last alignments would be $(1\times0)+(1\times-1)+(4\times-2) =$ **-9**, $(4\times0)+(3\times-1) =$ **-3** and $11\times-1 =$ **-11** respectively. However, if $0,-2$ and -1 are correspondingly assigned to match, indel and substitution scores, the first and second alignments in Figure 3.1 would have both the score of **-6**, and the last would have a score of **-22**. Alternatively, in case that both substitution and indel are scored 1 and match is scored 0, the maximum score value **11** is reached for the third alignment, while for the first one it would be **5**, and for the second alignment it would be **3**. Consequently, if the best alignment is evaluated based on score, a variant scoring scheme can importantly modify the outcome. Furthermore, the scoring scheme can affect clustering result, for example two compared sequences may be in the same cluster by the last scoring scheme and not be in the same using the first one.

It is also possible to set different weights for each edit operator on specific pair of symbols, namely the score of match, indel or substitution so that they are not the same in the whole alignment. For example, indels between a and "-" could be 1 but 2 between b and "-" and 3 between c and "-" . Besides, substitution score of a and b or b and c is 2, which is different from 1 of a and c. Additionally, match score of a and b could be different with 1 and 0, respectively. Applying this extended scoring scheme to the two sequence pairs in Figure 3.2, leads to the final alignment scores of $1+2+1+0+1+2 =$ **7** and $2+3+1+0+1+2 =$ **9** correspondingly. Likewise, these distinct results due to scoring scheme have a significant impact on alignment evaluation.

```
ACABA-        ACABA-
-BCBAB        B-CBAB
```

(a) First alignment (b) Second alignment

Figure 3.2: Example of sequence alignment with extended scoring scheme.

Similarity and dissimilarity

In a nutshell, alignment method is a well-known approach which writes one sequence on top of the other and insert necessary spaces to equalize their length, then align one-to-one opposite elements. In different web applications, there has been a variety of measures respecting alignment-based sequence comparison, including similarity and dissimilarity ones, adopted for evaluating the similarity of sessions. In the objective of similarity evaluation, the distance scores as presented in the previous section should be minimized to make the alignment more appropriate. On the contrary, dissimilarity measures seek the highest score which is obtained by mismatches or gaps to decide which alignment is the best. Concerning prior sequence alignment and primarily known as dissimilarity measures, edit-distance type measures compute the distance between sequences through the amount of insertions, deletions, substitutions and even transpositions to make them identical. Another interesting type of measures using sequence alignment is statistical methods, which show relationship between sequences by appearance frequency of common and uncommon elements. In the following section, we address some specific features that a similarity measure should meet to figure out hidden groups of visitors having similar browsing patterns. They have the ability to work with variable length sequences, taking element order into consideration, fully counting not only global but local matching of sequences.

Approaches

Among various clustering techniques, many previous studies have focused on sequence alignment algorithms to evaluate the similarity of sessions. Accordingly, pairwise alignment is commonly used to compare sequences optimally. Defining a similarity measure between web sessions is very important as it is generally the first step to apply clustering algorithms. Indeed, these algorithms generally rely on similarity measure to automatically create groups of web sessions, like the famous k-means which was used by *Chitraa and Thanamni* (2012) or *Si et al.* (2012).

As introduced previously, we present five categories of existing measures as following: each category is composed of a summary table and corresponding description which provides more details on their features. The summary table includes columns

such as "Measure feature" for category features, some corresponding measures of category and their short descriptions are shown in "Measure name" and "Measure description" columns, respectively.

Not taking sequences with different lengths into account

Table 3.1: Measures that are not taking sequences with different lengths into account.

Measure feature	Measure name	Measure description
Not counting sequences with different length	Euclidean	Calculate pairwise distance between sequences based on square root of numerical values, and then produce a symmetric distance matrix.
	Manhattan	Known to be similar to Euclidean distance, it figures out dissimilarity between arrays of numerical values, rather than between sequences of symbol.
	Hamming	Find the position number on sequences where corresponding symbols are not similar. It disregards insertions and deletions to make sequences identical.
	Cosine	Page visits in session are numbered by unique values and Cosine distance is utilized to compute distance between two sessions.

Length of the shortest path connecting two points is commonly referred to when people mention about distance. Accordingly, there have been distances such as Euclidean, given by the Pythagorean theorem, that computes square root of point's coordinates, or Manhattan distance based on the sum of absolute coordinate differences of high dimensional vectors. However, distance between pair of objects could possibly be determined by other information as their properties besides their numeric location in space. Additionally, *Mandal and Azad* (2014) presented the calculation of distance between two sessions using Cosine measure but it requires sessions to have the exact same length, which is more suitable to vectors than web accesses. Furthermore, this approach also ignores regions of local similarity of session pages. *Hair et al.* (2006) denotes the dissimilarity by using XOR on corresponding bits between two binary strings, as illustrated in Figure 3.3. In other words, this metric works on distance between pairs of object as Euclidean, Manhattan or Cosine, thus strings are also required to be of equal length like vectors of symbols, and only replacements allowed. Nevertheless, these strings are considered to be binary vectors. Figure 3.3 is used as a

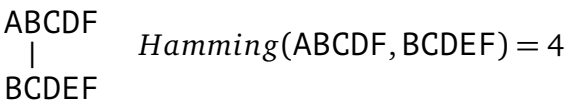


Figure 3.3: Hamming distance is computed by aligning two equal length sequences to count the number of dissimilar symbol pairs.

representation of mentioned alignment methods in this category. These distances, as introduced in Table 3.1, are obviously inappropriate to web sessions which are variable in length. Therefore, their interest is limited in web usage mining that primarily works on session with different length.

Taking sequences with different lengths into account but not the order of element

Table 3.2: Measures that are taking sequences with different lengths into account but not the order of element

Measure feature	Measure name	Measure description
Counting different length sequences but not taking order of element into account	Jaccard	Jaccard coefficient or distance measures the overlap degree among two sets of objects. It shows the chance of a random element from sets union also appears in the overlap.
	VLVD	Comparable to Jaccard, VLVD does not consider the order of page visits but counts the distinct pages between two sessions to assess their dissimilarity.

Statistical distances are also used in finding similarity between sequences. Jaccard returns the value of intersection size divided by union size as the similarity coefficient of two sequences of symbol. In other words, the correspondence and diversity are compared to obtain the distance between two symbol sets. A simple example describing how Jaccard works is presented in Figure 3.4. *Vorontsov et al.* (2013) applied Jaccard as a metric of Positional weight matrices similarity. *Poornalatha and Raghavendra* (2011b) proposed the function VLVD that is comparable to Jaccard. Like Jaccard, that is capable of working on strings of variable length, VLVD handles web sessions regardless of the length difference. However, both approaches quantitatively define the correspondence between sequences through simple count of common contained elements. This kind of approach may be suitable for transactional data as in (*Bouguessa*,

$$\begin{matrix}\mathsf{ABCDE}\\ \mathsf{EDCBA}\end{matrix} \quad Jaccard(\mathsf{ABCDE}, \mathsf{EDCBA}) = \frac{(\mathsf{ABCDE} \cap \mathsf{EDCBA})}{(\mathsf{ABCDE} \cup \mathsf{EDCBA})} = 1$$

Figure 3.4: Jaccard index of the sequences pair equals to 1, hence the corresponding Jaccard distance is 0

2011) which compares homogeneity of sequence pair by frequency of common items occurrence, but it does not take into account their order which is a key feature to differentiate sessions. The quantitative approaches yet progressively more qualitative are mentioned in the next categories.

The feature that these measures haves in common, as presented in Table 3.2, is the ignorance of element appearance order in sequences. As they regard the common pages visited as sequence similarity indication, they could not target different visitor behavior groups. In other words, to understand user interest through their navigational pattern discovery, visiting page A then B and visiting page B then A should not be considered the same. Features of methods in the category is given by the example in Figure 3.4. Consequently, $Jaccard(\mathsf{ABCDE}, \mathsf{EDCBA}) = 1$.

Taking sequences with different lengths and order of elements into account but not their succession

Table 3.3: Measures that are taking sequences with different lengths and order of elements into account but not their succession

Measure feature	Measure name	Measure description
Counting different length sequences, taking order of element into account but not their succession	Levenshtein	This distance is characterized as the minimum number of element needs to insert, delete or replace to convert between two sessions to assess their dissimilarity
	Algiriyage	An approach applying Levenshtein to calculate similarity of navigational sequences. The similarity level of two sessions is considered by this and time correspondence spent on pages
	LCS	LCS detects a subsequence of maximal length that presents in both compared strings. This subsequence ignores the continuity of matched symbols between two strings.
	NW	NW measures the global similarity between two sequences. When sequences are in similar length, it guarantees to find their best alignment
	SAM	SAM (also called string edit distance) is a non-Euclidean distance. It scores similarity of sequences through insertions and deletions of unique elements and transpositions of common elements to make sequences identical. However, SAM discounts sequence length.
	FOGSAA	Fast Optimal Global Sequence Alignment Algorithm (FOGSAA) globally explores two sequences by establishing a branch and bound tree where each path from root to leaf is shown as an achievable path to align the sequence pair, then greedily grows branches to end up with a best path.
	DTW	Dynamic Time Warping (DTW) has been recognized as a good alignment method for time-series. Nonetheless, DTW neglects the identical continuing elements in sequences due to its flexible transformation allowance on time series so that their similar shapes can be revealed.

Main features of metrics in this category is shortly shown in Table 3.3. Referred to as edit distance, Levenshtein is a string metric to score the difference between two sequences. Different from Hamming, Levenshtein distance is able to deal with se-

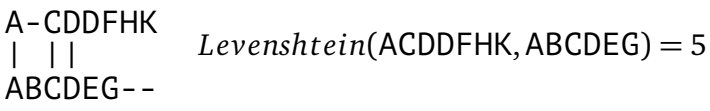


Figure 3.5: Levenshtein distance is computed by counting the minimal number of single-symbol edit operations.

quences of different length, hence not only substitutions but deletions and insertions of elements are used to transform one sequence into another. Levenshtein guarantees to find the minimal number of edit operators which each changes one symbol into the other. Its extension, Damerau-Levenshtein, even permits transposition that reverse the order of sequence elements. Web access similarity measure presented by *Algiriyage et al.* (2015) is one of the approaches using Levenshtein to calculate navigation sequence similarity (in collaboration with time spent on page by visitors) but not identifying the succession of common visits.

There have been similarity algorithms working on different length sequences and recognizing the importance of their items order. Longest Common Subsequence (LCS) is another edit distance which seeks for unique longest common sequences with the concatenation of separate pieces between two strings Thus, the LCS length indicates the similarity and number of unpaired symbols to produce the dissimilarity between sequences. It does not take mismatches and the continuation of items in strings into account, hence it only allows deletions and insertions when building longest common sequence. Likewise, regardless of the disjointedness when aligning over the entire length of two biological sequences, *Needleman and Wunsch* (1970) (NW) computes the homology of sequences globally. The similarity score resulted by this algorithm is optimal because NW is using dynamic programming (DP) to compare sequences. DP is a well-known computational method which repeatedly breaks a complex problem into smaller parts to facilitate its resolution (*Navarro*, 2001). *Lu et al.* (2005) studied how to generate significant usage patterns using NW, however it ignores consecution that is essential to evaluate similarity of web session pairs. Alternatively, NW can be modified to adapt to $semi-global$ alignment when start and end gaps are disregarded on purpose to find considerable overlaps of sequences. Accordingly, gaps appearing before the first element and after the last one are not taken into consideration while scoring.

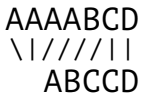
Another suggested measure is SAM (*Hay et al.*, 2004, 2002, 2001), which operates

as a generalized version of the edit distance through calculating deletion, insertion cost for unique pages and re-ordering cost for common pages, that needed to equalize web session pairs. Repeatedly, like the two previous approaches, SAM respects the sequential order of elements but not their continuity as it uses open sequences to evaluate the experimental result. For certain applications such as web prediction and recommendation, user preferences mining, the succession of common pages are essential to detect hidden patterns.

An example of Levenshtein distance is shown in Figure 3.5 to exemplify this category of methods. The alignment considers two sequences of different lengths, it respects their element order but does not count the separation of matches in the distance result. In other words, Levenshtein is able to deal with unequal length sequences, to count sequence element order but fail to grasp element succession.

Different from the matrix or grid which is built by dynamic programming like NW, but somehow related to their back tracking process, FOGSAA (*Chakraborty and Bandyopadhyay*, 2013b) is a similarity measure which attempts to find the optimal alignment path of two sequences by greedy pairwise alignment. This method repetitively expands possible paths until no better path is found, at each node on the way, accumulating similarity score. On the other hand, best and worst alignment scores of two sequences are defined before traversing paths. In order to select the best symbol as node to add into optimal alignment path, maximum or minimum *fitness score* at each node is considered to be the sum of current accumulated score and the best or worst alignment score. Consequently, the node with the greatest maximum fitness score is chosen to continue the expansion, and so forth. Other nodes with lower maximum fitness score are stored in a priority queue and ordered by their values to be used later on, if there is another candidate path through. Either FOGSAA finishes with an optimal path or finds the correspondence of the two sequences to be lower than 30% at any point on the way. It ends up with the path accumulated score as their similarity degree.

A unique method which works differently from others in this category, as it is not a pairwise alignment method and does not allow gaps, is DTW (*Petitjean et al.*, 2014b). DTW optimally minimizes the cost function (*Nakamura and Kudo*, 2011) (*i.e.* distance between pair of data point or event sequences) whereas NW optimally maximizes similarity score. As a result, DTW measures the dissimilarity between sequences. In context of web usage mining, two sessions with less dissimilarity in page visits are



$$DTW(\text{AAAABCD}, \text{ABCCD}) = 0$$

Figure 3.6: DTW score is equal to zero as successive identical symbols in sequences are considered to be one.

more similar, then DTW can be taken into account in considering proposed methods of sequence alignment. DTW aligns one individual element from one sequence with potentially many identical and consecutive elements in the other, if they all are alike, which makes the warping path segment vertical or horizontal. In other words, in this case, identical and consecutive elements are merged into only one. This is a drawback in our application as a series of duplicate visits is a visitor behavior to consider. Therefore, DTW is appropriate for time series stretching or compressing but not for mining web visits. The sequence alignment mechanism of DTW, which returns the distance, is briefly explained in Figure 3.6.

Taking sequences with different lengths, order of elements and locally their succession into account

Together with NW, *Smith and Waterman* (1981) is one of two popular alignment approaches implementing dynamic programming (*Likic*, 2008; *Zahid et al.*, 2015). Both NW and SW take into account the similarity between sequences in different alignments (*Yan et al.*, 2013) and have their own strengths and drawbacks (*Giegerich and Wheeler*, 1996). Exploring local regions, which is in contrary to NW, of high similarity between protein or nucleotide sequences, SW optimizes the adjustment to maximize the similarity score. As ones can see in Figure 3.7, SW locally detects similarity segments of two sequences, and hence local alignment algorithm such as SW can only detects partial similarities (*Aruk et al.*, 2012). If gaps are only allowed at the ends instead of all over sequences to detect if one sequence is partially a substring of the other, it is not SW but a *semi-global* alignment. Stimulated by the basic function of SW and to make it more global, SABDM (*Poornalatha and Raghavendra*, 2011a) takes advantage of SW score and the length of the longest sequence in web session pair to evaluate their likelihood. As a result, it focuses on local similarity and discounts the unaligned pages even if they have a significant effect on similarity comparison.

Table 3.4: Measures that are taking sequences with different lengths, order of elements and locally their succession into account

Measure feature	Measure name	Measure description
Counting different length sequences, taking order of element and locally their succession into account	SW	Built on but contrary to NW, SW optimally searches for local similarity segments and deals with any possible regions lengths. It is appropriate to work with sequences of unequal length.
	SABDM	This method takes advantages of SW to seek local identical portions between two sequences and then calculate the ratio of their length and longer sequence length to score the similarity.
	N-gram	N-gram utilizes sub-sequences of length N to compute similarity score between sequences. Accordingly, the matching of small segments might lead to the correspondence of whole sequences.
	SBS	SBS (Similarity Between Sessions) is an approach of sequence similarity evaluation through a characteristic sequence that covers all of possible sequence pattern. Thus, two sequences are not similar if one is not the subset of the other.

Another distance in this category, N-gram (*Kohonen*, 1985), focuses on mismatch of length N when comparing two sequences. It is called a *bigram* in case that $N = 2$, *trigram* in case that $N = 3$, and used as a threshold to recognize the similarity of sequences. Namely, if it does not exist any mismatch of length N, then two sequences are similar, and vice versa. One more statistical approach features N-gram (*Kondrak*, 2005) considers not all existing orders in a sequence but only N consecutive symbols. It runs as a probabilistic model to predict the next item based on N-item set before and can be used as a unigram sequence similarity measure. Alternatively, SBS (*Anupama and Gowda*, 2015) considers the similarity between web sessions by being subsets of a representative session, that carries most achievable patterns of browsing page occurrence. It leads to a similarity measure defined through value of sequences subset occurrence. Therefore, if one sequence is not the subset of the other, even by one different item, they are supposed to be different.

As the similarity between sessions should be evaluated over their entire length, regional similarity metrics is not appropriate to web usage mining. They do not make sessions well clustered. Providing scoring scheme of match is 1, SW with similarity score is illustrated in Figure 3.7 and can be regarded as the representative method of

```
ABCDEFG
 ||  ||
-BC-EF-
```
$SW(\text{ABCDEFG}, \text{BCEF}) = 4$

Figure 3.7: Scoring SW alignment by counting pairwise matches between two sequences.

this category. Additionally, their features are outlined in Table 3.4.

Taking sequences with different lengths, order of elements and their global and local matching into account

Table 3.5: Measures that are taking sequences with different lengths, order of elements and their global and local matching into account

Measure feature	Measure name	Measure description
Counting different length sequences, taking their global and local matching into account	HSAM	This proposed distance measure is a hybrid of SAM and SABDM. As a result, Hybrid Sequence Alignment Measure (HSAM) can globally and locally deal with uneven length web sessions without altering page order.
	Combination	The combination between global and local alignments based on dynamic programming like NW and SW to perform a comprehensive similarity evaluation
	Hybrid	Similar to the Combination, Hybrid is a composite of NW and SW but does not equally take global and local pairwise alignment into account in scoring similarity between two sequences
	S^3M	Take into account both element existence and their occurrence order when assessing the sequence correspondence by a collaboration of featured LCS and Jaccard

In this category, approaches fully take both quantitative and qualitative aspects when considering the similarity of web sessions, as presented in Table 3.5. HSAM (*Poornalatha and Prakash*, 2013) incorporates SAM into SABDM to take advantage of their effectiveness and reduce shortcomings of the two approaches. By this manner, distance between sequences is measured through unique elements and number of alignments with and without gap insertions. Consequently, a regional and overall alignment is performed on sequences. However, there may be some misunderstanding in the distance formula that causes confusion between aligned page number and operation cost needed to insert gaps in order to create it. Nonetheless, the methodology

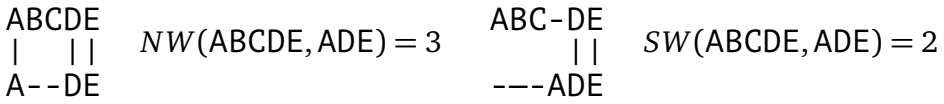
to build HSAM distance formula is improper. Since the NDA (Number of Direct Alignments) value of original aligned session pairs and NAP (Number of Aligned Pages) value of aligned session pairs by inserting gaps are independent, their difference does not show the actual number of pages that could be aligned by inserting gaps. Additionally, although the diversity $|NAP - NDA|$ in the formula is supposed to be directly proportional to the distance, if $NAP > NDA$, two sequences may be more similar than the opposite case when $NAP < NDA$.

The idea of Hybrid (*Chordia and Adhiya*, 2011; *Dimopoulos et al.*, 2010) and Combination (*Luu et al.*, 2015a) is a considerable combination of NW and SW. Since this merging strategy incorporates global into local similarity algorithm, it makes the output much better in similarity than from single ones. Figure 3.8 shows global and local alignments illustration of a sequence pair by NW and SW respectively. *Chordia and Adhiya* (2011) described a hybrid measure, concerning a consolidation of global and local sequence similarity scoring. Similarly, a measure was developed by *Dimopoulos et al.* (2010) to measure the similarity between two sequences. According to these methods, distance between two sequences are computed by taking global, local alignment and their weights into consideration. These weights are inversely proportional to each other, depending on the sequence difference in length. Specifically, local alignment weight would be greater if sequence lengths are different. Thus, the more difference in sequences length, the more local alignment should be taken into account, and vice versa. Because it turns to be global for shorter sequence and local for longer sequence, this measure is meaningful in some specific situations. Nevertheless, it is not really useful in web accesses similarity evaluation as a local alignment scoring scheme like SW does not count the rest of difference in length of sequences. Indeed the longer these parts lengths are, the more important it should be in similarity measure when sequence lengths are significantly different. In other words, as local alignment scoring does not take the difference in sequence lengths into account, the dissimilarity in visitor browsing behavior is not revealed.

As Combination (*Luu et al.*, 2015a) considers NW and SW equally, it comes up with the empirical improvement of Hybrid. Namely, the scoring of local and global similarity in Combination is not affected by the difference of session length as in Hybrid. Unlike biological sequences with comparative lengths as the system proposed by *Brudno et al.* (2003), sessions have a variation of lengths appearing frequently through

variable user behavior (which is valuable for e-commerce to target different groups).

Proposed by *Kumar et al.* (2011), Sequence and Set Similarity Measure(S^3M) integrates the element content of sequences into their order information. Jaccard metric is adopted to find the ratio of common and unique elements of the two session sequences which indicates the similarity in their composition. The approach also exploits the proportion of LCS length to longer sequence length in order to reflect the similarity of element order across two sequences. These two aspects are then merged in a sum, with their own coefficients as relative weights. Therefore, coefficient values and their total range between 0 and 1. These coefficients have to be defined by the users.



(a) (b)

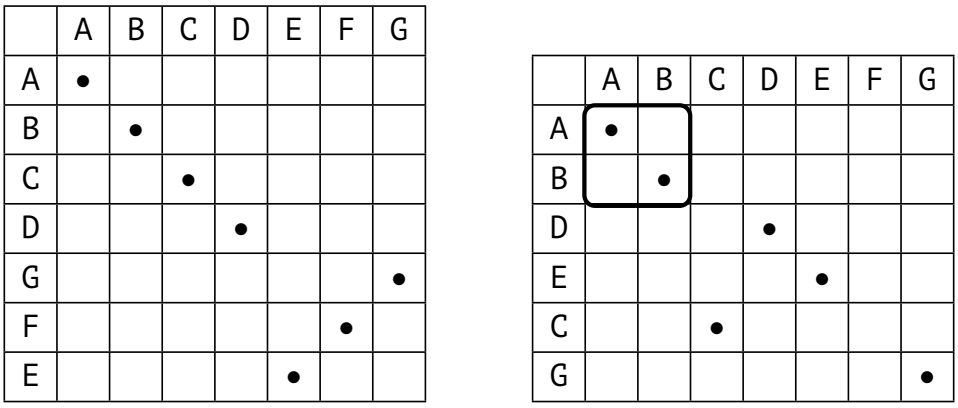
Figure 3.8: The difference between NW similarity and SW similarity, applying the same scoring scheme.

Other approaches

Dot-matrix (DM) (*Sonnhammer and Durbin*, 1995) treats two sequences as dimensions of a dot-matrix plot so that their similar regions can be revealed. Nevertheless, DM is not an alignment method as it just visually compares and shows similarity regions but ignores gaps or considers them to be mismatches between sequences. Although DM does not score the comparison, it provides a better view of possible alignments that can be made through pair of sequences than other alignment methods. Yet another unique feature of DM is the ability of *inverse- matching* exposure. For instance ABCD in one sequence and DCBA in the other, which is useful in collaboration with edit-distance methods which allow transposition like Damerau-Levenshtein or SAM (*Hay et al.*, 2002). It is also a useful extension of other edit distances disallowing transposition like Levenshtein or LCS. However, it can be improved to be a formal alignment by adding gaps (corresponding to horizontal and vertical gaps in dot-matrix) to link similarity segments and then applying some scoring schemes. Alternatively, the distinct level of dot color shades may be adopted to facilitate the different degree of matching

between symbols. In addition, a combination of dot plots and n-gram (*Maetschke et al.*, 2010) may help to avoid common noise randomly caused by cross-matches of long sequences. In order to print countable matching dots, this kind of combination needs a definition of window size n and a ratio of matches over window size as threshold.

Figure 3.9 shows how two sequences can be aligned using dot matrix. In Figure 3.9a, the corresponding symbol matches between ABCDEFG and ABCDGFE is shown as 4 dots in the main diagonal. The inverse matching portion between the two sequences, which is GFE and EFG, is displayed by 3 other dots. In Figure 3.9b, there are gap and mismatch between two sequences ABCDEFG and ABDECG but their distinction is not presented in the dot plot. Furthermore, a window size of 2 is illustrated by the solid square working as a criteria of similarity in order to filter noise such as the matches of C or even G in two sequences. In other words, AB and DE are recognized as similarity segments between sequences.



(a) (b)

Figure 3.9: In dot matrix method, each sequence is put as an axis of a grid. Subsequently, dots are positioned in cells to represent matching portions of sequences. Visual diagonal lines formed by the dots are used to track the expanse of matches.

Hidden Markov Model (HMM) is a probabilistic sequence alignment method (*Mount*, 2009) which is equivalent to a finite state machine. HMM contains interconnected states, transitions among them, and probability values of each state and transition describing their frequency distributions. Accordingly, a given sequence can be represented by a Markovian path (*i.e.*, next state in path only depends on the current state) through series from initial to terminal hidden states (*Eddy*, 2004) via transitions In

order to align a given set of sequences to a template, HMM finds probabilities of states such as symbols, insertions and deletions and transitions by training a set of unaligned sequences. This training creates a template or family of possible alignments, called *profile*, which is used to characterize the alignment and search for more similar sequences by computing the similarity score between them and the profile. This score, which is sequence probability, is calculated by the multiplication of sequence corresponding state and transition probabilities along the path. From the initial model of training sequences, there may be iterative update models while adding other sequences to compare. If a specific profile is defined in the first place, feature parameters such as transition number or frequency of states can be estimated and deployed to find new sequences with maximum likelihood as a family member. One drawback of this model is the the training set, if it is highly similar and not large enough, initial model will be specifically overtrained. Likewise, alignment characteristics such as global or local type are required to feature in the training set. Nevertheless, HMM profiles have a formal probabilistic basis which provides the true residue frequency at any given position. It does not use or depend on gap penalty or scoring scheme, yet it aligns multiple sequences as well as any other alignment methods.

Figure 3.10 illustrates HMM mechanism of multiple sequence alignment. Each of illustrated nodes has its associated probability that reflects the chance of seeing the corresponding residue or operation at a specific position, thus total of match probabilities in each *Mx*, *Ix*, *Dx* or total of probabilities of transitions initiate from them is equal to *1.0*. As a result, each given sequence from Figure 3.10a is used as an input of Figure 3.10b to generate the identity with its own probability. For instance, if hidden state sequence of the trained model is ABCDEF, a sequence such as ABCDEF from Figure 3.10a will directly go through the match states from $M1$ until reaching the END state. Nonetheless, another sequence from Figure 3.10a consists of a deletion at node 6 like *ABCDEFG* will transition from $M5$ to $D6$ and ends up at the END state. Alternatively A_CDEF from the sequence set consists of an insertion between nodes 1 and 2, the corresponding path goes to $I2$ after $M1$ and before reaching $M2$. This insertion may be iterative depending on the necessary insertion number for aligning to the hidden state sequence of model. An alignment of the HMM profile to any observed sequence derives the most probable state sequence with a corresponding probability. Namely, probability of each sequence from Figure 3.10a are computed by multiplying

probabilities from initial to end states and the multiplication result can be used as sequence similarity score. As in Figure 3.10c, the probability of A_CDEF along the path is calculated as $0.5 \times 0.5 \times 0.5 \times 0.5 \times 0.5 \times 0.2 \times 1.0 = 0.00625$.

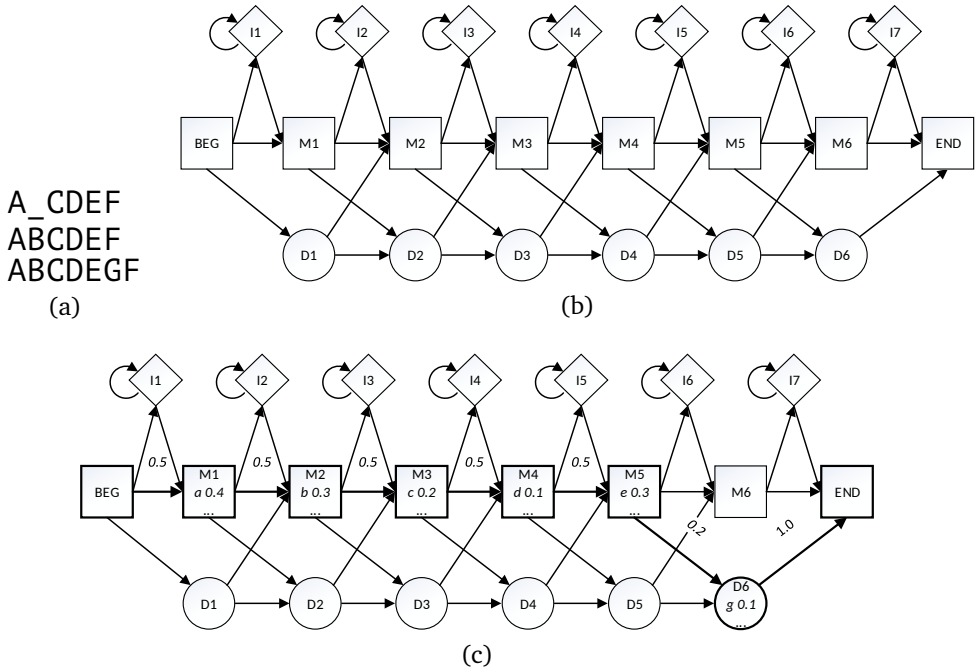


Figure 3.10: Sequence set of symbols (a) are aligned using HMM (b)(c) which is a trained state machine consists of node types: *Mx* represents matches in column *x*, *Dx* represents deletions in column *x*, *Ix* represents insertions in column *x*, arrows represent transitions among them.

Pandi et al. (2011) proposed a novel sequence comparison method through common elements in sequence pair, considering correlation, distance and steps between them. Accordingly, from original pair of sequences, there are two corresponding extracted sequences which hold common elements. Other elements which are uncommon of original sequences are converted to distance between common elements. Considering the two featured sequences to be referred and referring ones, the shortest distances between each connected pair of elements in referred sequence are recorded in order to build a Hasse diagram exploring referring sequence in association with referred one. More specifically, each element of referring sequence is formed as nodes and its connections which satisfied referred sequence connection which is added as edge of the Hasse diagram. The step length of these added edges has to be minimum

in valid candidate edge set. This adding process is iterative and finished when a valid Hasse diagram is built. In the next step, each edge weight is computed based on its distance difference and its step length in the Hasse diagram, and the sum of edge weights is the similarity between referring and referred sequences. Finally, similarity between two original sequences is the average of the two alternative similarity when one takes referred part, the other takes referring part, and vice versa.

Figure 3.11 describes a simple example of how the approach works. The two original sequences as in Figure 3.11a, $S_1 = \texttt{AFBEGCEFDYZ}$ and $S_2 = \texttt{IAIBJJDNCM}$ are converted to featured sequences of $S_1 = (\texttt{1A2B3C3D3})$ and $S_2 = (\texttt{2A2B3D2C2})$ by keeping common elements and changing unique elements to numeric values of distance. For instance, distance between A and B is 2 in both sequences, or it takes 1 from open parenthesis as the beginning of sequence to the first element A of featured S_1. In order to illustrate the case where featured S_1 and featured S_2 are referring and referred sequences respectively, we extract the distance information of featured S_2. Since elements of featured S_2 are unique, distances between them are shortest distances, such as shortest distance of "(" and B is 4, shortest distance of B and D is 3, and so forth. Featured S_1 is used to build the Hasse diagram in Figure 3.11c by adding edges to featured sequence elements in Figure 3.11b as diagram nodes. As ones can see, referring to the coherence of connections in featured S_2, there is no connection between C and D although it exists in S_1. Furthermore, the added edges are of shortest distances since featured S_2 are similar to featured S_1 with unique elements. Consequently, given the weight formula in *Pandi et al.* (2011), where weight of an edge, for example between "(" and A, weighted as $\frac{1}{1(1+|2-1|)}$, total weight of referring sequences can be computed as follows: $S(S_1,S_2) = \frac{1}{1(1+|2-1|)} + \frac{1}{1(1+|1-1|)} + \frac{1}{1(1+|3-5|)} + \frac{1}{2(1+|6-3|)} + \frac{1}{1(1+(3-4))} = 2.46$ as the similarity of referring to referred sequence. The next steps, as previously mentioned, are computing the alternative similarity and final average similarity to indicate the correspondence of the two original sequences.

Discussion

Web usage mining is the exploitation of data mining techniques to discover knowledge from web visitor navigation recordings. In this context, web sessions are considered as sequences. In order to make sequences well clustered, a significant effect on dis-

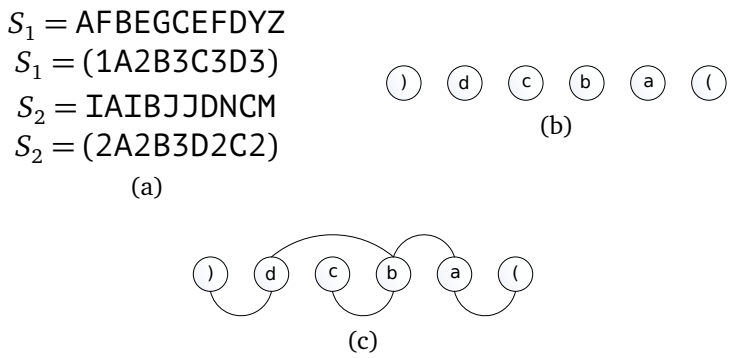


Figure 3.11: Input sequences S_1 and S_2 in (a) are parsed into nodes in (b) and then used to build Hasse diagram in (c).

tinguishing between normality and outliers is not enough. There have been other challenges that a sequence similarity measure has to face to be applicable, efficient and widely used. We describe these challenges in the following sections.

Web applicable features

As click stream generated by visitor interaction with web pages often contains more information besides the order of pages, a similarity or dissimilarity measure would make sense if it can make good use of this information. For instance, if the significance of each web page are identified based on business values through their weights, the extended scoring scheme defined previously can be considered and applied in most of sequence alignment methods. The approach is, by some means, similar to substitution biology matrices Point Accepted Mutation (PAM) (*Mount*, 2008) and BLOcks SUbstitution Matrix (BLOSUM) (*Henikoff and Henikoff*, 1992) to seek for the homogeneity of sequences. These matrices show the dependence of substitution cost upon amino acid sequences involved which were abolished in some NW and SW applications (*Luu et al.*, 2015a; *Poornalatha and Raghavendra*, 2011a).

However, through reducing unnecessary features, many algorithms originally dedicated to bioinformatics have been adapted to web usage mining. For instance, from the web usage mining point of view, number of gaps in session sequence alignment (no matter a gap of length k or k isolated gaps) does not make sense as in bioinformatics, thus the *constant, linear* or *affine gap penalty* may be failed to grasp. Additionally, progressive and iterative alignment as used in T-COFFEE (*Notredame et al.*, 2000) and

MUltiple Sequence Comparison by Log-Expectation (MUSCLE) (*Edgar*, 2004) respectively, are not necessary because there is no need of session compatible arrangement like protein sequences to study their evolutionary relationship. Furthermore, greedy approach such as ClustalW (*Thompson et al.*, 2002) was not maintained as an optimal alignment due to its heuristic and slowness, that caused by "once a gap always a gap"when aligning a sequence with an alignment. Besides, alphabet size of biological sequences is limited to 20 (amino acids) or 4 (DNA, RNA) that is certainly not enough to index all pages of most web sites. They all make biological methodologies and tools inappropriate to web traversal pattern, yet the algorithms can be modified or inspire web usage mining approaches.

Alternatively, web usage pattern discovery can benefit from other similarity aspects of web sessions such as time spent on pages or visit frequency of pages. Considering these aspects in session similarity assessment facilitates the prediction and recommendation performance on web visitor segments. *Hay et al.* (2001) recommended to integrate time spent on pages into SAM as another dimension of non-Euclidean algorithm. *Banerjee and Ghosh* (2001) introduced a web session similarity evaluation method considering the time visitors spent on pages. Nonetheless, only pages reside in the LCS which is the intersection of the two sessions that are taken into consideration. As mentioned previously, this approach has the limitation of catching the succession of the common elements. In a similar way, *Xiao and Zhang* (2001) regard page viewing time of users as one of the factors when measuring similarity between sessions, and *Gündüz and Özsu* (2003) merge normalized time spent on page into the alignment score component together with session length and page order. However, according to *Saleiro*, the approach has one drawback which is the time spent on web pages can be influenced by external reasons. In order to partially improve this kind of method, *Li* (2009) proposed to weigh the interaction time between web server and client before finishing pageload, and page size. Yet the difference between activity duration and the duration from start time to end time on page is another disadvantage of all these approaches, since there is no distinction of time spent on page by scrolling, highlighting, hovering, zooming, etc. and interruption time caused by other irrelevant activities as noise. Another approach of *Chakraborty and Bandyopadhyay* (2013a) takes this problem into consideration through the fraction of time spent on page and page size as well as the portion of that fraction in sum of fraction of a session. However, the

explanation of this portion function is inappropriate since it does not reveal the idle time of web page while displaying for a long time.

One more web feature referred to in *Chakraborty and Bandyopadhyay* (2013a) is page similarity assessment through URL structural similarity. However, since URLs nowadays tend to be shorten by random representation string rather than any fixed structure form, the correspondence in URL levels should not be widely used to evaluate the similarity between pages. Such shortcomings can be found in other similar approaches, for instance *Wang and Zaïane* (2002) and *Li and Lu* (2007). Additionally, web visitor interest is not ruled by URL structure, for example `car.html` and `bike.html` may have exactly the same prefix in their URLs but each page can be accessed by a corresponding visitor group. There are also other elements that could be considered for visitor clustering such as meta tags, `robots.txt` file or titles yet they are not in the scope of this review.

Computational complexity

All the considered algorithms should have an acceptable computational complexity. It refers to space and time cost that are affected by data, algorithm itself, memory and CPU requirement. A collection of algorithms have been implemented not for two but multiple sequences alignment, consisting of accurate but low dynamic programming or less accurate yet faster probabilistic ones. For either dynamic programming applications like NW and SW or edit distances such as LCS and Levenshtein, as they output one result per comparison, their time and space complexity are $O(N^2L^2)$ and $O(L^N)$ respectively (where L is average sequence length and N is number of sequences). Jaccard or DTW equivalently obtains the same time complexity. Time complexity of $O(N^2L^2)$ can be reached, instead of the theoretical $O(L^N)$ due to the discard of diagonal and one of two symmetric parts values computation, as sequence similarity or distance matrix is symmetric. Accordingly, it is problematic (known as NP-complete problem) if the number and length of sequences are large as in commercial web session data because the process would be significantly slowed down and memory space consumption is exponentially increased, although it is an excellent solution compared to exhaustive search of all possible alignment. As web usage mining experiments are commonly conducted on thousands of session or more, it is probably uncertain to draw

conclusion on small datasets.

Chakraborty and Bandyopadhyay (2013b) introduced FOGSAA as a fast sequence alignment. This method is supposed to accomplish a time gain of 35.84% compared to NW for the same optimal output. Nonetheless, such a global alignment as previously referred is not sufficient to satisfy a proper web session alignment.

External validation

Web usage mining detects hidden visitor groups (*i.e.* unsupervised learning) sharing similar behavior of navigation. For that reason, the last but not least challenge is clustering result validation. There have been specific cluster characteristics to quantify the cluster quality such as compactness or separation, which are carried on by internal and external validation. In order to eliminate the bias toward synthetic data in internal validation, external validation should be performed on some given data to prove that a proposed measure can make sense in real web usage situation. To conduct experiments on external data, not any datasets but those containing appropriate segmented data have to be provided.

In one of trendy cluster analysis such as hierarchical clustering, besides the distance/similarity matrix produced by adopted sequence alignment algorithm, the selection of linkage criteria has a considerable impact on accuracy of clusters. Single-linkage mostly takes advantage on elements that have the compact among them and separated with others adequately. On the contrary, complete-linkage is sensitive to irrelevances of segments due to its non-local merge criterion. Another popular criteria, average-linkage, neutrally computes distance between two clusters as the average between elements in one and elements in the other, thus it is commonly slower than others. In internal validation, an appropriate linkage criteria can be selected to relatively suit generated synthetic dataset characteristic, and consequently the clustering result primarily depends on sequence alignment efficiency. However, in external validation, data characteristic and how clusters should be formed from external ground truth are frequently ambiguous, then the decision of the proper linkage among possible criterion is complicated. External validation indices used to evaluate cluster quality such as Rand or Jaccard (*Milligan and Cooper*, 1986) are not able to make a distinction of the influence of sequence alignment algorithm and linkage criteria.

In situation where no appropriate labeled data is available for external benchmarking, internal benchmarking may be used. *Liu et al.* (2010) analyzed the two kind of benchmarkings and concluded the importance of internal benchmarking in multiple validation aspects compared to some external benchmarkings. On the other hand, *Rendón et al.* (2011) revealed that a superior accuracy can be achieved in their scenario by internal instead of external benchmarking.

Conclusion

Sequence alignment methods for web sessions have recently gained significant interest in many web usage clustering applications. In this paper, we introduced sequence alignment mechanism, exhibit a summary of alignment methods which have been used in various approaches of distance and similarity measures. We pointed out their associated features and relationship using a proposed categorization. Despite many approaches in the same category are leaned on a common basic methodology, the utilization details can individually have their own effect on the result. In addition, some possible non-alignment approaches for session similarity evaluation, for instance dot matrix or Hidden Markov Model, are mentioned for further investigations of their application in discovering sequence similarity. We also discussed remaining challenges like web usage context parameters, computational complexity or datasets for clustering validation, so that a significant improvement may be brought to sequence alignment algorithm implementation. Our literature review may expectedly be useful in algorithm comparison and selection to implement web session alignment performance.

Chapter 4

Contributions

During nearly 3 years of PhD study at Université de Haute-Alsace, I have been focusing on web usage mining. From a novice who noticed the changeable web recommendations corresponding to his browsing activities in search engines and e-commerce web sites, I have been getting to know what may lie beneath, and suggested to improve that personalization. In the following brief review, ahead of publication contents, we will introduce three of our publications which leading to this thesis.

In the first step, we focused on the inequality of global and local alignment and time-series measure algorithms to find similarity between sequences. Particularly, we discussed the disadvantage of the previous approaches when handling with sequences of significantly different lengths or duplicate elements. We also presented experimental results to show the opposing views of Needleman-Wunsch (NW) and Smith-Waterman (SW), the important role of longer sequence length in sequence similarity evaluation. The initial combination of NW and SW in this publication, which was presented in PAKDD workshop (*Luu et al.*, 2015a), set up a base for next research works.

In order to improve previous sequence similarity measures, the second paper introduced in MLDM conference (*Luu et al.*, 2016a) aimed to build a new efficient measure that combined global and local alignment measure. The merit of this one compared to competitors, especially when sessions contain consecutively identical pages or session lengths are very different, has been experimentally proved. Accordingly, the session clustering accuracy based on our sequence similarity measure is empirically better than others.

Recently, we have suggested a novel method of web prediction and recommendation using session clustering result. Shown in ICDIM conference (*Luu et al.*, 2016b),

this work considered the ambiguity of prediction and recommendation, and implemented a prediction clustering based on our previous investigation. Additionally, we proposed a new method of recommendation taking page activity time and index into consideration. Furthermore, we discussed adaptation cost estimation to convert web structure into recommender system to benefit the behavioral marketing.

Segmentation using hybrid alignment

We are working on behavioral marketing in the Internet. On one hand we observe the behavior of visitors, and on the other hand we trigger (in real-time) stimulations intended to alter this behavior. Real-time and mass-customization are the two challenges that we have to address. In this paper, we present a hybrid approach for clustering visitor sessions, based on a combination of global and local sequence alignments, such as Needleman-Wunsch and Smith-Waterman. Our goal is to define simple approaches able to address about 80% of visitor sessions to be segmented, and which can be easily turned into small pieces of program, to be run in parallel in thousands of web browsers.

Introduction

Behavioral marketing in the Internet includes adapting web sites to the interests of the visitors in real-time, while they are browsing. Web usage mining has been widely used to transform low-level browsing data (such as page- and click-stream) into actionable knowledge, which makes sense in the business arena. This calls for operators able to compute a measure of similarity between any two sessions, in order to define groups of similar sessions, and further to segment the audience. In our case, as we want to act in real-time, we also have to provide similarity operators which can be executed quickly (in a time which is compatible with the browsing speed of visitors). Sessions can be considered as sequences of events. The granularity of these events can be fine-tuned, from pages-loads down to low-level JavaScript events. In this paper, for the sake of simplicity, we will talk of sequences of symbols, such as A-B-C-D-E-F. Luckily, we have access daily to hundred thousands of such sequences, which are recorded by our industrial partner (BeamPulse). These sequences originate mainly from e-commerce

applications.

There is a large amount experience in sequence analysis in the field of DNA sequences comparison. Sequence alignment has been widely used to identify regions of similarity of DNA, RNA or protein sequences in bioinformatics. Two main approaches - global and local alignment of sequences - have been proposed, respectively by *Needleman and Wunsch* (1970) (NW) and *Smith and Waterman* (1981) (SW). *Wang and Zaïane* (2002) labeled sitemaps as tree structures and compared pairs of sessions using sequences comparison by applying global sequence alignment. In another research, *Li and Lu* (2007) introduced a scoring method by combination of visiting time and URLs similar to (*Wang and Zaïane*, 2002). However, according to *Poornalatha and Raghavendra* (2011a), the optimal similarity between two sequences or clustering outliers can be found by an algorithm based on Smith-Waterman local alignment method only. An approach to cluster web sessions was proposed by *Chordia and Adhiya* (2011) where the clusters are initialized using the longest and most dissimilar sequences. Then, a combination of local and global alignment is used to update the clusters. Alternatively, *Dimopoulos et al.* (2010) modeled users' navigation history and web page content using weighted suffix trees. Their system was then used for the prediction of web page usage. Dynamic Time Warping (DTW), another widely used sequence alignment, has been frequently used to cluster time series (*Petitjean et al.*, 2014a). For example, *Meesrikamolkul et al.* (2012) proposed a method to combine DTW and K-Means to cluster time-series efficiently. They significantly improved the execution time and improved the accuracy of the clusters. Meanwhile, *Nakamura and Kudo* (2011) studied a method named packing alignment to study sequences of various length. This method is partly similar to DTW but allows gaps and limits consecutive events. DTW approach has also been the basis or reference to propose new algorithms, such as (*Marascu et al.*, 2012) taking distance measure based on Longest Common Subsequence (LCSS) having similar distance matrix like DTW into consideration to detect similarity matching in data streams.

This paper aim at introducing a new approach for clustering sessions, by defining a combination of local and global sequence alignments for computing similarity between two pages visit sequences. As we do not target 100% applicability, issues such as impreciseness of sequence glocal (hybrid) metric, caused by the somewhat ignorance of sequences dissimilarity when lengths are uneven, can be overcome.

The remainder of this paper is organized as follows: Section 2 details our approach and provides illustrative examples. Experimental result is described in Section 3. In Section 4, we present related works and explain how our approach compares to theses earlier techniques. Finally, section 5 concludes our paper and suggests some future research directions.

Proposed method

As introduced previously, the Needleman-Wunsch (NW) algorithm, which was developed by Saul B. Needleman and Christian D. Wunsch in 1970, creates a global alignment of two sequences. This algorithm aims at detecting the optimal alignment over the entire length of two sequences. Thus, this algorithm is appropriate to align pair of sequences of similar length. Meanwhile, the Smith-Waterman (SW) algorithm, which introduced by Temple F. Smith and Michael S. Waterman in 1981, is dedicated to local sequence alignment and is then suitable when comparing two sequences with significant difference in lengths. In this paper, we used the NW scoring scheme of +1 for matching and -1 for non-matching pair of items in sequences, and the SW scoring scheme of +2 for matching and -1 for non-matching inside matching, ignore non-matching outside. We selected these two algorithms for their simple and efficient alignment scoring scheme. To detect the best alignment of sequences pair, these two algorithms use a matrix with a number of rows and columns corresponding to the sequences' lengths. This matrix is filled by aligning score between these two sequences, and finally a trace back is performed (*Needleman and Wunsch*, 1970; *Smith and Waterman*, 1981).

NW and SW, by their featured alignments, calculate similarity of sequence pairs in different evaluations. For instance, SW score of Figure 4.1 and NW scores of Figure 4.2 and 4.3 are equal. However, if we take the rate of similarity lengths over sequence lengths into consideration, the similarity of Figure 4.1 is not as much as Figure 4.2 and 4.3.

In the comparison of web access sequences, the pairs of sequences in Figure 4.2 and 4.3 are more likely to be similar than the sequences of Figure 4.1 as they contains the same number of items. However, their similarity scores are mostly the same. Alternatively, Figure 4.4 and 4.5 show cases of pairs of sequences that have the same

ABCDEFGHIJK
A

Figure 4.1: Sequence alignment on two sequences having a common subsequence but different lengths

AB
AB

Figure 4.2: Sequence alignment on two identical sequences

ABCD
ABCE

Figure 4.3: Sequence alignment on two sequences having a common subsequence and similar lengths

length and the same NW score. However, the first pair in Figure 4.4 is more consecutive than the one in Figure 4.5. In other words, SW score of the first pair is higher than the second. This consecution, in our opinion, makes the first pair more similar in web access sequence comparison.

Since the clustering of web sessions is based on the alignment scores, these scores have to reflect the real similarity of the sequences. In our opinion, the *real* similarity of web sessions pair should not only consider a specific rate of common pages but should also take into account the consecution in those common pages. This consecution plays an important role in web usage mining, where the same set of pages but in different order represents dissimilar accessing behaviors. Accordingly, we propose a method to compute what is expected of similar pairs of sequences. As mentioned previously, a global view in sequence alignment is NW strong advantage. Its scoring scheme takes

A**BC**D
X**BC**Y

Figure 4.4: Sequence alignment on two sequences having a common subsequence and similar lengths

A**B**D**C**
X**B**Y**C**

Figure 4.5: Sequence alignment on two sequences having common subsequences and similar lengths

Table 4.1: Rule matching and non-matching pairs in sequence alignments result

	NW score > 2	NW score > 2 and SW score > 10
ABCDEFG BCDEFG	✓	✓
ABCDEFGH ABXDYFGH	✓	✗
ABCDEFG CDEFG	✓	✗
DEFG DEFG	✓	✗
ABCDEF CDEF	✗	✗

both similarity and dissimilarity into account but does not really reflect the consecution of similar items. Therefore, another algorithm focusing on this consecution should be employed to process the result provided by NW. SW is a good candidate as it focuses on local similarity in sequence alignment. Thus, the method proposed in this paper takes the advantages of NW and SW and reduce their disadvantages in web access sequence alignment.

We selected five pairs of sequences: (ABCDEFG, BCDEFG) (ABCDEFGH, ABXDYFGH) (ABCDEFG, CDEFG) (DEFG, DEFG) and (ABCDEF, CDEF) that have specific properties to illustrate the method. We proposed a set of rules that combine NW and SW alignment scores. We expect that similar pairs of sequences match the rules for both alignments. Furthermore, the order of the rules should not affect the final result. We recommend to first check the rules using NW alignment score and then the rules using SW alignment. Using this process, we start by considering a global alignment of the sequences and then a local alignment. In other words, SW alignment works on NW's result. We define rule matching (✓) and rule non-matching (✗) pairs through checking as result in Table 4.1.

The values 2 and 10 have been chosen as initial thresholds based on the average lengths of sequence pairs. As one can see, by defining these thresholds the similar pairs match the NW rule. However, some others pairs such as (ABCDEF, CDEF) does not.

If we want sequence pairs to have a similarity score higher than half of the sequence length, the longer sequences length has to be used within the rule. Integrating this length also allows overcoming another drawback of NW alignment algorithm as NW scoring scheme counts correlation between similarity and dissimilarity but ignores the ratio of similarity/dissimilarity over sequence lengths. Thus, we enhanced the rule to make NW score value dependent of the longer sequence length as described in Table 4.2, with the corresponding coefficient equals to 1/4 then all pairs match.

Table 4.2: Rule matching and non-matching pairs in sequence alignments result after taking longer sequence length into account through its coefficient

	NW score > longer sequence length/4	NW score > longer sequence length/4 and SW score > 10
ABCDEFG BCDEFG	✓	✓
ABCDEFGH ABXDYFGH	✓	✗
ABCDEFG CDEFG	✓	✗
DEFG DEFG	✓	✗
ABCDEF CDEF	✓	✗

However, by applying SW rule as threshold for the expected consecution, many pairs in NW's result are non-matching with this rule. We analyze the non-matching pairs as following:

- ABCDEFGH/ABXDYFGH: Resulting SW score = 10 when aligning with ABCDEFGH or other sequences because of its inner dissimilarity comparing to the other ones. This web access sequence is not similar to the other one in pair because the consecution is not matching SW rule

- CDEFG/DEFG: Resulting SW score = 8 when aligning with the other sequence or itself because of one disadvantages of this approach: SW score set in rule affects the sequence lengths in result, because these lengths have to be equal or greater than the threshold. Nevertheless, this can be improved by setting the

SW score in the rule dependent of the shorter sequence of the set. For example, in order to select pairs that shorter sequence are sub sequence of longer one, similarity length aligned by SW must equal to the shorter sequence length.

With the above given matching score of SW aligning is 2, we change the rule condition from "> 10" to "=shorter sequence length x 2". Corresponding result is in Table 4.3, which shows the final result of proposed combination of NW and SW:

Table 4.3: Rule matching and non-matching pairs in sequence alignment result after taking longer and shorter sequence length into account through their coefficients

	NW score > longer sequence length/4	NW score > longer sequence length/4 and SW score = shorter sequence length x 2
ABCDEFG BCDEFG	✓	✓
ABCDEFGH ABXDYFGH	✓	✗
ABCDEFG CDEFG	✓	✓
DEFG DEFG	✓	✓
ABCDEF CDEF	✓	✓

Another possible approach is binary Dynamic Time Warping (DTW). Back to sequence pair examples from Figure 4.1 to 4.5, the application of DTW results are close to NW. If the rule is, for example, DTW score ≤ 2, pairs in Figure 4.2 to 4.5 are similar and pair in Figure 4.1 is not. The combination of DTW and SW returns a similar result than NW and SW when sequences pair in Figure 4.5 eliminated from similarity set of pairs. In DTW, conditional value in rule can depend on sequence length too, since the sequences pair is considered similar if the dissimilarity is not greater than some threshold

For instance, (AAAA,A) is a case that could not be considered similar in web usage mining context because there might be a reason why a web visitor stayed longer on a page. Nevertheless, DTW does not align with gaps as NW; hence it treats sequence of

identical symbols not as a kind of user accessing behavior but as duplication. Therefore, DTW scores is 0 for this example, no matter how long is the duplication in the longer sequence. Consequently, it is not able to evaluate the dissimilarity in browsing behavior when comparing a specific web page loaded many times with the same page loaded only one time. This limitation makes NW more suitable than DTW in page visit sequence alignment.

Time and Space Complexity: According to *Chan* (2013), time and space complexity of NW and SW are the same, $O(mn)$, given by m and n are sequence lengths. In our proposed method, each sequence pair is aligned by both of algorithms, thus the total time and complexity processing each pair should be $O(mn)$.

Experimental result

The dataset used for the experiments was collected from a University campus website. This website has more than 20,000 visits monthly. A deployed service has taken part in preparation phase (*Cooley et al.*, 1997, 1999) of the clustering process. Written in Javascript and Java, these services allow us to extract information from University campus data like cookies and other associated information such as page visit order, activity time or duration of page visit. In addition, the output format is optional which is convenient to work with variety of mining tool if needed. The extracted information is then checked and validated before applying algorithms to mine them.

Building web access sequences is the next phase. As mentioned earlier and in related works (*Wang and Zaïane*, 2002; *Li and Lu*, 2007; *Poornalatha and Raghavendra*, 2011a; *Chordia and Adhiya*, 2011; *Dimopoulos et al.*, 2010), sequence of visits plays an important role in user behaviors analysis. In order to improve the performance of sequence alignment, URLs have to be shortened optimally by the presentation of symbols set like numbers. Similar to *Poornalatha and Raghavendra* (2011a); *Chordia and Adhiya* (2011); *Dimopoulos et al.* (2010), session contains ordered URLs is as the following example:

1 = http://www.campus-fonderie.uha.fr/fr/droit/

2 = http://www.campus-fonderie.uha.fr/fr/economie-et-societe/

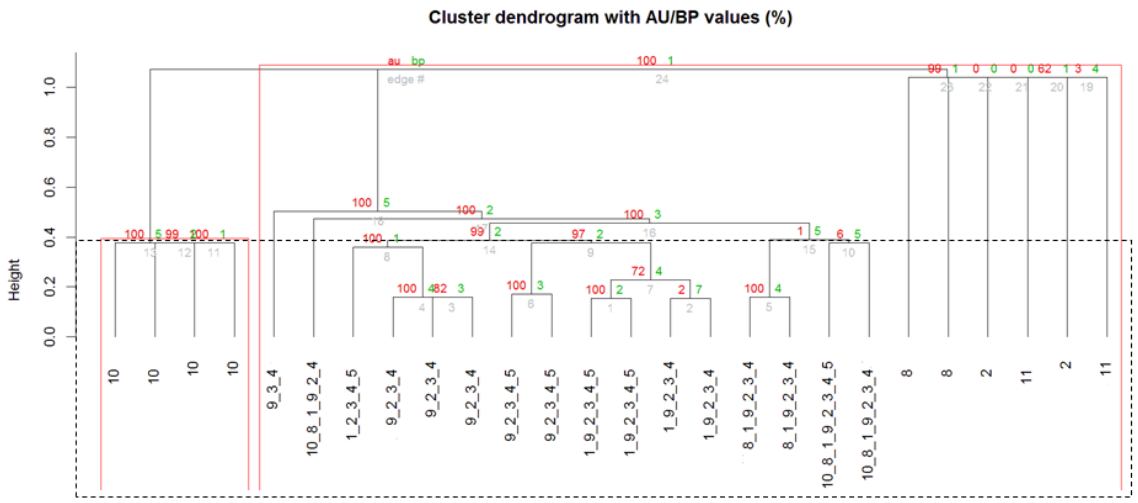


Figure 4.6: Dendrogram of NW score > longer sequence length/4 (NW)

3 = http://www.campus-fonderie.uha.fr/fr/management/

4 = http://www.campus-fonderie.uha.fr/fr/management-interculturel/

will be represented by symbol set {1, 2, 3, 4}, and turn to be page visit sequence like S = 1_2_3_4. In this sequence, page access order is respected and each symbol represents only a unique page. Pairwise alignments are made through all pairs of page visit sequence to score their similarity. As proposed by *Wang and Zaïane* (2002), similarity matrix of web sessions is then computed from this pairwise alignment results.

In the first experiment, we show results on 32 sample sessions that have been selected according to their length, duplication and order of visits as representative of the whole dataset. The goal of this experiment is to highlight specific features of the method. We focus this experiment on three rules: "NW score > longer sequence length/4" (NW), "SW score = shorter sequence length x 2" (named SW), and the combition "NW&SW" (the rule in the last column of Table 4.4). We applied independently the three rules on the similarty matrix obtained by comparing the 32 sequences. We then computed single linkage clustering using the three matrices using R[1]. The clustering results are displayed using dendrograms on Figure 4.6 for NW, 4.7 for SW and 4.8 for NW&SW

As we can see on Figure 4.6, 4.7 and 4.8, there are respectively 26, 32 and 23

[1]https://stat.ethz.ch/R-manual/R-patched/library/stats/html/dendrogram.html

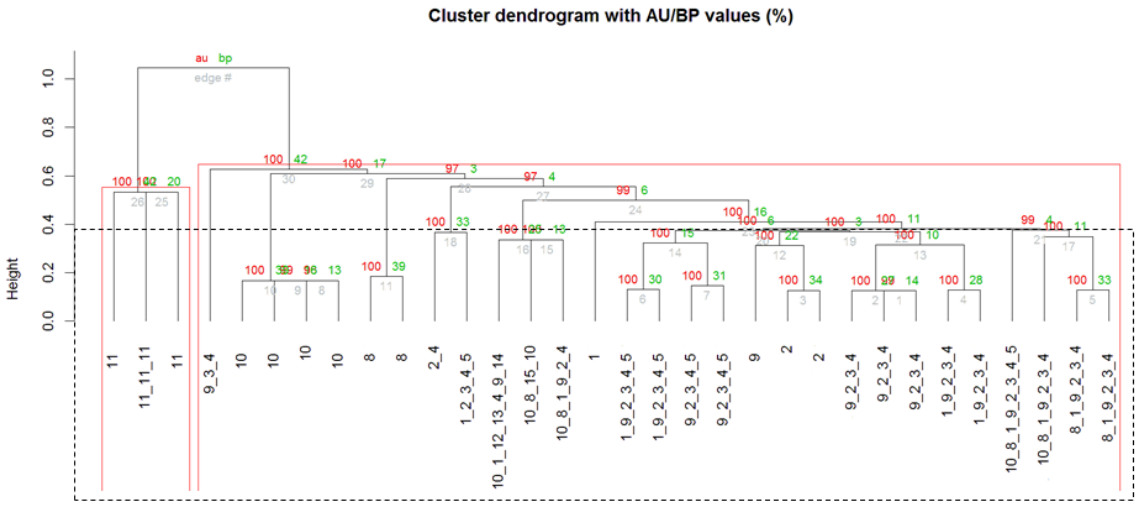


Figure 4.7: Dendrogram of SW score = shorter sequence length x 2 (SW)

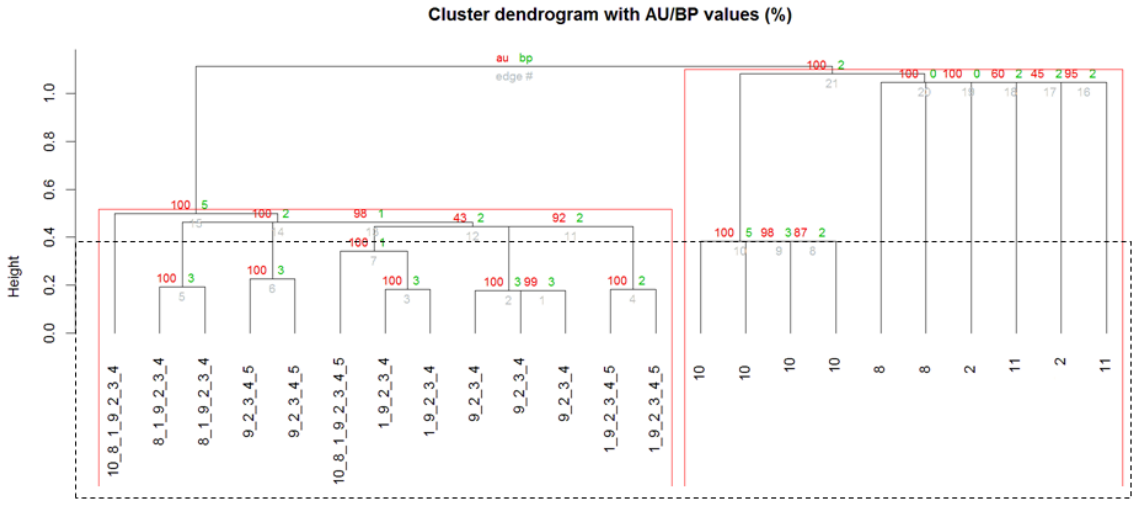


Figure 4.8: Dendrogram of NW score > longer sequence length/4 and SW score = shorter sequence length x 2 (NW&SW)

sessions after the applications of the rules. Applying NW rule results (Figure 4.6) leads to more similar sessions with higher global similar, but sequences like 10_8_1_9_2_4 or 1_2_3_4_5 are not locally similar to others by SW rule. In contrary, applying only SW rule (Figure 4.7) leads to the existence of sequences such as 10_1_12_13_4_9_14 9_3_4 11_11_11,10_8_15_10, 10_8_1_9_2_4, etc. that are not globally similar to others by NW rule.

A noticeable sequence (9_3_4) exists in single rule cases because it is matching either but not both. Finally, the combination of NW and SW (Figure 4.8) rules extracts less but satisfied sequences of global and local similarity.

Consequently, the first top-down pair of Figure 4.8 is different from the others on global and local similarity. Considering clusters at a specific height, for instance 0.4, by looking inside dashed frame, some following features can be seen:

- The number of clusters produced using NW, SW and NW&SW rules are respectively 11, 10 and 13 ;
- The NW&SW similarity inside the clusters is the best, most of the clusters are 100% similar inside, except one. Meanwhile, there are two and three of such clusters in NW and SW. The rate of clusters with 100% inside similarity over number of cluster are respectively 92%, 81% and 70% in NW, SW and NW&SW ;
- Clusters with dissimilarity inside of NW&SW are smaller than with NW, and such clusters using NW are generally smaller than the ones using SW, considering the cluster size by number of sequence ;
- Clusters with dissimilarity inside of NW&SW are more similar than with NW, and such clusters using NW are generally more similar than them in SW, considering the rate of similar sequences over number of sequences ;
- The necessary hierarchical level number by NW, SW and NW&SW rules are respectively 24, 30 and 21.

Experiments performed in the sample of 32 sessions show that the combination of NW and SW rules eliminates dissimilar sequence pairs from the similarity matrix compared to single NW and SW rules. As described previously, the combination of

Table 4.4: Number of clusters on hierarchical tree at some specific levels, by no rule and NW rule.

	No rule	NW score > longer sequence length/4
Hierarchical level 5	8	17
Hierarchical level 7	11	34
Hierarchical level 9	20	49
Hierarchical level 11	33	63

NW and SW rules generates more clusters from less sequence, more clusters 100% similar inside, even smaller and better similarity inside of dissimilar clusters, requires less hierarchical level than single NW and SW rules.

We also performed a similar experiment on another dataset containing 1128 page visit records of 282 sessions. We used the implementation proposed in [2], agglomerative hierarchical clustering algorithm in [3], and implemented an algorithm to get clusters by cutting the tree at a given level. We also include the "no rule" when working on similarity matrix, to compare the number of clusters and execution times among rules application as in Table 4.4, 4.5 and 4.6, that present result after one of about fifty performing times of this experiment. Thus, Table 4.4 shows the number of clusters at some specific hierarchical levels by applying no rule and rule of "NW score > longer sequence length/4". Correspondingly, Table 4.5 is about cluster number by applying rules of "SW score = shorter sequence length x 2" and "NW score > longer sequence length/4 and SW score = shorter sequence length x 2" at same levels. Finally, in Table 4.6 we present execution times after running on corresponding rules. This result presents all mentioned advantages of NW and SW rules combination, as in the previous experiment with 32 sample sessions. Additionally, the execution times are also in the same order of "No rule" > "SW rule" > "NW+SW rule" > "NW rule" like Table 4.6. Therefore, the rules combination of NW and SW is better than others in clustering exactness, with some difference in execution time.

[2] https://code.google.com/p/himmele/source/browse/trunk/Bioinformatics/
[3] https://github.com/lbehnke/hierarchical-clustering-java/tree/master/src

Table 4.5: Number of clusters on hierarchical tree at some specific levels, by SW rule and rule combination of NW and SW

	SW score = shorter sequence length x 2	NW score > longer sequence length/4 and SW score = shorter sequence length x 2
Hierarchical level 5	12	18
Hierarchical level 7	21	37
Hierarchical level 9	29	49
Hierarchical level 11	33	67

Table 4.6: Clustering execution time by no rule, NW rule, SW rule and rule combination of NW and SW

	Execution Time (in second)
No rule	692.7
NW score > longer sequence length/4	558.9
SW score = shorter sequence length x 2	578.0
NW score > longer sequence length/4 and SW score = shorter sequence length x 2	561.3

Related work

Our approach focuses on sequence alignment and ignores URL structural similarity suggested in some previous approaches (*Wang and Zaïane*, 2002; *Li and Lu*, 2007). The reason is, without content mining, the use of website tree structure to represent the similarity of user interest may practically encounter some shortcomings such as:

- URLs are not always in any fixed structure form. Nowadays, they tend to be shorten by some unstructured presentation string
- The similarity of URL structure does not completely reflect the common interest of visitors. Two pages like `math.html` and `art.html` probably explored by two separated visitors groups and maybe shown in different categories on website, though they got the same prefix

In another approach (*Poornalatha and Raghavendra*, 2011a), SW might be more global by counting the longer sequence length in pairwise alignment but our contribution focuses on the combination of primitives. Proposed sequence alignment methods in (*Chordia and Adhiya*, 2011; *Dimopoulos et al.*, 2010), using a hybrid metric by incorporation global into local alignment of 2 sequences. According to the formula, the more different in sequences length, the more local alignment should be taken into account, and vice versa. Because it turns to be global for the shorter sequence and local for the longer sequence, this metric is meaningful in some contexts. Nevertheless, it is not really in ours, because a local alignment scoring scheme like SW not counting the rest different length of sequences, although the longer these parts length is, the more important it will be in similarity metric when sequence lengths are significantly different.

Using similar pairwise alignment implemented by dynamic programming, DTW optimally minimizes the cost function (*Meesrikamolkul et al.*, 2012; *Nakamura and Kudo*, 2011) (*i.e.* distance between pair of sequences) whereas NW optimally maximizes similarity score. As a result, DTW measures the dissimilarity between sequences. In our context of web usage mining, two sessions with less dissimilarity in page visits are more similar, then DTW can be taken into account in considering proposed methods of sequence alignment. As our DTW analysis result above, DTW is appropriate for time series stretching or compressing but not for strings like our approach.

Conclusion

Sequence alignment techniques have been used widely in DNA sequences comparison, and have also been applied to segmentation of Web sessions. However, these techniques were not originally dedicated to web usage clustering, and there is room for optimization in order to adapt these alignments techniques to the specificities of real-time Web marketing, which is our field of application.

We have made the choice of a simple threshold-driven combination of the well-known Needleman-Wunsch and Smith-Waterman global and local alignment techniques. Values of these thresholds can be considered parameters of a given Web site, and we follow currently some simple heuristics to define them.

Our experiences show that our pairwise distance metric, based on the successive alignment of NW and SW in sequence pair, is a simple and realistic way to combine global and local approaches.

With the raise of mobile devices and tablets, there is now a significant difference in terms of low-level events that can observed between those devices and traditional computers (with a mouse). We need to better understand how the granularity of the events included in the sequences affects these thresholds. Therefore, future work is needed to fine-tune our heuristics for setting thresholds.

Segmentation using glocal event alignment

This work takes place in the context of conversion rate optimization by enhancing the user experience during navigation on e-commerce web sites (*Soonsawad*, 2013). The requirement is to be able to segment visitors into meaningful clusters, which can then be targeted with specific call-to-actions, in order to increase the web site turnover. This paper presents an original approach, which equally combines global- and local-alignment techniques (Needleman-Wunsch and Smith-Waterman) in order to automatically segment visitors according to the sequence of visited pages. Experimental results on synthetic datasets show that our approach out-performs other typically used alignment metrics, such as hybrid approaches or Dynamic Time Warping.

Introduction

Conversion rate optimization is considered as one of the most promising approaches for improving the turnover of e-commerce web sites. A lot of researches have already focused on understanding web browsing event-patterns, in order to improve the online content delivery. Clustering visitors into meaningful segments, associated to targeted call-to-actions and related item recommendation, is one of the techniques typically used for cross- and up-selling (*Srivastava et al.*, 2000). As clustering aims at organizing similar items into the same group with no prior knowledge of item class, it is seen as an approach of unsupervised learning. In web usage mining context, the similarity of page visits and their order in a session is one of the relevant information to cluster. For example, cluster analysis helps to reach people who are interested in some specific kind of goods or services so that the owner can recommend to such groups other related things, or offer them some discounts. The clustering result can also be applied to advertising placement organization on web sites, based on page visiting frequency in each cluster.

In this paper, we present our work for computing the similarity between event-sequences of significantly different lengths. Our proposal is based on a new way of equally combining global- and local-alignment techniques (*i.e.* Needleman-Wunsch (*Needleman and Wunsch*, 1970) (NW) and Smith-Waterman (*Smith and Waterman*, 1981) (SW)). The originality of our measure is to take into account the length of longest sequence in the pair of compared sequences.

Thus, regardless of the difference in sequence lengths, the result provided by our metric is accurate and can be used to perform clustering. Experimental results show that our approach outperforms other typically used similarity measures, such as hybrid approaches or Dynamic Time Warping (DTW), in the context of event-sequences of different lengths. This paper is divided into five sections with the following structure: Section 2 explains the proposed method. Section 3 describes experimental results. The discussion of these results is in Section 4. Section 5 presents related work. Finally, Section 6 concludes the paper and gives some future research directions.

Proposed method

Before discussing our sequence alignment approach, we introduce a few basic concepts: Given a finite set Σ whose elements are characters, called alphabet, any possible string of length $k > 0$ over Σ is a k-tuple built by characters from Σ. For example, if $\Sigma = \{A, B, C\}$, a set $S = \{s_1, s_2, \ldots, s_n\}$ of n finite strings over Σ can consist of $s_1 = AB, s_2 = ABC, \ldots, s_n = ACB$. In our model, each web session contains a series of page visits is assumed to be a sequence (*i.e.,* visits which are ordered). Hence, each sequence s_i is composed as a string from Σ, representing a session. A set of navigation sequences S, as mentioned, contains sessions from multiple visitors. To group sessions based on visit order of visitors, our method works as follows: S is processed to create clusters containing comparable sequences that are dissimilar to sequences in other clusters. For this purpose, our alignment-based similarity measure is proposed. An alignment over a set of sessions $S = \{s_1, s_2, \ldots, s_n\}$ can be described as another set $S_a = \{s_{1a}, s_{2a}, \ldots, s_{na}\}$ of equal length sessions which built by adding necessary gap "-"to s_i, for $1 \leq i \leq n$ (*Della Vedova*, 2000). Next, elements at the same *index* of session strings are compared and scored by a scoring scheme, so-called similarity definition.

Sequence similarity definition of specific context (or application-dependent) is essential to perform a relevant similarity evaluation. For instance, the correspondence of DNA sequences (*Qi et al.*, 2015) is not identical to time-series (*Meesrikamolkul et al.*, 2012), and both of them are different to web session similarity. Therefore, session similarity calculation has to be adapted to web usage situation. At first, it has to deal with the variety of session lengths and thus traditional vector distances like Euclidean, Manhattan or even Hamming cannot be applied. Such distances require equal-length sequences like $s_1 = ABCD$, $s_2 = ABCD$. Secondly, as sessions are expected to be differentiated by page visit orders, the appropriate measure has to take into account this order. Thus, metrics such as Levenshtein and Variable Length Vector Distance (VLVD) (*Poornalatha and Raghavendra*, 2011b) are inappropriate as they consider the visit of page$A$ before B and vice versa to be the same. As a result, $s_1 = ABCD$ and $s_2 = BDCA$ are identical. These statistical approaches count the occurrence of element in each sequence to calculate their similarity, regardless of element order. Additionally, the continuity of common pages between two browsing behaviors is a significant factor to evaluate their correspondence. Therefore, Longest Common Subsequence (LCS) or

SAM (*Hay et al.*, 2001) should not be used as it consider sessions started by page A and ended by B, that are common pages, like $s_1 = AB$ and $s_2 = ACDFB$ to be the same, regardless of how many unique pages between them. As the matter of fact, there is a meaningful difference in web visitor interest when one hits C,D and F between A and B and the other hits no pages between those two but they are not counted in this kind of metrics. In summary, an applicable approach in web usage mining context should be able to (1) process sequences with variable length, (2) take the order and succession of common pages into consideration. Such measures are suitable to compute the similarity between two sets of visits.

The NW method is a dynamic programming algorithm for sequence alignment. Dynamic programming makes it possible to find the optimal alignment of sequences, is easy to implement and popular in computer science. When aligning elements of a sequence, matching and mismatching scoring scheme are given. A corresponding score matrix is then established to find the highest score of all possible alignments. In NW, this alignment is carried out from beginning to end of each sequence, it is called a *global alignment*. Global alignment is appropriate to work with sequences of similar length to find their best alignment. However, sequences may inherently not have the same length but might contains similar subsequences. Thus, a *local alignment* is relevant to detect them. To address this issue, the SW method performs the alignment by taking high comparable regions within sequences into account, regardless of the dissimilar parts and even the difference of sequence lengths. These two dynamic programming algorithms are commonly adopted for aligning protein or nucleotide sequences (*Liu et al.*, 2015; *Muhamad et al.*, 2015).

As NW and SW alignment methods are somehow opposite, each one has their own advantages and drawbacks. NW finds the optimal similarity of the entire sequence, while SW detects regions of likeliness between two sequences. As a result, a combination of both methods is better than using a single one, since the correspondence between sequences can be evaluated correctly (*i.e.* globally and locally). We pointed out the effectiveness of the NW and SW rules combination compared to DTW (*Petitjean and Gançarski*, 2012) and hybrid measure (*Dimopoulos et al.*, 2010; *Chordia and Adhiya*, 2011) in our previous work (*Luu et al.*, 2015b). Following this finding, we propose a new similarity measure called *combination*. This measure is based on NW,

SW and the size of the longest sequence:

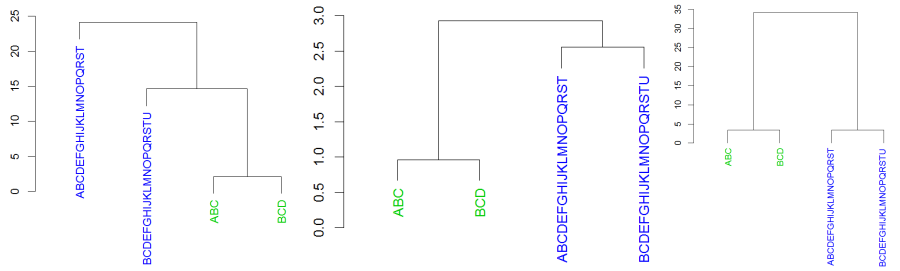
$$S(s_i, s_j) = \left[\frac{NW(s_i, s_j)}{l}\right] + \left[\frac{SW(s_i, s_j)}{(2 * l)}\right] \tag{4.1}$$

with $NW(s_i, s_j)$ and $SW(s_i, s_j)$ respectively NW and SW scores between the two sequences s_i and s_j, l the length of longest sequence in the pair (*i.e.* $max(|s_i|, |s_j|)$), the NW scoring scheme of +1 for matching and -1 for non-matching pair of items in sequences, the SW scoring scheme of +2 for matching and -1 for non-matching inside matching, ignore non-matching outside.

Another way to combine NW and SW was proposed previously through the *hybrid measure* scores similarity (*Dimopoulos et al.*, 2010; *Chordia and Adhiya*, 2011) between two sequences s_i and s_j, which is defined as:

$$S(s_i, s_j) = (1 - p) * SW(s_i, s_j) + p * NW(s_i, s_j) \tag{4.2}$$

with the defined parameter $p = |s_i|/|s_j|$. From this definition (Eq. 4.2) , one can notice that hybrid measure does not equally take both NW and SW into account because of the difference in sequence lengths. As a consequence, the advantage of *combination* (Eq. 4.1) over the *hybrid* (Eq. 4.2) is that the similarity measure works better when sequence lengths are very different. In this case, hybrid measure only focuses on SW, while combination measure focuses on both NW and SW. Consequently, the more different in lengths the sequence pair are, the more the hybrid measure will focus on SW to calculate their similarity. Therefore, the hybrid measure would consider two sequences of different classes to be in same class, if their difference in length and SW are important enough. On the other hand, two sequences of same class with comparable lengths may not be similar enough to be in the same cluster because their similarity score by hybrid measure is smaller than in the previous case. To illustrate this scenario, we created cluster dendrograms with a toy example. The Figure 4.9a shows the similarity evaluation of the hybrid measure in case of sequences with quite different length. As the two sequences of blue class are not similar enough to be merged in agglomerative hierarchical clustering, the resulting clustering is of poor quality. As illustrated in Figure 4.9b, this case does not happen using the combination measure as both NW and SW are considered equally, regardless of the length difference be-



(a) Hybrid measure. (b) Combination measure. (c) DTW

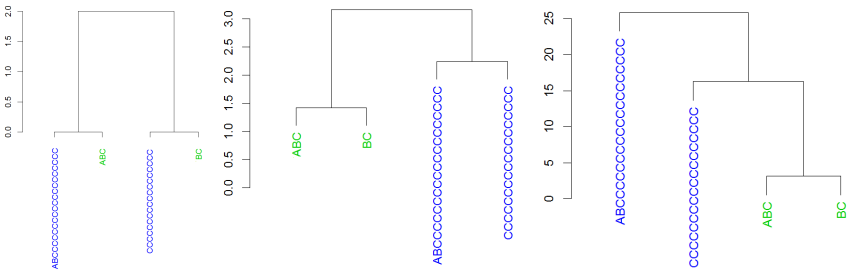
Figure 4.9: Example of clustering of 4 sequences of 2 classes (blue and green) with quite different length for hybrid (a), combination (b) and DTW (c) metrics.

tween the sequences. Consequently, two clusters green and blue of perfect accuracy are obtained when the dendrogram is cut.

Dynamic Time Warping (DTW) is known to be effective to find good alignment between two time-series. However, in the context of symbolic sequence pairs of quite different lengths and no duplicate as in Figure 4.9, DTW works mostly as well as combination measure.

As illustrated in Figure 4.9c, it makes blue and green sequences to be merged in agglomerative hierarchical clustering. However, DTW minimizes the distance of one sequence to another by allowing flexible transformation so that time-series with similar shapes can be detected. This feature leads to a problem when identical consecutive elements in sequences are merged. Thus, the warping path of sequence pairs is vertical or horizontal. In other words, a single element from one sequence is aligned with many successive and duplicate elements in the other sequence as they are all identical. This feature is a drawback in web usage mining as sessions containing duplicate web pages should not be "skipped" but mined for web visitor behavior. Our proposed measure considers them to be traversal pattern and take this duplication into account while DTW regard them as only one page no matter how many visits are duplicated.

Figure 4.10a and 4.10b respectively illustrate the similarity evaluation of the combination measure and DTW in case of sequences with duplicate elements. Furthermore, with sequence pairs which contain duplicate elements and quite different in length like in Figure 4.10, hybrid measure is less effective than the combination one, as presented in Figure 4.10c.



(a) Hybrid measure. (b) Combination measure. (c) DTW

Figure 4.10: Example of clustering of 4 sequences of 2 classes (blue and green) with duplicated elements for hybrid (a) and combination (b) and DTW (c) metrics.

Experimental results

Synthetic data

In order to evaluate the performance of our combination measure, clustering validation including internal and external measures are considered. As some appropriate classified data is not currently available for external validation, we used internal validation on generated synthetic datasets. Nevertheless, *Liu et al.* (2010) analyzed both kind of validations and revealed the relevance of internal validation measure in many aspects over some other external validation measures. *Rendón et al.* (2011) also concluded that they can get better precision by validity internal indexes than external ones, on their datasets and scenarios. To evaluate the performance of the proposed measure and competitors (*i.e. hybrid measure* and *DTW*), we generated 10 synthetic datasets randomly. Each of them contains more than 500 sequences of sessions (about 520 in average) which are grouped into three defined classes with following features:

- Class 1: About 170 sequences of lengths $[20-22]$, sharing a common subsequence, for instance: A**BC**DU3YU31DQ6Q4FO2JGHW, A**BC**DSI5OPHH9EDGLPFLST, A**BC**DAFF5UAK7GEX3XJIU. Generated sequences in other classes are related to this common sub sequence ;

- Class 2: About 170 sequences of lengths $[3-4]$, sharing a common sub-sequence (that is also sub-sequence of common sub-sequence in Class 1), for instance:

Table 4.7: Results for the three methods on the 10 datasets.

	Original datasets								
	Hybrid			DTW			Combination		
	sing.	compl.	avg.	sing.	compl.	avg.	sing.	compl.	avg.
μ	**100%**	**58.5%**	**100%**	**90.5%**	**85.5%**	**89.7%**	**100%**	**98.2%**	**100%**
σ	±0%	±8.5%	±0%	±9.7%	±1.4%	±9.8%	±0%	±5.7%	±-0%

Z**BC**, A**BC**5, **BC**0 ;

- Class 3: About 170 sequences of lengths $[18-20]$, mostly containing identical and consecutive symbols (that appear in the common sub-sequence in Class 1), for instance: DDDDDDDDB**BC**CCCCCCC, DDDDDDDD**C**CAAAAAAA, BBBBBBB**B**BADBBBBBBBBB ;

Note that all the datasets used in the experiments are available for download here[4]. As shown in Section 4.2.2, sequences with common subsequence but different lengths such as Class 1 and 2 are likely to be misclassified by hybrid measure. Furthermore, sequences with same consecutive characters in Class 3 are likely to be confused with sequences of Class 2 using DTW. These three classes are assumed to be representative of the behavior that can be witnessed when analyzing real sessions of web usage.

Using these sequence datasets, similarity matrices for each measure were computed. In order to implement agglomerative hierarchical clustering (*Duraiswamy and Mayil*, 2008), these matrices are then used as input with three well known hierarchical methods: *single-linkage* (sing.), *complete-linkage* (compl.) and *average-linkage* (avg.). This variety of hierarchical methods contributes to the effectiveness of the evaluation. Table 4.7 shows the means (μ) and standard deviations (σ) of clustering result precision for the three methods with the three hierarchical methods over the 10 datasets. Note that as hierarchical clustering is deterministic, running the experiments multiple times is not required. Thus, the means and standard deviations correspond to the execution on the 10 different datasets.

The correlation of experimental results in Table 4.7 are illustrated by dendrograms in Figure 4.11, 4.12 and 4.13 on sample set of the sequences (for sake of clarity) with

[4] https://www.dropbox.com/sh/b6wxv5opn1u3n6n/AAB8ObwvqBPbDsnXvB9xZ_yca

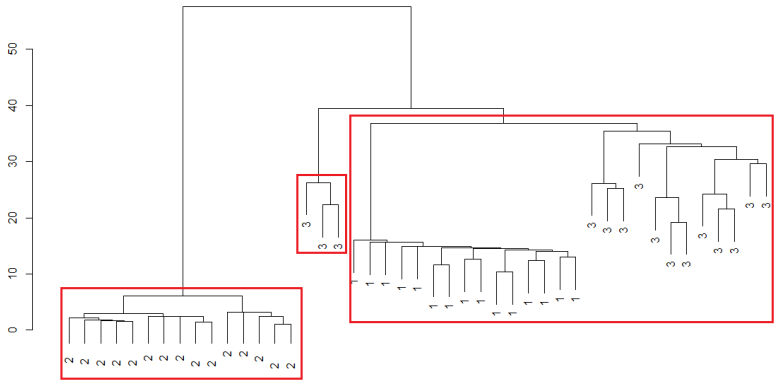


Figure 4.11: Hierarchical clustering using hybrid measure on original dataset.

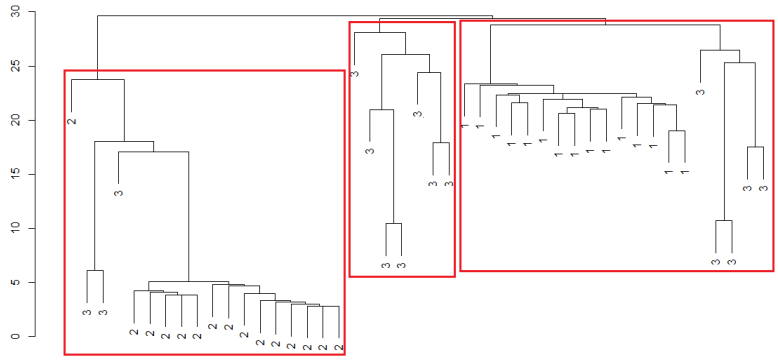


Figure 4.12: Hierarchical clustering using DTW on original dataset.

leave values as defined classes. By cutting dendrograms at the desired level, clusters are separated into frames and their quantity matching the number of defined classes. As shown in Figure 4.11, there is one cluster with unexpected accuracy containing sequences from both Class 1 and 3. The number of inaccurate clusters in Figure 4.12 is two as they correspond to Class 2, 3 and Class 1, 3 sequences. However, clusters in Figure 4.13 achieve an perfect accuracy. These figures also illustrate that combination works very well compare to hybrid and DTW methods.

In the following, we present additional results performed to consider two popular aspects of data in web usage mining: the noise and unbalanced density of classes (*i.e.* classes with important difference in number of elements).

Noise: About 15 sequences of lengths $[3, 24]$ were randomly generated from alphabet and numbers, for instance: APE8V98MDTIH77I, H96YXT7N, M9AKKAA, etc. were

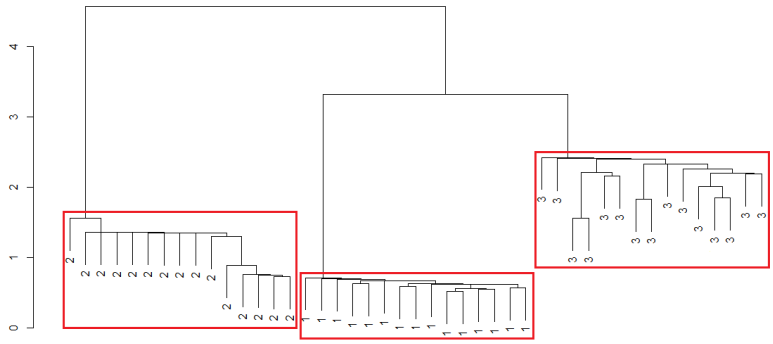


Figure 4.13: Hierarchical clustering using combination measure on original dataset.

Table 4.8: Results for the methods on the 10 datasets with noise.

	Datasets with noise								
	Hybrid			DTW			Combination		
	sing.	compl.	avg.	sing.	compl.	avg.	sing.	compl.	avg.
μ	**90.4%**	**84.3%**	**90%**	**65.8%**	**73.1%**	**86.7%**	**100%**	**89.2%**	**100%**
σ	±11.6%	±3%	±7.2%	±1%	±9%	±11%	±0%	±6%	±-0%

added to the original datasets with 3 classes. The accuracy of clustering results on these datasets is presented in Table 4.8.

Similarly to the previous results, dendrograms on sample of sequences with leave values as defined classes are presented on Figure 4.14, 4.15 and 4.16. These figures illustrate the correlation of experimental results presented in Table 4.8. Dendrograms are cut at the desired level to make cluster quantity match the defined number of classes and separate them into red frames.

Unbalanced density: The number of sequences of Class 1, 2 and 3 are respectively around 320, 170 and 10 in the first three datasets. In the next four datasets, number of sequences in Class 1, 2 and 3 are respectively around 170, 320 and 10. Lastly, in the remaining three datasets, sequence numbers are around 10, 170 and 320 for Class 1, 2 and 3. As the number of users having the same usage can be very different according to specific behaviors, this kind of datasets are assumed to be representative of the data available in web usage mining. The results of the experiments using these unbalanced datasets are presented in Table 4.9.

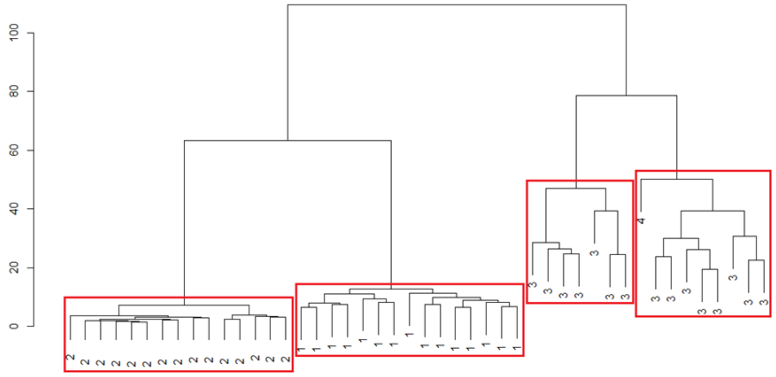


Figure 4.14: Hierarchical clustering using hybrid measure on dataset with noise.

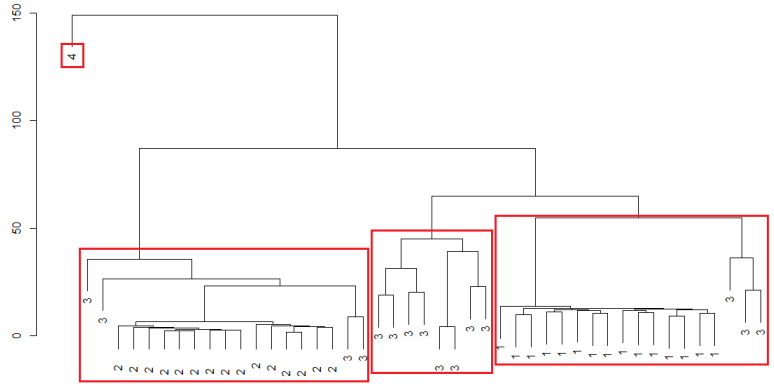


Figure 4.15: Hierarchical clustering using DTW on dataset with noise.

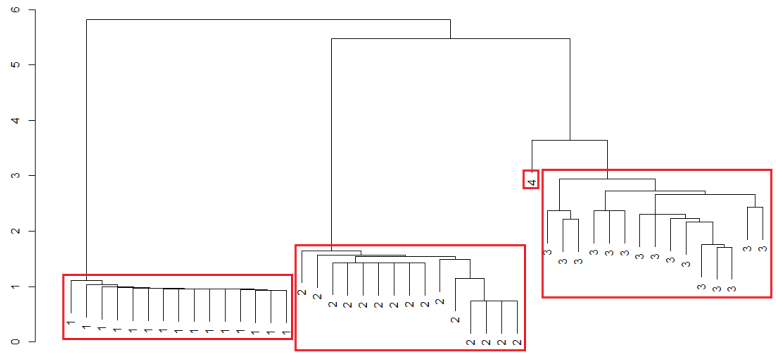


Figure 4.16: Hierarchical clustering using combination measure on dataset with noise.

Table 4.9: Results for the methods on the 10 datasets with unbalanced classes.

	Unbalanced density datasets								
	Hybrid			DTW			Combination		
	sing.	compl.	avg.	sing.	compl.	avg.	sing.	compl.	avg.
μ	**81.6%**	**73.2%**	**84.8%**	**90.9%**	**91.1%**	**91.2%**	**100%**	**93.7%**	**91.1%**
σ	±19%	±18.9%	±24.5%	±16.5%	±12.7%	±12.8%	±0%	±10%	±-0%

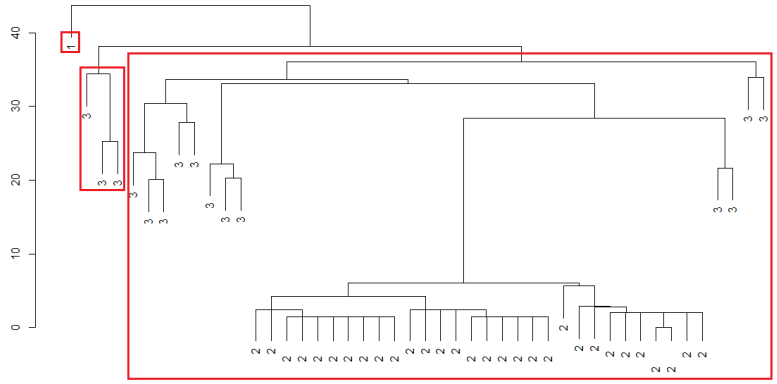


Figure 4.17: Hierarchical clustering using hybrid metric on unbalanced dataset.

As presented before, the experimental results in Table 4.9 are illustrated by dendrograms of sample data in Figure 4.17, 4.18 and 4.19, with leave values as defined classes. Similarly, red frames separate elements into clusters into the defined number of classes by cutting dendrograms at the desired level. Similar to the previous dendrograms, they presents the best accuracy obtained using combination measure compared to the other methods.

Real data

The dataset used for our external validation was collected from a commercial website. The dataset was provided by the Beampulse company which commercializes a product written in Javascript and Java, which collects information about web visitors behaviours such as page visit order, activity time or duration of page visit. As shown in dendrograms in Figure 4.20,4.21 and 4.22 that is a sample extracted from experimental result on 1500 individual sessions, each contains numbered page(s), the

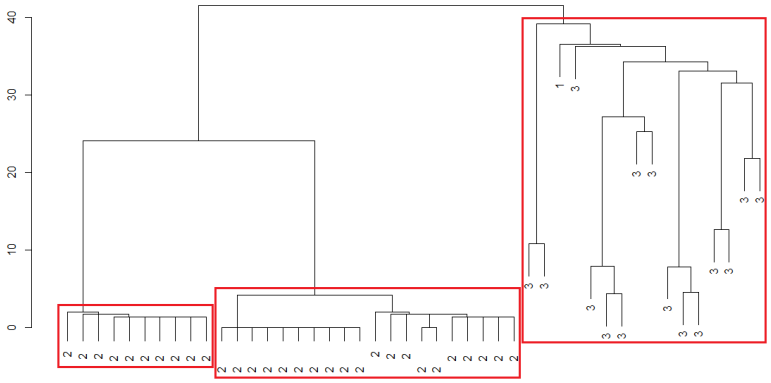


Figure 4.18: Hierarchical clustering using DTW metric on unbalanced dataset.

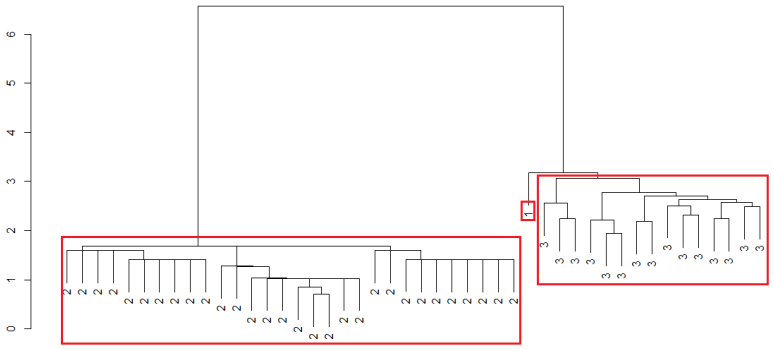


Figure 4.19: Hierarchical clustering using combination measure on unbalanced dataset.

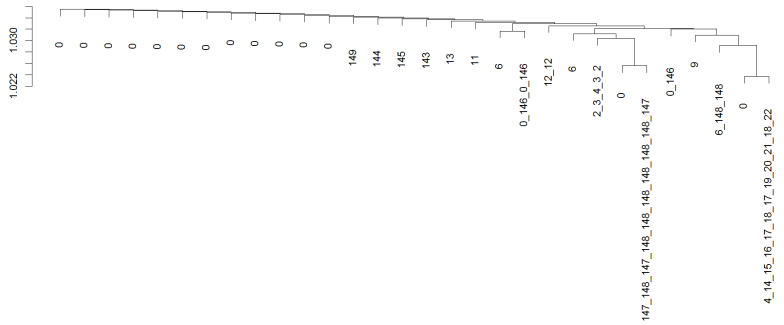


Figure 4.20: Hierarchical clustering using DTW metric on real dataset

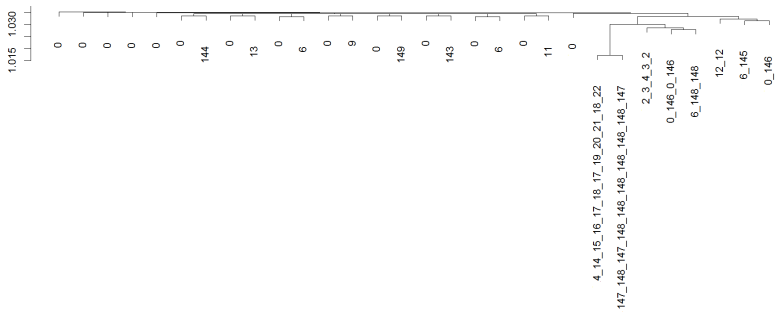


Figure 4.21: Hierarchical clustering using hybrid metric on real dataset

advantages of our metric is highlighted compared to the other methods. Using our metric, sessions with similar pages, similar page order and similar length are likely to be grouped in the same cluster. However, such features are not obtained using hybrid and DTW metrics.

Discussion

As shown in Table 4.7, throughout the 10 normal datasets (*i.e.,* with neither noise nor unbalanced clusters density), similarity matrix produced by DTW metric outputs the lowest precision clustering using single-linkage and average-linkage. However, DTW precision is higher than hybrid metric by complete linkage. Hybrid metric is as good as combination measure using single and average-linkage but is significantly worse using complete-linkage where combination measure is mostly stable.

As noise may impact the performance of clustering algorithm, it is always good to obtain datasets without noise before clustering. However, a clustering algorithm should practically have the ability to deal with noise because of the difficulty to avoid

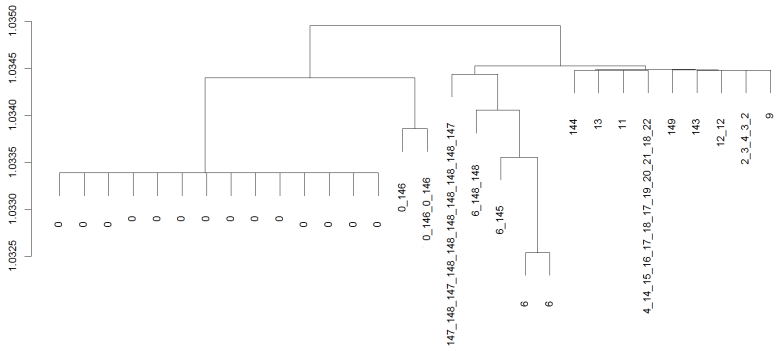


Figure 4.22: Hierarchical clustering using combination measure on real dataset

its presence, especially with big datasets. Our metric maintains a perfect precision using average and single-linkage, and also provides the highest precision using complete-linkage in Table 4.8, where 10 datasets including noise are used as input. Meanwhile, hybrid handles noise more accurately than DTW in sequence similarity evaluation.

Similarly, clustering methods are challenged by various density datasets because of its importance in the clustering process. On unbalanced datasets, the good clustering precision obtained using our metric remain almost the same using single and average hierarchical methods. In contrast to DTW, hybrid metric is highly influenced by unbalanced density. Again, our metric reached the best accuracy using single-linkage and always works better than the other two methods using complete and average linkage (see Table 4.9).

Related work

The goal of web usage mining is to identify hidden pattern from visitor browsing data. This involves clustering of different visits having similar navigational patterns. One of the most popular approaches to discover these clusters is session classification. Among various classification forms, many previous studies have focused on sequence alignment algorithms to evaluate the similarity of sessions. *Mandal and Azad* (2014) presented the calculation of distance between two sessions using Cosine measure but it requires sessions with exactly the same length, which is more suitable to vectors than web accesses. Furthermore, these approaches also ignore regions of local similarity of session pages. *Chitraa and Thanamni* (2012) intended to find usage patterns using k-means algorithm application, yet clustering is to discover hidden pattern without

specific input parameters.Similarly, defining k by the granularity that clusters should be in order to group web users proposed by (*Si et al.*, 2012) is inappropriate to browsing sessions.

Pairwise alignment is commonly used to compare sequences optimally. There exist sequence alignment algorithms, both global and local adopted through pairwise alignment such as NW (*Likic*, 2008) and SW (*Zahid et al.*, 2015). These two algorithms take into account the similarity between sequences in different alignment (*Yan et al.*, 2013) and have their own strengths and drawbacks (*Giegerich and Wheeler*, 1996). *Lu et al.* (2005) studied how to generate significant usage patterns using NW, however it ignores consecution that is essential to evaluate similarity of web session pairs. In contrary, a local alignment algorithm such as SW can only detect partial similarities (*Aruk et al.*, 2012).

Consequently, there have been previous works on integrating one into the other to take advantage of the combination. For example, *Brudno et al.* (2003) proposed a system to align genomes with biological features "glocally". *Chordia and Adhiya* (2011) described a hybrid metric, concerning a consolidation of global and local sequence similarity scoring. Correspondingly, the same metric was developed by *Dimopoulos et al.* (2010) to measure the similarity of two sequences. This formula computes the distance between sequences by taking global and local alignment and their weights into consideration. These weights are in inverse proportion to each other, depending on how different sequences are in length. Specifically, local alignment weight would be greater if sequence lengths are different. As local alignment scoring does not take the difference in sequence lengths into account, this computation may work in some specific situations but not in web accesses similarity because that difference reveals the dissimilarity in visitor browsing behavior. Similarly, *Algiriyage et al.* (2015) used Levenshtein distance as a similarity metric for web session pairs but it does not identify the succession of common visits. On the other hand, with regard to sequential characteristic of web session, there has been approach such as *Bouguessa* (2011) compares homogeneity of sequence pair by frequency of common items occurrence but does not take into account their order, that is key feature to differentiate sessions. DTW, that is a popular algorithm in comparing two sequences of events (*Nakamura and Kudo*, 2011) or data points (*Petitjean et al.*, 2014b), is also used in symbolic sequence comparison. However, this kind of approach is not effective in web usage mining since

DTW ignores duplicated elements. Consequently, it is not able to evaluate the dissimilarity in browsing behavior comparing a specific web page loaded many times with the same page loaded only one time.

Note that the lack of available benchmarks in the domain of web usage mining makes the comparison of the different existing methods difficult. In order to address this issue, we released with this paper all datasets[5] (*i.e.* original, with noise and with unbalanced density) that were used for the experiments. We hope that these datasets will be used in future research to compare new contributions.

Conclusion

In this paper, we have presented our contribution to event-sequence comparison, with a specific focus on sequences of significantly different lengths. This new way of combining global- and local-alignment techniques is based on the equal combination of both approaches. We experimentally evaluated this approach in context of clustering web site visitors, by analyzing the browsing patterns. Under those settings, it was observed that our sequence similarity metric outperformed other related techniques. In the close future, we plan to introduce mutations in the current datasets to better stress the combination technique in the presence of noise. We also want to compare the approach with other techniques such as PAM/k-medoids, ROCK or Ward. Furthermore, other distance measures such as Hamming or Levenshtein, which have been studied in a variety of sequence comparisons including spectra (*Perner*, 2014) should be considered in upcoming works.

Web usage prediction and recommendation

In recent years, a strong interest has been given to web usage prediction and recommendation methods to improve e-commerce, search engines and other online applications. There have been various efforts carried out in this field, particularly focused on using recordings of web user interactions with websites. In this context, our research focuses on developing a novel approach for web prediction and recommendation. The proposed method relies on hierarchical session clustering by sequence similarity mea-

[5] https://www.dropbox.com/sh/bse1ifyu2gdywm4/AACRSbKysPVGudinjTBg6Ocsa

sure and takes advantage of access activity time and access position in prediction session to make a recommendation. The performed experiments reveal that hierarchical parameter and prediction accuracy are relevant. In addition, the paper introduces cost estimation to adapt web visitor behavior to web business purposes using prediction and recommendation results.

Introduction

The Internet is today the richest information source in the world. However, with this extremely large amount of information, the problem web visitors are facing is how to reach relevant content and to discard irrelevant resources or unrelated content. Web visitors exhibit various types of behavior through their browsing activities which are captured during their visit. Consequently, it becomes more and more essential to present proper recommendation content adapted to visitor interests. In other words, visitor needs should be understood and taken into account correctly. Although the raise of individual privacy concern can receive negative comments as the privacy violation, visitors' interest is currently the backbone of e-marketing campaigns. Correspondingly, thousands of different web visitors, for example on an online shopping website, may see thousands of distinct versions of the homepage, which is called content personalization. In order to effectively use collected visitor browsing data for such behavioral targeting, web usage prediction and recommendation (WPR) have been adopted and plays a vital role in behavioral targeting strategy of search engines, entertainment and e-commerce websites. Taking advantage of multiple techniques to target visitor, the process predicts the upcoming request of visitors and sends related promotional contents information with specific recommendation. By pre-fetching, pre-sending or caching such recommendations, network latency effect can be also reduced. Since WPR helps to increase revenue growth, it has turned into a fundamental feature of commercial websites, or even helps to improve search engines performance.

Predictive model construction, which indicates the chances of next accesses of visitor browsing, is the earlier phase of WPR process. In order to build this kind of model, Predictive Analytics (PA) (*Hair Jr*, 2007) is among the most appropriate methodologies. Consisting of technologies which assist users in predicting web visitor action, PA appliances are widely known to be efficient in e-commerce marketing, search engines

or other big data systems by instantly analyzing and discovering web usage patterns (*Lee et al.*, 2014). Alternatively, recommender system (*Chan et al.*, 2012) assists web visitors in making real-time choice and even transform them into customers. There have been multiple presentation techniques to guide personalized navigation such as images, text, hyperlinks, etc. with the support of font size, color, etc. to lead visitors to tailored content as a recommendation, and these features have changed the way of interaction between websites and visitors. Later, under the influence of online campaigns, client behavior is altered and hence advanced web data is produced. Overall, the prediction and recommendation work with web data as a mutual reciprocity is illustrated in Figure 4.23.

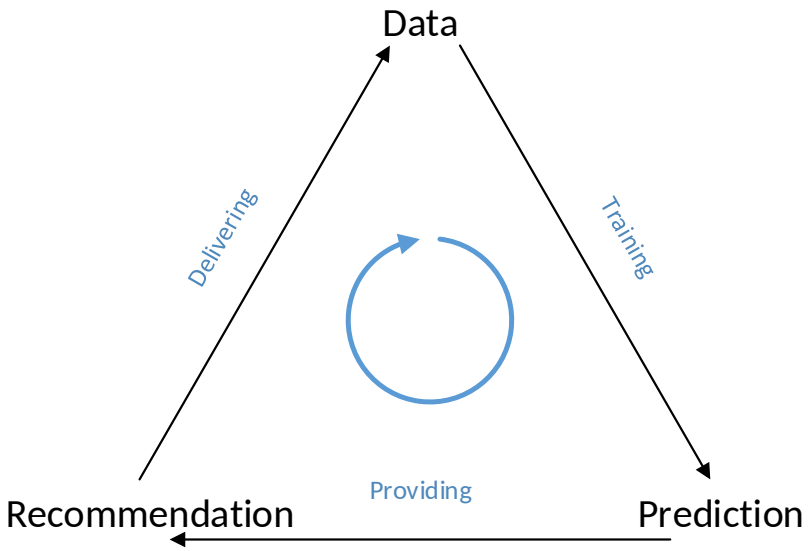


Figure 4.23: Round process of prediction, recommendation and web data

Nonetheless, several works have shown an ambiguity of prediction and recommendation by stirring them together. Obviously, correct prediction of visitor action itself does not reflect what they prefer or is suitable for them. The visitor online behavior should be inferred not necessarily from their custom but according to the web sites or search engines business. Also, a further prediction and recommendation related part is customization cost estimation between web document design and visitor need. In this paper, in order to bridge those research gaps, we introduce a prediction model of web visitor behavior which takes advantage of clustering techniques based on session alignment. The proposed technique produces recommendation patterns without needing web structure but using activity time visitors spend on pages and page indexes,

and a measure for web site structure to visitor interest adaptation cost.

This paper includes five sections as follows: Section 2 presents the proposed WPR model. Section 3 describes the experimental results. Related works are discussed in Section 4. Section 5 shows the conclusion and reveals some research plans for the future.

Proposed method

Prediction

This paper proposes to perform web prediction on top of web session clusters. In order to build relevant session cluster, a similarity measure is adopted to calculate the similarity between train queries data, then clusters are hierarchically constructed. As the resulting clusters contain comparable historical sessions with identical initial pages, it facilitates the prediction process of next accesses from the first ones.

Modified combination measure

The original proposed combination measure (*Luu et al.*, 2016a) was experimentally proved to be appropriate to evaluate the global and local similarity of sequences (*Luu et al.*, 2015a). Nonetheless, clusters of similar sessions are not sufficient to make prediction if session lengths are very short and entry pages are not unified. Firstly, sessions with less than two accesses should be eliminated since they are definitely not suitable for prediction. Second, as an illustration, to forecast next pages that visitor is likely to hit after page A, clusters with sessions initialized by A like (ABCDE, ABCD) should be considered if they are available. Similarly, for more-than-one-page inputs such as AB, clusters with A as the first page will be analyzed to see if they include AB to make prediction. As a consequence, it is inappropriate to use the set (ABCDE, BCDE, CDE) as a corresponding prediction cluster of any specific input access.

Consequently, the sequence similarity measure is required to take this issue into consideration to avoid such situations. In other words, order related sessions but including different first accesses should not be considered as similar and in the same cluster. Accordingly, we modified similarity combination measure by assigning a very low similarity score to session pairs with different prefix, hence only pairs of sessions

which have the duplicate initials are considered. This modification does not impact the association of similar and duplicate initial sessions. In addition, such a *glocal* (*i.e.* global and local) similarity measure performs somehow similarly to association rules or Markov model in "learning" order and relation between elements, which are pages in session.

Clustering

As a tree diagram commonly adopted to build a hierarchical form of clusters, dendrogram does not merely illustrate a cluster set but its multilevel set (*Langfelder et al.*, 2008; *Doan et al.*, 2015). The approach consists in merging two most similar clusters at a level into one at the upper level. Correspondingly, cutting the dendrogram at variable levels outputs different sets of clusters and the context-dependent appropriate level can be selected. Furthermore, for the purpose of making elements in the same cluster similar and different clusters dissimilar, it is not trivial to decide which level fits the best the data. An inappropriate level to stop clustering could create indefinite or over-particular clusters. For instance, combining two clusters containing sequences such as (ABCD, ABC) and (ABCDEFG, ABCDEF) may make the cluster result not as united as the original two. On the contrary, clusters like (ABC) and (ABC) should be joined together as they are identical. We set a heuristic threshold to collect clusters based on their intra and inter distances, as the formula below:

$$diff = max(inter\_distance) - min(intra\_distance) \tag{4.3}$$

Given $inter\_distance$ the distance of elements between single clusters and merged cluster, $intra\_distance$ is the distance of elements inside merged cluster and *diff* is the difference of inter and intra distances.

Diff is used as a threshold value to decide to continue or stop to merge clusters. The larger *diff* is, the more distant clusters are. As we use single-linkage as hierarchical strategy due to its experimental advantage with similarity combination measure, clusters are merged if they are nearest neighbors. In order to observe if two clusters are worth being merged, the minimum distance inside merged cluster is compared to the maximum distance from it to single clusters. For example, if we decide to stop cluster at $diff > 0.1$, then there are more clusters created than $diff > 0.9$ although

these clusters are less separated. This threshold is flexible to find optimal clusters depending on the context, and there is even no cluster created if this threshold value is very high.

Additionally, there might be couples of clusters which include different entry page sessions due to linkage strategy. In order to flush prediction clusters, they will be eliminated from the cluster set. If the session similarity measure is not improved as described above, the number of applicable prediction clusters will be significantly less by this removal, and thus the prediction efficiency will be correspondingly reduced.

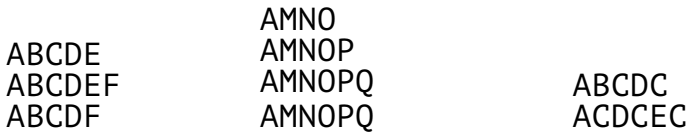
Prediction implementation

If historical sessions are not clustered, time and space consuming investigations through every element of train set must be implemented to consider if a prediction can be made. Accompanying prediction clusters, corresponding session groups of an input can be called and require considerable less time and space. Based on one or more initial prefix of input sessions from the test set, if there exist identical sessions in the matching clusters, the prediction is correct. Figure 4.24 shows an example of page sequence inputs to anticipate next pages, and a complete session to confirm the accuracy of the prediction. Figure 4.25 illustrates three examples of corresponding clusters of inputs in Figure 4.24, with an equal session to the complete session of Figure 4.24 in Cluster 1. In this case, the prediction is then accurate.

A	AB	ABCDEF
(a) Input 1	(b) Input 2	(c) Complete session

Figure 4.24: Possible inputs and complete session to predict, and investigate the prediction accuracy.

Nevertheless, prediction cluster set is generally not able to cover every input to make the entire prediction. Due to the limited size of the train set, removal of unclean clusters, unique or rare queries, etc., there may be access that cannot be predicted in real-time. On the other hand, a proper train should eliminate invalid patterns like input errors, incomplete visitor traces etc., to be best suited for the application context.



(a) Cluster 1 (b) Cluster 2 (c) Cluster 3

Figure 4.25: Three prediction clusters corresponding to Input 1, and Cluster 2 will be eliminated to predict Input 2 in Figure 4.24. Besides, complete session of Figure 4.24 matches the second session of Cluster 1.

Recommendation

The recommendation, in any form, should make visitors more convenient in their browsing. Since recommendation is regularly based on prediction, the prediction should be effective in defining targets. Nevertheless, prediction information is basically not relevant to use in recommendation. In other words, browsing behavior and site proposition are two correlated but distinct concepts. In order to improve the system usability, prediction models are required to be integrated into people aspects. For example, if a visitor has been querying about hotel deals, it may be helpful to show them airline promotions to calculate how much they can save totally, instead of more hotel options they are likely to search, that makes them confused. Otherwise, when three information pages of an online course have been browsed and users are predicted to visit the fourth page, it is completely not essential to suggest the fourth one. Alternatively, it is probably the right time to show subscription benefit or schedule advise before they leave. Consequently, one of the prospective approaches to make a recommendation by taking advantage of prediction is to dynamically recommend new information to the visitor. This kind of approach is context-dependent and based on the categories that visitors are predicted to belong to. Namely, prediction model may recognize the matching categories of a visit through its corresponding prediction clusters, then appropriate suggestions can be performed to make some specific content more accessible. In order to effectively comply with visitor demand, such recommendation should be directly started from the first access and active during the visitor session.

Alternatively, another considerable feature to support visitors is browsing time reduction. As visitors have their targets while accessing a website, one essential thing

we can focus on is time saving to quickly reach those targets. For this purpose, visitors' preferences like browsing order should not be considered as it may provoke a waste of time. It is reasonable to assume that visitors are prone to spend more activity time on their interesting content (*Nielsen*, 2011; *Kim et al.*, 2001) (even opening them in a new tab, keep searching in other sites and then going back to them). The difference between activity time and the duration from start time to end time of a web page is that activity time does not include interruption time caused by other irrelevant activities. Particularly, the total time visitors spent on scrolling, highlighting, hovering, etc. on a web page without an idle time is activity time (*Claypool et al.*, 2001). Furthermore, visitors are likely to leave after reaching these target information (*Shen et al.*, 2008) as the notion of *maximal forward reference* by *Chen et al.* (1996). Following that, a session group or cluster of similar navigational page sequences reveals some interests in visiting order of pages that is good for predicting accesses, but not for interest prediction. For instance, the visit sequences $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$ and $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E \rightarrow F$ may be in the same cluster of navigation but the destination page of sessions could possibly be E or F. Also, B may be kind of a "bridge" such as category or search page to jump to real content, so it does not reflect the common interest of visitors and they are likely to spend a limited time on it. Therefore, their corresponding navigation cluster probably does not make sense in the recommendation as introduced by *Nkweteyim* (2005) or *Jalali et al.* (2010). Briefly, access activity time and index should be two indicators of visitors' interest that should be considered in the recommendation.

According to the previous assumption about the relation between time spent and last visited page to user interest, we assume that two main factors affect the interest prediction of a specific page are: (1) activity time of visitor and (2) the visit order. There may be more than one target for each visitor and these targets are probably of different priorities for the visitor. In this context, a metric of visit destination probability which takes into account position of the page in visit session and time spent on it is proposed. Consequently, this metric shows the probabilities of each page of a session to be interesting for a vistor, by the formula below:

$$Pr(i) = Pos(i) \times T(i) \tag{4.4}$$

Given $Pr(i)$ the interest probability of page i in a session, $Pos(i)$ the position of page i in that session (1^{st} page of session get the position of 1, the second one's posi-

tion is 2, and so forth), and $T(i)$ the time visitor spend on page i by their activities. Repeatedly, $T(i)$ is different from the total access duration from start time to end time on a page, it does not include page loading time, browsing interruption, etc.

For example, according to (2), the interest probability of page C, with $T(\mathsf{C})$ equals to 12 secs, in session $\mathsf{A} \rightarrow \mathsf{B} \rightarrow \mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E}$, is $3 \times 12 = 36$. In the same session, $Pr(\mathsf{E})$ with 5 secs of time and $Pos(\mathsf{E}) = 5$ would be 25. It is noticeable that even an exit visit is not likely to have a high probability if the time spent on it is not significant. In this case, the visitor probably left the site not because they find what they needed. Assuming that $Pr(\mathsf{C}) > Pr(\mathsf{D}) > Pr(\mathsf{E}) > Pr(\mathsf{A}) > Pr(\mathsf{B})$ for the session, the probabilities of recommendation should be similarly, in descending order, such as $\mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E} \rightarrow \mathsf{A} \rightarrow \mathsf{B}$ with C and B having the highest and lowest values of all following (2), respectively. This kind of sequences is a form of recommendation that converted from the prediction session, and we aim at building recommendation clusters from them since such clusters are convenient for the dynamic recommendation process. From a cluster of prediction as described in Figure 4.26, a corresponding recommendation cluster like Figure 4.27 can be derived for example using previously mentioned computation.

$\mathsf{A} \rightarrow \mathsf{B} \rightarrow \mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E}$
$\mathsf{A} \rightarrow \mathsf{B} \rightarrow \mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E}$
$\mathsf{A} \rightarrow \mathsf{B} \rightarrow \mathsf{D} \rightarrow \mathsf{E}$
$\mathsf{A} \rightarrow \mathsf{B} \rightarrow \mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E}$
$\mathsf{A} \rightarrow \mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E}$

Figure 4.26: Cluster of prediction.

$\mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E} \rightarrow \mathsf{A} \rightarrow \mathsf{B}$
$\mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E} \rightarrow \mathsf{A} \rightarrow \mathsf{B}$
$\mathsf{D} \rightarrow \mathsf{E} \rightarrow \mathsf{A} \rightarrow \mathsf{B}$
$\mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E} \rightarrow \mathsf{A} \rightarrow \mathsf{B}$
$\mathsf{C} \rightarrow \mathsf{D} \rightarrow \mathsf{E} \rightarrow \mathsf{A}$

Figure 4.27: Cluster for recommendation.

In order to be in one recommendation cluster, these sequences certainly have to be related in order and length because it reflects a group of common browsing behaviour together with the correlation of time spent on particular pages, which is the subject of this approach. As combination measure clusters sequences based on *glocal* similarity, it makes clustered sessions similar in length. Also, combination measure makes the

position of a specific page in one session not very different from its position in other sessions in the same cluster, if existing. Therefore, a threshold of session correspondence computed by combination measure is mandatory to validate recommendation clusters. If target contents of visitors in prediction cluster are not similar, such visitors are not targeted for the recommendation. Consequently, there may be no recommendation presented as no equivalent recommendation cluster built on those prediction clusters.

Concerning the frequencies of page appearance in recommendation cluster, the recommendation should start with the most frequent page from first to last index, as long as it has not been visited. For example, such frequencies of $\mathsf{C} = 4$ and of $\mathsf{D} = 1$ in 1^{st} index of recommendation cluster, visitors who belong to the prediction cluster in Figure 4.26 are interested mostly in C, then D, E, A and B. Therefore, C should be on the top of recommendation list if it has not been accessed, and so forth. In other words, the page with the highest probability of interest should be recommended first, then other pages recommended in descending probability (or priority).

The dynamic recommendation process indeed exploits the prediction results when monitoring visitor run-time behavior. If this kind of behavior matches prefix of prediction patterns, their recommendation content will appear in multiple forms. Accordingly, when visitor accesses page A or A then B as described in Figure 4.28 for example, prediction cluster in Figure 4.26 may be among the used clusters. Correspondingly, recommendation cluster in Figure 4.27 is then prepared for navigation suggestion. The process works as shortest paths instruction for the visitor so that they can instantly end up with their expected information.

A (a) Input 1 AB (b) Input 2

Figure 4.28: Possible inputs for prediction using navigation cluster in Figure 4.26 and then recommended by recommendation cluster in Figure 4.27.

The development of recommendation clusters from prediction clusters and prediction ones from collected sessions are presented in Figure 4.29. Likewise, the prediction and recommendation steps from visitor opening entries are shown in Figure 4.30.

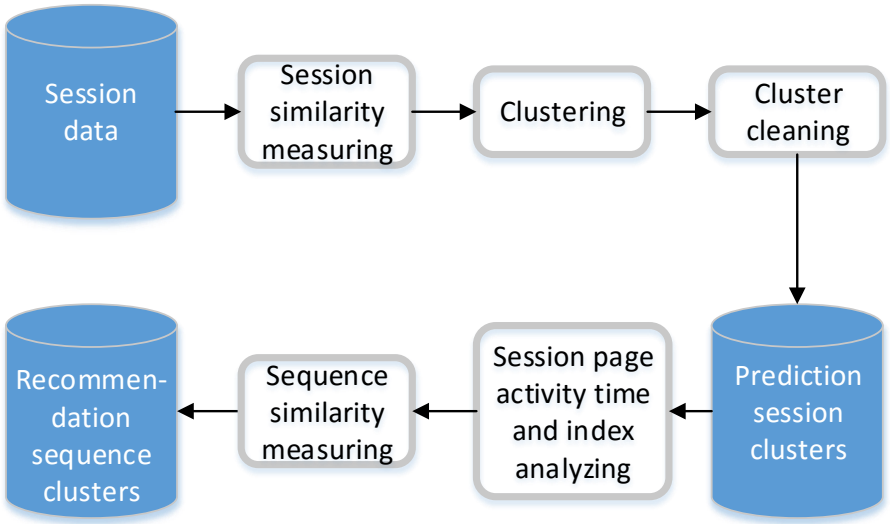


Figure 4.29: Visitor sessions grow into prediction session clusters, and prediction session clusters turn into recommendation sequence clusters.

Cost to adapt web site structure to recommender system

As the visitor navigation is supposed to be lead by web site structure, it may cost site owners or content providers if they require enhancing site structure to benefit visitors. Accordingly, a cost metric taking advantages of prediction and recommendation cluster should be considered. Since it is difficult to make an assumption about the specific structure of site or recommender system, an effort of estimation to reconstruct visitor tendency to the endorsement of the website is reasonable to study. Particularly, a conversion cost to migrate a prediction cluster to its corresponding recommendation cluster can be regarded as an initial solution. In detail, this conversion needs to be based on sub-conversion of each prediction sequence to correlative recommendation one. In addition, the cost of sub-conversion should be globally and locally computed through combination measure since the measure correctly reflects the similarity between them and the more similar sessions are, the less conversion cost is supposed to be. Namely, sequence conversion cost and similarity are in inverse ratio. For example, the two prediction and recommendation sequences of web access in Figure 4.31 have the similarity of 0.4 by combination measure. Consequently, their conversion cost will then be -0.4. This conversion cost is also appropriate to apply in web usability evaluation.

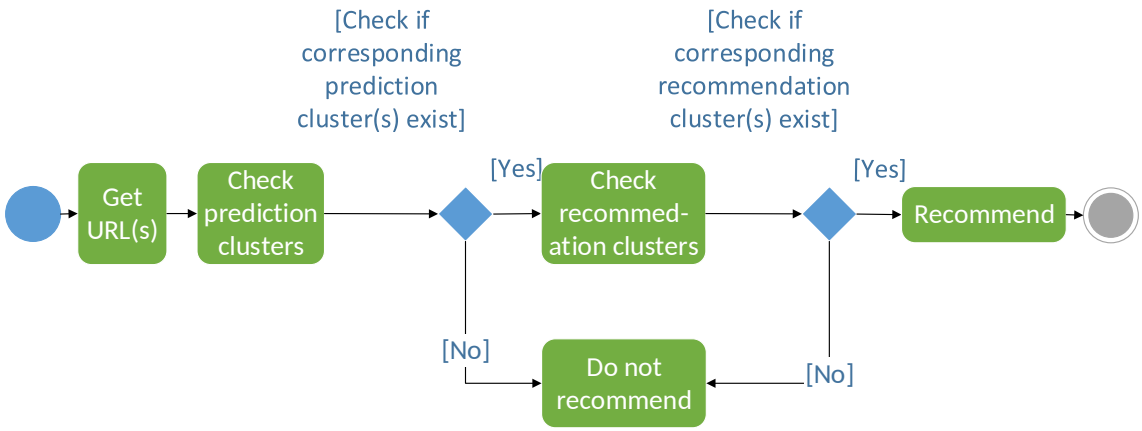


Figure 4.30: The representation of prediction and recommendation workflow.

$$A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$$
$$C \rightarrow D \rightarrow E \rightarrow A \rightarrow B$$

Figure 4.31: First prediction and recommendation sequences of clusters in Figure 4.26 and 4.27.

The adaptation of visitor manner to business strategy is more beneficial if it meets more end-users, hence cluster size plays an important role in conversion cost calculation. Concerning two equal prediction-to-recommendation conversion costs, which one consists of greater number of sessions is the preferred one. Alternatively, the one which costs less for conversion among same size pairs of prediction and recommendation clusters is most productive. The formula to compute total conversion cost, in order to switch prediction cluster sequences to recommendation ones, should take cluster size into consideration as follows:

$$C(P,R) = \frac{Sum(sim(i,j) \times -1)}{N} \tag{4.5}$$

where $C(P,R)$ is given as total conversion cost between clusters P(prediction) and R(recommendation), $sim(i,j)$ is the computed similarity score between 2 corresponding prediction and recommendation sequences i and j and N is the sequence number of each cluster.

As the total conversion cost between clusters in Figure 4.26 and 4.27 is -1.9 by combination measure, this cost should be divided by the number of sequences in the cluster to find the representative cost, as $-1.9/5 = -0.38$, in accordance with (3).

For the sake of optimization, that minimizes this cost but makes web site more convenient to visitors, pairs of clusters with lower total conversion cost should be in higher implementation priority to advance web site reconstruction.

Experimental result

The prediction accuracy performed on test sets should take into account both matching number of sessions and set size. *Nigam et al.* (2015) measured this accuracy as the ratio between correct prediction number and test session number. The formula is defined as follow:

$$Prediction\_Accuracy = \frac{Correct\_prediction\_number}{Test\_session\_number} \tag{4.6}$$

The experiments were conducted on three datasets of 2000 individual sessions each from a web site of University campus with more than 20,000 visits monthly. The three datasets were recorded at the different time in the month and day in order to have representative visits. Dataset collection contains information of sessionID, accessed URLs, activity time etc. of sessions was implemented by Beampulse company, which provides such services written in Java and Javascript. Each dataset is randomly split into train and test sets, with ratio 80% and 20% respectfully. Experimental results show the significant impact of hierarchical criteria on prediction clusters number and thus prediction accuracy. As previously named in (1) as *diff*, this parameter is based on intra and inter distances between created clusters. Consequently, it is a threshold to decide whether the merging process of sub-clusters should carry on in the dendrogram. The sooner this process stops, the more clusters are generated in the result and vice versa. Apparently, more clusters imply less elements contained in a cluster, that makes clusters more specific. This provokes fewer eliminations to make the cluster set pure, and thus more visit patterns remain. Although a lower *diff* value improves the accuracy of prediction besides time saving in bottom-up hierarchical clustering, it may trade off time cost of seeking and loading appropriate clusters for visitor access inputs. Figure 4.32 and 4.33 illustrate the experimental correlation between the number of clusters, prediction accuracy and the hierarchical parameter. As ones can see, the number of matches between test set and prediction clusters increases if there are more prediction clusters created (*i.e.* smaller values of hierarchical parameter). For

example, in Figure 4.32 the cluster number reaches 174 when *diff* = 0.1 in Dataset 1. Similarly, the prediction accuracy rises 27.5% with Dataset 2 at the same *diff* value, as presented in Figure 4.33. Also, in accordance with the experimental results, the number of output prediction clusters is consistently higher than correct prediction number at every hierarchical parameter. Furthermore, the prediction accuracy is impacted not only by prediction and recommendation nature but also the coverage of the train set to test set.

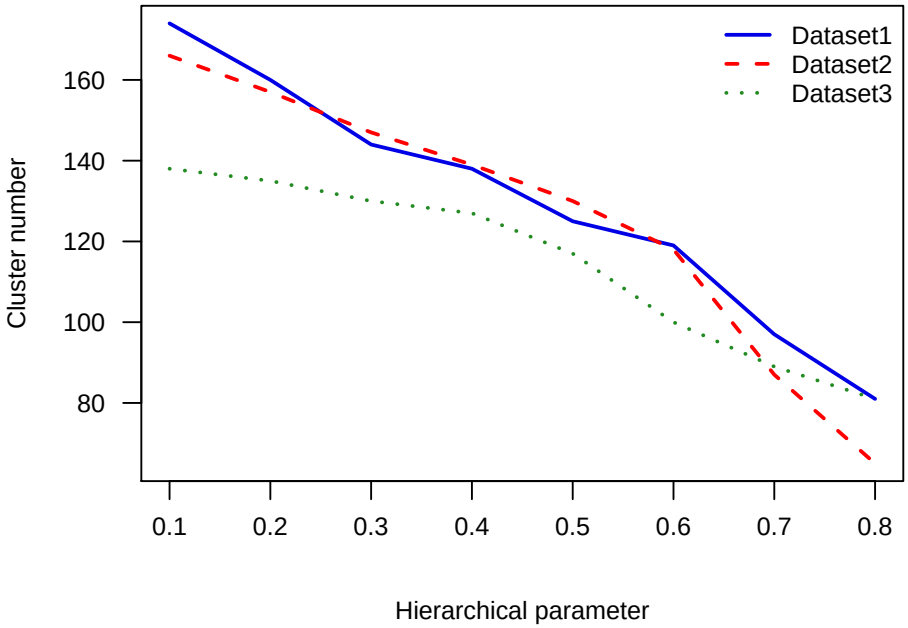


Figure 4.32: The hierarchical parameter is inversely proportional to the number of clusters.

In order to conduct the experiments of recommendation that take advantages of prediction experimental results, a recommender system should be implemented and tested under real user conditions. Accordingly, the recommendation performance can be measured by the ignorance of visitor on non-target pages and their activity time on target ones.

Related work

A considerable amount of mining techniques in web usage prediction and recommendation (WPR) has been proposed. *Anitha* (2010) proposed WPR model by integrating

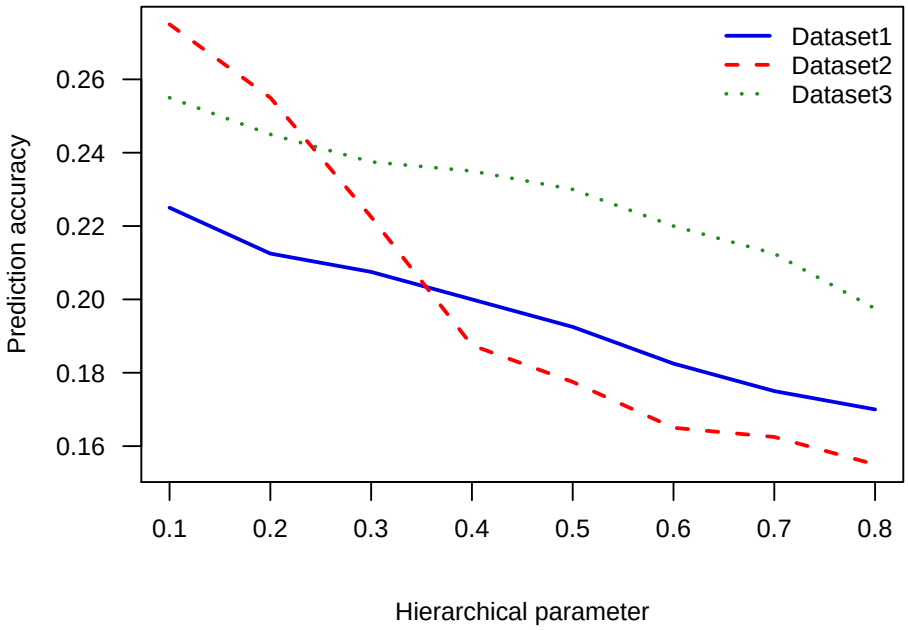


Figure 4.33: The hierarchical parameter is inversely proportional to the prediction accuracy.

pair-wise nearest neighbor clustering with support pruned in all k-th order Markov model. This approach is similar to (*Sejal et al.*, 2015), that takes advantage of Markov model in processing clusters of non-sequential sequences, but Markov trades prediction accuracy for space and time complexity as well as low coverage. Additionally, pair-wise nearest neighbour clustering may group sequences with common elements in different orders, and eliminates similarly ordered sequences with less common elements. Another related work by *Thwe* (2013) came up with Popularity and Similarity based Page Rank (PSPR) algorithm to solve vague result output by Markov model. The algorithm takes page properties such as frequency, duration, page size into account to rank the popularity of web pages. Nevertheless, duration may not reflect the interest of visitor in web page and page size can possibly be inaccurate at client size due to network traffic. A hybrid prediction model by *Liu et al.* (2014) recommends a combination of Markov and Hidden Markov although there are difficulties handling Hidden Markov such as time and space complexity, or small training set making initial model over-trained. *Su et al.* (2000) proposed a n-gram prediction referred to path-based model. However, they did not take short session sequences into consideration although these sequences can be part of their corresponding cluster. Applying only

the maximum occurred frequency of next click in WPR is another weakness of this method as it narrows down appropriate choices of visitor when browsing. One more disadvantage of n-gram process is to compromise between precision and applicability.

In (*Dimopoulos et al.*, 2010; *Chordia and Adhiya*, 2011), a prediction scheme using Web Access Sequence (WAS) clustering was presented, yet it was based on session similarity measure which is not effective in variable length sessions. *Wang and Shao* (2004) combined Jaccard index and k-medoids in the HBM clustering algorithm to group correspondence sessions before applying association rules. As this approach does not deal with web pages order in session when clustering, it may not result the optimal clusters for mining by association rules (*Lobo*, 2014). In (*Jalali et al.*, 2008), Jalali et al. worked with Longest Common Subsequence (LCS) to classify visitors' navigation pattern to forecast and serve their future requests. Nonetheless, LCS has the disadvantage of finding out the succession of common web pages of sessions which plays an important role in session similarity evaluation. For the reason of usability, accuracy and changeability, visitor-profile based suggestions in WPR like (*Espinosa et al.*, 2012; *Liu et al.*, 2014) that require visitors' input are impractical, since nowadays even cookies may not be available. Yet another shortcoming of this approach is the nature of prediction and recommendation frequently depends on previous requests of the same session, since even the same visitor may hit the site for particular objectives at different times. Alternatively, this type of model is more appropriate to social network or e-learning system. It is noticeable that most of the mentioned works are ambiguous between prediction and recommendation since they confusedly regard the most likely next access as visitor interest.

Conclusion

Our web usage prediction and recommendation (WPR) proposes a personalized modeling of web visitor navigation based on prediction and recommendation clusters. Concerning the improved similarity of sessions in prediction clusters, a sequence similarity combination measure was efficiently modified and applied. The experimental result also revealed the correlation between hierarchical clustering criteria and prediction accuracy. The recommendation process exploits prediction cluster to anticipate future behaviour of corresponding visitors, then considers activity time visitors spend

on and position of page in sessions to suggest them shortest path to the supposed desirable content. Besides, we discussed the adaptation cost estimation for converting behaviour patterns of visitors to well-defined rules of business, correspondingly to help client users save browsing time. In order to calculate this kind of estimation cost, we used the WPR result and the combination measure to consider the adaptation effort from website design to the proposition.

The results of our preliminary phase indicate an applicable process of WPR. In future work, besides current visit order and activity time, we want to enhance our WPR model by using other visitor behavioral features such as device, browser, operating system types, etc. Obviously, it is necessary to implement a recommender system on the idea and later find out the correctness of adaptation cost estimation. Additionally, the experiments should be extended to additional web sites, with more advanced usage pattern than the used one, to evaluate the WPR model efficiency. We are also interested in web visitor satisfactory measure to enhance the system performance. Last but not least, a performance comparison between our proposition and others is required to better highlight the method advantages.

Chapter 5

Conclusion

Sequence alignment techniques have been used widely in DNA sequences comparison, and have also been applied to segmentation of Web sessions. However, these techniques were not originally dedicated to web usage clustering, and there is room for optimization in order to adapt these alignments techniques to the specificities of real-time Web marketing, which is our field of application. We have made the choice of a combination of the well-known Needleman-Wunsch(NW) and Smith-Waterman(SW) global and local alignment techniques.

From an initial set of heuristic rules based on threshold-driven values which can be considered parameters of a given Web site, we improved it to present our contribution to event-sequence comparison, with a spe- cific focus on sequences of significantly different lengths. This new way of combin- ing global- and local-alignment techniques is based on the equal combination of both approaches. We experimentally evaluated this approach in the context of clustering web site visitors, based on the analysis of the browsing patterns. Under those set- tings, it was observed that our sequence similarity metric outperformed other related techniques. Our experiences show that our pairwise distance metric, based on the successive alignment of NW and SW in sequence pair, is a simple and realistic way to combine global and local approaches. The drawback of this work is the computational complexity which is NP-complete problem if sequence length and number are large.

Our web usage prediction and recommendation (WPR) proposes a personalized modeling of web visitor to convenience visitor browsing. After building prediction session clusters on visitor navigation using the combination measure, activity time they spend on and position of pages are concerned to recommend them appropriate web content. Also, we take advantages of the combination measure to calculate the

dissimilarity between prediction and recommendation sequences to preliminary calculate the cost to adapt the design of website to the proposition.

Contributions summary

Segmentation using hybrid alignment

A NW and SW combination rule sets are experimentally proved to be better than competitors such as hybrid measure and DTW in session similarity evaluation, especially with variable length sessions. This approach, which was featured by an equal consideration of global and local sequence similarity, implemented as an initial step to shape an upcoming effective session similarity measure.

This contribution has been validated by a workshop paper published at the 19th Pacific Asia Conference on Knowledge Discovery and Data Mining 2015 (PAKDD) (*Luu et al.*, 2015a).

Segmentation using glocal event alignment

At this stage of the investigation, a "glocal" measure which called combination was formed based on the achievement of PAKDD 2015 publication. Accordingly, we conducted experiments to assess the performance of our method and others. The experimental results on synthetic and real datasets clustering showed the advantage of combination measure compared to hybrid measure and DTW dealing with web sessions in different lengths.

This contribution has been validated by a paper published at the 12th International Conference on Machine Learning and Data Mining 2016 (MLDM) (*Luu et al.*, 2016a).

Web usage prediction and recommendation

The publication exploited the advantage of "glocal" measure in session clustering to build prediction session clusters. Thereafter, the recommendation for web visitors can be made by considering page activity time and index of prediction cluster sessions. Experiments were conducted to show the influence of clustering hierarchical strategy

upon prediction accuracy. In addition, the cost estimation method to adapt current web site to recommender system was initially discussed.

This contribution has been validated by a paper published at the 11th International Conference on Digital Information Management 2016 (ICDIM) (*Luu et al.*, 2016b).

Perspectives

The Internet user community including individuals, businesses and organizations has been developing rapidly. There has been also a broad diversity of this community with different circumstances, purposes and interests. As a result, web data provided by them is extremely large and still growing as a huge, dynamic but complex resource. Nevertheless, each web visitor can hold the attention to a very limited portion of this data. Thus, these visitors should be clustered into different groups for specific web contents. As one of influential methods for web user clustering, web usage mining has been in great interest. Generally, web usage mining can be considered to be an application of data mining to user data such as web log, recorded events, meta tags, robot.txt file etc., to uncover usage pattern, then better understand users' need. Most of the presented approaches focus on web sessions, sequence of the page visits, as the representative object of web user behavior to cluster. Accordingly, sessions within one cluster are similar to each other and dissimilar to sessions in other clusters.

In order to compare the similarity between web sessions to decide whether they should be in the same cluster, sequence alignment is one of efficient but yet to be widely implemented. In bioinformatics, sequence alignment has been popularly adopted to detect similar regions of DNA, RNA or protein sequences. As distance or similarity between two sequences can be generally revealed by necessary operation number to transform one into the other, the alignment is also applicable for non-biological sequences such as natural language strings or web sessions. In the context of web usage mining, there are gaps to bridge for adapting sequence alignment to the specificities of web session data. Concerning the correspondence of web usage behaviors, the global and local similarity of sessions must be taken into consideration. In addition, these kinds of similarity should be equally considered to eliminate the disadvantage when handling sessions with different lengths.

Obviously, an appropriate similarity measure leads to adequate clustering, regard-

less of partitional or hierarchical clustering. There are clustering applications to serve the browsing personalization, and prediction and recommendation is among them. For the purpose of web prediction, the point of clustering is, if a visitor interested in some particular order of page visiting, he is predicted to belong to cluster of others with similar web navigation. The prediction then frequently works as the fundamental part of web recommendation but not the nature of it, since the cluster itself and what is likely to bring to its users are different. The recommendation aims to conduct web visitor behavior to satisfy their needs or benefit the behavioral marketing, and there has been a wide variety of recommendation forms implemented. Besides, the cost to form the web site into a recommender system can somehow probably be estimated based on the distance between prediction and recommendation.

Bibliography

Algiriyage, N., S. Jayasena, and G. Dias (2015), Web user profiling using hierarchical clustering with improved similarity measure, in *Moratuwa Engineering Research Conference (MERCon), 2015*, pp. 295–300, IEEE.

Anitha, A. (2010), A new web usage mining approach for next page access prediction, *International Journal of Computer Applications*, *8*(11), 7–10.

Anupama, D., and S. D. Gowda (2015), Clustering of web user sessions to maintain occurrence of sequence in navigation pattern, *Procedia Computer Science*, *58*, 558–564.

Aruk, T., D. Ustek, and O. Kursun (2012), A comparative analysis of smith-waterman based partial alignment, in *Computers and Communications (ISCC), 2012 IEEE Symposium on*, pp. 000,250–000,252, IEEE.

Banerjee, A., and J. Ghosh (2001), Clickstream clustering using weighted longest common subsequences, in *Proceedings of the web mining workshop at the 1st SIAM conference on data mining*, vol. 143, p. 144, Citeseer.

Bose, R. J. C., and W. M. van der Aalst (2012), Process diagnostics using trace alignment: opportunities, issues, and challenges, *Information Systems*, *37*(2), 117–141.

Bouguessa, M. (2011), A practical approach for clustering transaction data, in *Machine Learning and Data Mining in Pattern Recognition*, pp. 265–279, Springer.

Brudno, M., S. Malde, A. Poliakov, C. B. Do, O. Couronne, I. Dubchak, and S. Batzoglou (2003), Glocal alignment: finding rearrangements during alignment, *Bioinformatics*, *19*(suppl 1), i54–i62.

Chakraborty, A., and S. Bandyopadhyay (2013a), Clustering of web sessions by fogsaa, in *Intelligent Computational Systems (RAICS), 2013 IEEE Recent Advances in*, pp. 282–287, IEEE.

Chakraborty, A., and S. Bandyopadhyay (2013b), Fogsaa: Fast optimal global sequence alignment algorithm, *Scientific reports*, *3*.

Chan, A. (2013), An analysis of pairwise sequence alignment algorithm complexities: Needleman-wunsch, smith-waterman, fasta, blast and gapped blast.

Chan, N. N., W. Gaaloul, and S. Tata (2012), A recommender system based on historical usage data for web service discovery, *Service Oriented Computing and Applications*, *6*(1), 51–63.

Chen, M. S., J. S. Park, and P. S. Yu (1996), Data mining for path traversal patterns in a web environment, in *Distributed Computing Systems, 1996., Proceedings of the 16th International Conference on*, pp. 385–392, IEEE.

Chitraa, V., and A.-S. Thanamni (2012), An enhanced clustering technique for web usage mining, in *International Journal of Engineering Research and Technology*, vol. 1, ESRSA Publications.

Chordia, B. S., and K. P. Adhiya (2011), Grouping web access sequences using sequence alignment method, *Indian Journal of Computer Science and Engineering (IJCSE)*, *2*(3), 308–314.

Claypool, M., P. Le, M. Wased, and D. Brown (2001), Implicit interest indicators, in *Proceedings of the 6th international conference on Intelligent user interfaces*, pp. 33–40, ACM.

Cooley, R., B. Mobasher, and J. Srivastava (1997), Grouping web page references into transactions for mining world wide web browsing patterns, in *Knowledge and Data Engineering Exchange Workshop, 1997. Proceedings*, pp. 2–9, IEEE.

Cooley, R., B. Mobasher, and J. Srivastava (1999), Data preparation for mining world wide web browsing patterns, *Knowledge and information systems*, *1*(1), 5–32.

Della Vedova, G. (2000), Multiple sequence alignment and phylogenetic reconstruction: Theory and methods in biological data analysis, Ph.D. thesis, Citeseer.

Dimopoulos, C., C. Makris, Y. Panagis, E. Theodoridis, and A. Tsakalidis (2010), A web page usage prediction scheme using sequence indexing and clustering techniques, *Data & Knowledge Engineering*, *69*(4), 371–382.

Doan, N.-Q., M. Ghesmoune, H. Azzag, and M. Lebbah (2015), Growing hierarchical trees for data stream clustering and visualization, in *2015 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE.

Duraiswamy, K., and V. V. Mayil (2008), Similarity matrix based session clustering by sequence alignment using dynamic programming, *Computer and Information Science*, *1*(3), p66.

Eddy, S. R. (2004), What is a hidden markov model?, *Nature biotechnology*, *22*(10), 1315–1316.

Edgar, R. C. (2004), Muscle: multiple sequence alignment with high accuracy and high throughput, *Nucleic acids research*, *32*(5), 1792–1797.

Espinosa, A., M. Regts, J. Tashiro, and M. Vargas-Martin (2012), Prediction model based on user profile and partial course progress for a digital media learning environment, in *The Fourth International Conference on Advances in Databases, Knowledge, and Data Applications*, pp. 120–123.

Giegerich, R., and D. Wheeler (1996), Pairwise sequence alignment, *BioComputing Hypertext Coursebook*, *2*.

Gündüz, Ş., and M. T. Özsu (2003), A web page prediction model based on click-stream tree representation of user behavior, in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 535–540, ACM.

Hair, J. F., W. C. Black, B. J. Babin, R. E. Anderson, R. L. Tatham, et al. (2006), *Multivariate data analysis*, vol. 6, Pearson Prentice Hall Upper Saddle River, NJ.

Hair Jr, J. F. (2007), Knowledge creation in marketing: the role of predictive analytics, *European Business Review*, *19*(4), 303–315.

Hay, B., G. Wets, and K. Vanhoof (2001), Clustering navigation patterns on a website using a sequence alignment method, *Intelligent Techniques for Web Personalization: IJCAI*, pp. 1–6.

Hay, B., G. Wets, and K. Vanhoof (2002), Web usage mining by means of multidimensional sequence alignment methods, in *WEBKDD 2002-Mining Web Data for Discovering Usage Patterns and Profiles*, pp. 50–65, Springer.

Hay, B., G. Wets, and K. Vanhoof (2004), Mining navigation patterns using a sequence alignment method, *Knowledge and Information Systems*, *6*(2), 150–163.

Henikoff, S., and J. G. Henikoff (1992), Amino acid substitution matrices from protein blocks, *Proceedings of the National Academy of Sciences*, *89*(22), 10,915–10,919.

Jalali, M., N. Mustapha, M. N. B. Sulaiman, and A. Mamat (2008), A web usage mining approach based on lcs algorithm in online predicting recommendation systems, in *Information Visualisation, 2008. IV'08. 12th International Conference*, pp. 302–307, IEEE.

Jalali, M., N. Mustapha, M. N. Sulaiman, and A. Mamat (2010), Webpum: A web-based recommendation system to predict user future movements, *Expert Systems with Applications*, *37*(9), 6201–6212.

Kim, J., D. W. Oard, and K. Romanik (2001), User modeling for information filtering based on implicit feedback.

Kohonen, T. (1985), Median strings, *Pattern Recognition Letters*, *3*(5), 309–313.

Kondrak, G. (2005), N-gram similarity and distance, in *String processing and information retrieval*, pp. 115–126, Springer.

Kumar, P., B. S. Raju, and P. R. Krishna (2011), A new similarity metric for sequential data, *Exploring Advances in Interdisciplinary Data Mining and Analytics: New Trends: New Trends*, p. 233.

Langfelder, P., B. Zhang, and S. Horvath (2008), Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r, *Bioinformatics*, *24*(5), 719–720.

Lee, W., B.-W. On, I. Lee, and J. Choi (2014), A big data management system for energy consumption prediction models, in *Digital Information Management (ICDIM), 2014 Ninth International Conference on*, pp. 156–161, IEEE.

Li, C. (2009), Research on web session clustering, *Journal of Software*, *4*(5), 460–468.

Li, C., and Y. Lu (2007), Similarity measurement of web sessions based on sequence alignment, *Wuhan University Journal of Natural Sciences*, *12*(5), 814–818.

Likic, V. (2008), The needleman-wunsch algorithm for sequence alignment, *Lecture given at the 7th Melbourne Bioinformatics Course, Bi021 Molecular Science and Biotechnology Institute, University of Melbourne*.

Liu, Q., Y. Wang, J. Li, Y. Jia, and Y. Ren (2014), Predicting user likes in online media based on conceptualized social network profiles, in *Web Technologies and Applications*, pp. 82–92, Springer.

Liu, Y., Z. Li, H. Xiong, X. Gao, and J. Wu (2010), Understanding of internal clustering validation measures, in *International Conference on Data Mining*, pp. 911–916, IEEE.

Liu, Y., Y. Hong, C.-Y. Lin, and C.-L. Hung (2015), Accelerating smith-waterman alignment for protein database search using frequency distance filtration scheme based on cpu-gpu collaborative system, *International journal of genomics*, *2015*.

Lobo, D. (2014), Association rules: Normalizing the lift, in *Digital Information Management (ICDIM), 2014 Ninth International Conference on*, pp. 151–155, IEEE.

Lu, L., M. Dunham, and Y. Meng (2005), Discovery of significant usage patterns from clusters of clickstream data, in *Proc. of WebKDD*, pp. 21–24, Citeseer.

Luu, V., M. Ripken, G. Forestier, F. Fondement, and P. Muller (2016a), Using glocal event alignment for comparing sequences of significantly different lengths, in *International Conference on Machine Learning and Data Mining in Pattern Recognition (MLDM)*, pp. 58–72.

Luu, V., M. Ripken, G. Forestier, F. Fondement, and P. Muller (2016b), Web usage prediction and recommendation using web session clustering, in *International Conference on Digital Information Management (ICDIM)*, IEEE.

Luu, V.-T., G. Forestier, F. Fondement, and P.-A. Muller (2015a), Web site audience segmentation using hybrid alignment techniques, in *Trends and Applications in Knowledge Discovery and Data Mining*, pp. 29–40, Springer.

Luu, V.-T., G. Forestier, F. Fondement, and P.-A. Muller (2015b), Web site audience segmentation using hybrid alignment techniques, in *Trends and Applications in Knowledge Discovery and Data Mining*, *Lecture Notes in Computer Science*, vol. 9441, pp. 29–40, Springer International Publishing.

Maetschke, S. R., K. S. Kassahn, J. A. Dunn, S.-P. Han, E. Z. Curley, K. J. Stacey, and M. A. Ragan (2010), A visual framework for sequence analysis using n-grams and spectral rearrangement, *Bioinformatics*, *26*(6), 737–744.

Mandal, O. P., and H. K. Azad (2014), Web access prediction model using clustering and artificial neural network, in *International Journal of Engineering Research and Technology*, vol. 3, ESRSA Publications.

Marascu, A., S. A. Khan, and T. Palpanas (2012), Scalable similarity matching in streaming time series, in *Advances in Knowledge Discovery and Data Mining*, pp. 218–230, Springer.

Meesrikamolkul, W., V. Niennattrakul, and C. A. Ratanamahatana (2012), Shape-based clustering for time series data, in *Advances in knowledge discovery and data mining*, pp. 530–541, Springer.

Milligan, G. W., and M. C. Cooper (1986), A study of the comparability of external criteria for hierarchical cluster analysis, *Multivariate Behavioral Research*, *21*(4), 441–458.

Mount, D. W. (2008), Comparison of the pam and blosum amino acid substitution matrices, *Cold Spring Harbor Protocols*, *2008*(6), pdb–ip59.

Mount, D. W. (2009), Using hidden markov models to align multiple sequences, *Cold Spring Harbor Protocols*, *2009*(7), pdb–top41.

Muhamad, F. N., R. Ahmad, S. M. Asi, and M. Murad (2015), Reducing the search space and time complexity of needleman-wunsch algorithm (global alignment) and smith-waterman algorithm (local alignment) for dna sequence alignment, *Jurnal Teknologi*, *77*(20).

Nakamura, A., and M. Kudo (2011), Packing alignment: alignment for sequences of various length events, in *Advances in Knowledge Discovery and Data Mining*, pp. 234–245, Springer.

Navarro, G. (2001), A guided tour to approximate string matching, *ACM computing surveys (CSUR)*, *33*(1), 31–88.

Needleman, S. B., and C. D. Wunsch (1970), A general method applicable to the search for similarities in the amino acid sequence of two proteins, *Journal of molecular biology*, *48*(3), 443–453.

Nielsen, J. (2011), How long do users stay on web pages, *useit. com: Jakob Nielsen's Website*.

Nigam, B., S. Tokekar, and S. Jain (2015), Evaluation of models for predicting user's next request in web usage mining, *international Journal on Cybernetics & informatics (UCi)*, *4*, 1–13.

Nkweteyim, D. L. (2005), A collaborative filtering approach to predict web pages of interest from navigation patterns of past users within an academic website, Ph.D. thesis, University of Pittsburgh.

Notredame, C., D. G. Higgins, and J. Heringa (2000), T-coffee: A novel method for fast and accurate multiple sequence alignment, *Journal of molecular biology*, *302*(1), 205–217.

Pandi, M., O. Kashefi, B. Minaei, et al. (2011), A novel similarity measure for sequence data, *Journal of Information Processing Systems*, *7*(3), 413–424.

Perner, P. (2014), A novel method for the interpretation of spectrometer signals based on delta-modulation and similarity determination, in *Advanced Information Networking and Applications (AINA), 2014 IEEE 28th International Conference on*, pp. 1154–1160, IEEE.

Petitjean, F., and P. Gançarski (2012), Summarizing a set of time series by averaging: From steiner sequence to compact multiple alignment, *Theoretical Computer Science*, *414*(1), 76–91.

Petitjean, F., G. Forestier, G. Webb, A. Nicholson, Y. Chen, and E. Keogh (2014a), Dynamic time warping averaging of time series allows faster and more accurate classification, in *IEEE International Conference on Data Mining*.

Petitjean, F., G. Forestier, G. Webb, A. E. Nicholson, Y. Chen, E. Keogh, et al. (2014b), Dynamic time warping averaging of time series allows faster and more accurate classification, in *International Conference on Data Mining*, pp. 470–479, IEEE.

Poornalatha, G., and S. R. Prakash (2013), Web sessions clustering using hybrid sequence alignment measure (hsam), *Social network analysis and mining*, *3*(2), 257–268.

Poornalatha, G., and P. Raghavendra (2011a), Alignment based similarity distance measure for better web sessions clustering, *Procedia Computer Science*, *5*, 450–457.

Poornalatha, G., and P. S. Raghavendra (2011b), Web user session clustering using modified k-means algorithm, in *Advances in Computing and Communications*, pp. 243–252, Springer.

Qi, Z., S. Redding, J. Y. Lee, B. Gibb, Y. Kwon, H. Niu, W. A. Gaines, P. Sung, and E. C. Greene (2015), Dna sequence alignment by microhomology sampling during homologous recombination, *Cell*, *160*(5), 856–869.

Rendón, E., I. Abundez, A. Arizmendi, and E. Quiroz (2011), Internal versus external cluster validation indexes, *International Journal of computers and communications*, *5*(1), 27–34.

Saleiro, P. (), Web sessions clustering for behavioral targeting.

Sejal, D., T. Kamalakant, V. Tejaswi, D. Anvekar, K. Venugopal, S. Iyengar, and L. Patnaik (2015), Wnpwr: Web navigation prediction framework for webpage recommendation, in *Recent Trends in Information Systems (ReTIS), 2015 IEEE 2nd International Conference on*, pp. 120–125, IEEE.

Shen, D., X. Wang, and H.-L. Chen (2008), Managing web-based learning resources for k-12 education: lessons learned from web analytics, in *World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, vol. 2008, pp. 470–475.

Si, J., Q. Li, T. Qian, and X. Deng (2012), Discovering *k* web user groups with specific aspect interests, in *Machine Learning and Data Mining in Pattern Recognition*, pp. 321–335, Springer.

Smith, T. F., and M. S. Waterman (1981), Identification of common molecular subsequences, *Journal of molecular biology*, *147*(1), 195–197.

Sonnhammer, E. L., and R. Durbin (1995), A dot-matrix program with dynamic threshold control suited for genomic dna and protein sequence analysis, *Gene*, *167*(1), GC1–GC10.

Soonsawad, P. (2013), Developing a new model for conversion rate optimization: A case study, *International Journal of Business and Management*, *8*(10), 41.

Srivastava, J., R. Cooley, M. Deshpande, and P.-N. Tan (2000), Web usage mining: Discovery and applications of usage patterns from web data, *ACM SIGKDD Explorations Newsletter*, *1*(2), 12–23.

Su, Z., Q. Yang, Y. Lu, and H. Zhang (2000), Whatnext: A prediction system for web requests using n-gram sequence models, in *Web Information Systems Engineering, 2000. Proceedings of the First International Conference on*, vol. 1, pp. 214–221, IEEE.

Thompson, J. D., T. Gibson, D. G. Higgins, et al. (2002), Multiple sequence alignment using clustalw and clustalx, *Current protocols in bioinformatics*, pp. 2–3.

Thwe, P. (2013), Proposed approach for web page access prediction using popularity and similarity based page rank algorithm, *International Journal of Science and Technology Research*, *2*(3).

Vorontsov, I. E., I. V. Kulakovskiy, and V. J. Makeev (2013), Jaccard index based similarity measure to compare transcription factor binding site models, *Algorithms for Molecular Biology*, *8*(1), 1.

Wang, F.-H., and H.-M. Shao (2004), Effective personalized recommendation based on time-framed navigation clustering and association mining, *Expert systems with applications*, *27*(3), 365–377.

Wang, W., and O. R. Zaïane (2002), Clustering web sessions by sequence alignment, in *Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on*, pp. 394–398, IEEE.

Xiao, J., and Y. Zhang (2001), Clustering of web users using session-based similarity measures, in *Computer Networks and Mobile Computing, 2001. Proceedings. 2001 International Conference on*, pp. 223–228, IEEE.

Yan, R., D. Xu, J. Yang, S. Walker, and Y. Zhang (2013), A comparative assessment and analysis of 20 representative sequence alignment methods for protein structure prediction, *Scientific reports*, *3*.

Zahid, S. K., L. Hasan, A. A. Khan, and S. Ullah (2015), A novel structure of the smith-waterman algorithm for efficient sequence alignment, in *International Conference on Digital Information, Networking, and Wireless Communications (DINWC)*, pp. 6–9, IEEE.

UNIVERSITÉ HAUTE-ALSACE

Vinh-Trung Luu

Using event sequences alignment
for automatic web users segmentation

Résumé en français

Une masse de données importante est collectée chaque jour par les gestionnaires de site internet sur les visiteurs qui accèdent à leurs services. La collecte de ces données a pour objectif de mieux comprendre les usages et d'acquérir des connaissances sur le comportement des visiteurs. A partir de ces connaissances, les gestionnaires de site peuvent décider de modifier leur site ou proposer aux visiteurs du contenu personnalisé. Cependant, le volume de données collectés ainsi que la complexité de représentation des interactions entre le visiteur et le site internet nécessitent le développement de nouveaux outils de fouille de données. Dans cette thèse, nous avons exploré l'utilisation des méthodes d'alignement de séquences pour l'extraction de connaissances sur l'utilisation de site Web (web mining). Ces méthodes sont la base du regroupement automatique d'internautes en segments, ce qui permet de découvrir des groupes de comportements similaires. De plus, nous avons également étudié comment ces groupes pouvaient servir à effectuer de la prédiction et la recommandation de pages. Ces thèmes sont particulièrement importants avec le développement très rapide du commerce en ligne qui produit un grand volume de données (big data) qu'il est impossible de traiter manuellement.

Mots clés : fouille de données web, classification, alignement de séquences

Résumé en anglais

This thesis explored the application of sequence alignment in web usage mining, including user clustering and web prediction and recommendation. This topic was chosen as the online business has rapidly developed and gathered a huge volume of information and the use of sequence alignment in the field is still limited. In this context, researchers are required to build up models that rely on sequence alignment methods and to empirically assess their relevance in user behavioral mining. This thesis presents a novel methodological point of view in the area and show applicable approaches in our quest to improve previous related work. Web usage behavior analysis has been central in a large number of investigations in order to maintain the relation between users and web services. Useful information extraction has been addressed by web content providers to understand users' need, so that their content can be correspondingly adapted. One of the promising approaches to reach this target is pattern discovery using clustering, which groups users who show similar behavioral characteristics. Our research goal is to perform users clustering, in real time, based on their session similarity.

Keywords: web mining, clustering, sequence alignment