On Measuring Similarity for Sequences of Itemsets

Elias Egho, Chedy Raïssi, Toon Calders, Nicolas Jay, Amedeo Napoli

On Measuring Similarity for Sequences of Itemsets

Elias Egho , Chedy Raïssi , Toon Calders , Nicolas Jay , Amedeo Napoli

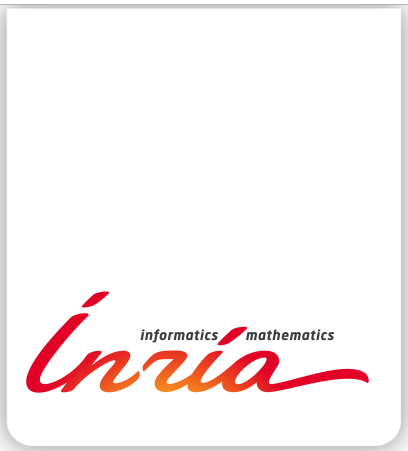
RESEARCH REPORT

N° 8086

October 2012

Project-Teams Orpailleur

ISSN 0249-6399 ISRN INRIA/RR--8086--FR+ENG



On Measuring Similarity for Sequences of Itemsets

Elias Egho *, Chedy Raïssi †, Toon Calders ‡, Nicolas Jay *, Amedeo Napoli *

Project-Teams Orpailleur

Research Report n° 8086 — October 2012 — 30 pages

Abstract: Computing the similarity between sequences is a very important challenge for many different data mining tasks. There is a plethora of similarity measures for sequences in the literature, most of them being designed for sequences of items. In this work, we study the problem of measuring the similarity between sequences of itemsets. We present new combinatorial results for efficiently counting distinct and common subsequences. These theoretical results are the cornerstone of an effective dynamic programming approach to deal with this problem. Experiments on healthcare trajectories and synthetic datasets, show that our measure of similarity produces competitive scores and indicates that our method is relevant for large scale sequential data analysis.

Key-words: Similarity measure, Clustering, Sequence mining

* LORIA, Vandoeuvre-les-Nancy, France
† INRIA, Nancy Grand Est, France
‡ Eindhoven University of Technology, Eindhoven, The Netherlands

RESEARCH CENTRE
NANCY – GRAND EST

615 rue du Jardin Botanique
CS20101
54603 Villers-lès-Nancy Cedex

Mesure de similarité pour les séquences de itemsets

Résumé : Le calcul de similarité entre les séquences est d'une extrême importance dans de nombreuses approches d'explorations de données. Il existe une multitude de mesures de similarités de séquences dans la littérature. La plupart de ces mesures sont conçues pour des séquences simples, dites séquences d'items. Dans ce travail, nous étudions le problème de similarité entre des séquences complexes (i.e., des séquences d'ensembles ou itemsets) d'un point de vue purement combinatoire. Nous présentons de nouveaux résultats afin de compter efficacement toutes les sous-séquences communes à deux séquences. Ces résultats théoriques sont la base d'une mesure de similarité calculée efficacement grâce à une approche de programmation dynamique. Des expériences réalisées et présentées sur des soins de santé et sur des jeux de données synthétiques, montrent que notre mesure de similarité produit des résultats intéressants et probants. Cette série d'expériences indique que notre mesure de similarité est pertinente pour les applications impliquant l'analyse de données séquentielles.

Mots-clés : Mesure de imilarité, Clustering, Feuille de données séquentielles

1 Introduction

Sequential data is widely present and used in many applications such as matching of time series in databases [1], DNA or amino-acids protein sequence analysis [2, 3], web log analysis [4], and music sequences matching [5]. Consequently, analyzing sequential data has become an important data mining and machine learning task with a special focus on the examination of pairwise relationships between sequences. For example, some clustering and kernel-based learning methods depend on computing distances or similarity scores between sequences [6, 7]. However, for a large part of literature, similarity measures on sequential data remains limited to *simple sequences*, which are ordered lists of items (i.e., symbols) [8, 9, 10, 11]. In contrast, in modern life sciences [12], sequential data sets are represented as ordered lists of itemsets (i.e., *sets* of symbols). This peculiarity is in itself a challenge as it implies to carefully take into account complex combinatorial aspects to compute similarities between sequences.

In this study, we focus on the notion of common subsequences as a means to define a distance or similarity score between a pair of sequences composed of a list of itemsets. The hypothesis that common subsequences can characterize similarity is not new. For instance, a very well known state-of-the-art algorithm: *Longest Common Subsequence* [13], uses the length of the longest common subsequence as a similarity measure between two sequences. However, and as clearly stated by H. Wang for simple sequences:*"This measure [...] ignores information contained in the second, third, ..., longest subsequences"* [11]. Additionally, this measure behaves erratically when the sequences contain itemsets. We motivate this claim by considering three sequences $S_1 = \langle\{c\}\{b\}\{a,b\}\{a,c\}\rangle$, $S_2 = \langle\{b\}\{c\}\{a,b\}\{a,c\}\rangle$ and $S_3 = \langle\{b,d\}\{a,b\}\{a,c\}\{d\}\rangle$. The longest common subsequence, denoted by LCS, between sequences S_1 and S_2 is $LCS(S_1,S_2) = \langle\{b\}\{a,b\}\{a,c\}\rangle$, and between S_1 and S_3 is $LCS(S_1,S_3) = \langle\{b\}\{a,b\}\{a,c\}\rangle$. This similarity measure is usually defined as $sim_{LCS}(S,T) = \frac{|LCS(S,T)|}{max(|S|,|T|)}$ and thus one may conclude that because $sim_{LCS}(S_1,S_2) = sim_{LCS}(S_1,S_3) = \frac{3}{4}$, then the sequence S_1 is equidistant from sequence S_2 and S_3. Clearly this is a wrong finding as S_1 is exactly the same sequence as S_2, but with a slight inversion of the two first itemsets. *How can one maximize the information used to compute a similarity measure between two sequences?* As for [11], we strongly believe that the *number of common subsequences* (and not only the length of the longest one) between two sequences is appealing in order to answer the previous question. We illustrate this intuition with the three previously considered sequences S_1, S_2 and S_3. Let $ACS(S,T)$ be the cardinality of the set that contains *all common subsequences* between S and T. $ACS(S_1,S_2) = 40, ACS(S_1,S_3) = 26$ and $ACS(S_2,S_3) = 26$. Based on this computation, it is trivial to conclude that sequences S_1 and S_2 share stronger affinity than with S_3 (a finding that was not detected by the longest common subsequence measure). To date, there *does not exist* any approach that computes efficiently ACS and use it as a basis for a similarity measure for complex sequences.

In this work, the main and significant contributions are summarized as follows:

Theoretical analysis We start by answering two fundamental theoretical open problems: (i) given a sequence of itemsets, can we count, *without enumerating*, the number of distinct subsequences? (ii) for a pair of sequences, can we *efficiently* count the number of common subsequences? We present two theorems that positively answer these questions.

Algorithmic and approximability results We discuss and present a dynamic programming algorithm for counting all common subsequences (ACS) between two given sequences. This dynamic programming algorithm allows us to define in a simple and intuitive manner our similarity measure which is a ratio between the number of common subsequences from two sequences S and T divided by the maximal number of distinct subsequences. As a

consequence of the huge size of the input sequences in some data sets, we present *two approximation techniques* to compute efficiently ACS: the first approach relies on approximating the size of a union of a family of sets in terms of the intersections of all subfamilies (i.e., *inclusion-exclusion principle*) based on the direct application of a result from Linial and Nisan [14]. The second approximation technique relies on limiting the depth of the backward sets computation between sequences. We discuss in details these approximations which are extremely efficient and useful on very long sequences.

Experiments and Evaluations We believe that the results reported in this work are a useful contribution with direct practical applications to different discriminative approaches, and in particular kernel methods, because new complex sequence kernels can be devised based on the theoretical results provided in this work. Moreover, the method is completely general in that it can be used (with slight modifications) for a broad spectrum of sequence-based classification or clustering problems. We report an extensive empirical study on synthetic datasets and a qualitative experiment with a dataset consisting of trajectories of cancer patients extracted from french healthcare organizations.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 briefly reviews the preliminaries needed in our development. Section 4 and 5 introduces our new combinatorial results. Two experimental studies are reported in Section 7 and we conclude our work in Section 8.

2 Related Work

In 1966, Levenshtein [8] proposed a measure to compute a distance between strings. Since then, many studies focused on developing efficient approaches for sequence similarities. The Levenshtein distance (or edit distance) between strings s and t is defined as the minimum number of edit operations needed to transform s into t. The edit operations are either an insertion, a deletion, or a substitution of a symbol. Many other approaches are built on this result but with notable differences like weighting the symbols and the edit operations [9], or using stochastic processes [15]. For time series, a well-known approach is the Dynamic Time Warping (DTW) technique for finding an optimal alignment between two series [10]. Intuitively, the sequences are warped in a nonlinear fashion to match each other. DTW had a huge impact and has been used to compare multiple patterns in automatic speech recognition to cope with different speaking speeds [16]. Zaki et al. [17] and Vlachos et al. [18] followed a radically different approach by developing longest common subsequences approaches for the comparison and similarity measure. However, the common information shared between two sequences is more than the longest common subsequence. In fact counting all possible common information between sequences provides a good idea about the similarity relationship between the sequences and their overall *complexity*. In addition, the common subsequences problem is related to the problem of counting the number of all distinct common subsequences between two sequences. Wang et al. [11] studied the usage of the count of all common subsequences (ACS) as a similarity measure between two sequences of items. Elzinga et al. [19] followed the same intuition and proposed a dynamic programming algorithm to count distinct common subsequences between two sequences of items.

In this work, we extend and generalize the previous works of [11, 19] for the complex structure of sequence of itemsets.

$\mathcal{D}_{ex} =$

S_1	$\langle\{a\}\{a,b\}\{e\}\{c,d\}\{b,d\}\rangle$
S_2	$\langle\{a\}\{b,c,d\}\{a,d\}\rangle$
S_3	$\langle\{a\}\{b,d\}\{c\}\{a,d\}\rangle$
S_4	$\langle\{a\}\{a,b,d\}\{a,b,c\}\{b,d\}\rangle$

Table 1: The sequence database used as the running example

3 Preliminaries

Definition 1 (Sequence)**.** *Let $\mathcal{I}$ be a finite set of* items*. An* itemset *X is a non-empty subset of $\mathcal{I}$. A* sequence *S over $\mathcal{I}$ is an ordered list $\langle X_1 \cdots X_n\rangle$, where X_i ($1 \leq i \leq n$, $n \in \mathbb{N}$) is an itemset. S^l denotes the l-prefix $\langle X_1 \ldots X_l\rangle$ of sequence S with $1 \leq l \leq n$. The j-th itemset X_j of sequence S is denoted $S[j]$ with $1 \leq j \leq n$.*

Definition 2 (Subsequence)**.** *A sequence $T = \langle Y_1 \cdots Y_m\rangle$ is a **subsequence** of $S = \langle X_1 \ldots X_n\rangle$, denoted by $T \preceq S$, if there exist indices $1 \leq i_1 < i_2 < \cdots < i_m \leq n$ such that $Y_j \subseteq X_{i_j}$ for all $j = 1 \ldots m$ and $m \leq n$. S is said to be a **supersequence** of T.*

*$\varphi(S)$ denotes the **set of all subsequences** of a given sequence S and $\phi(S) = |\varphi(S)|$. For two sequences S and T, $\varphi(S,T)$ denotes the set of **all common subsequences** between two sequences S and T: $\varphi(S,T) = \varphi(S) \cap \varphi(T)$ and $\phi(S,T) = |\varphi(S,T)|$.*

We now define the following similarity measure between two sequences of itemsets S and T.

Definition 3. *The **similarity between two sequences** S **and** T, denoted $sim(S,T)$ is defined as the number of common subsequences divided by the maximal number of subsequences of S and T; that is:*

$$sim(S,T) = \frac{\phi(S,T)}{\max\{\phi(S), \phi(T)\}}$$

From this point on, the rest of the paper, up to the experiments section, will be dedicated to devise efficient techniques for computing $\phi(S)$ and $\phi(S,T)$, as these form the backbone of our new similarity measure. As the explanation and the proofs of correctness of these computations involve complicated manipulations of sequences, we introduce the following operators on sets of sequences.

Definition 4 (Concatenation)**.** *Let $S = \langle X_1 \cdots X_n\rangle$ be a sequence, and Y be an itemset. The **concatenation of the itemset** Y **with the sequence** S, denoted $S \circ Y$, is the sequence $\langle X_1 \cdots X_n\ Y\rangle$.*

As usual, the powerset of an itemset Y will be denoted by $\mathcal{P}(Y)$, and $\mathcal{P}_{\geq 1}(Y)$ denotes all nonempty subsets of Y; that is, $\mathcal{P}_{\geq 1}(Y) = \mathcal{P}(Y) \setminus \{\emptyset\}$.

Example 1. *We use the sequence database $\mathcal{D}_{ex}$ in Table 1 as a running example. It contains 4 data sequences over the set of items $\mathcal{I} = \{a,b,c,d,e\}$. Sequence $\langle\{a\}\{b\}\{c,d\}\rangle$ is a subsequence of $S_1 = \langle\{a\}\{a,b\}\{e\}\{c,d\}\{b,d\}\rangle$. The 3-prefix of S_1, denoted S_1^3, is $\langle\{a\}\{a,b\}\{e\}\rangle$ and $S_1[2]$, the second itemset in sequence S_1, is $\{a,b\}$.*

The set of all subsequences of S_4^2 is

$\varphi(S_4^2) = \{\langle\rangle, \langle\{a\}\rangle, \langle\{b\}\rangle, \langle\{d\}\rangle, \langle\{a,b\}\rangle, \langle\{a,d\}\rangle, \langle\{b,d\}\rangle, \langle\{a,b,d\}\rangle, \langle\{a\}\{a\}\rangle, \langle\{a\}\{b\}\rangle, \langle\{a\}\{d\}\rangle, \langle\{a\}\{a,b\}\rangle, \langle\{a\}\{a,d\}\rangle, \langle\{a\}\{b,d\}\rangle, \langle\{a\}\{a,b,d\}\rangle\}$

Hence, $\phi(S_4^2) = 15$.

The concatenation of the sequence S_4^2 *with the itemset* $\{a, b, c\}$*, denoted as* $S_4^2 \circ \{a, b, c\}$*, is the sequence* $\langle\{a\}\{a, b, d\}\{a, b, c\}\rangle$.

In addition,the set of all common subsequences of S_1^4 *and* S_2^3 *is*

$\varphi(S_1^4, S_2^3) = \{\langle\rangle, \langle\{a\}\rangle, \langle\{b\}\rangle, \langle\{d\}\rangle, \langle\{c\}\rangle, \langle\{c, d\}\rangle, \langle\{a\}\{a\}\rangle, \langle\{a\}\{b\}\rangle, \langle\{a\}\{c\}\rangle, \langle\{a\}\{d\}\rangle, \langle\{a\}\{c, d\}\rangle, \langle\{b\}\{d\}\rangle, \langle\{a\}\{b\}\{d\}\rangle\}$.

The similarity between S_1^4 *and* S_2^3 *is*

$$sim(S_1^4, S_2^3) = \frac{\phi(S_1^4, S_2^3)}{\max\{\phi(S_1^4), \phi(S_2^3)\}} = \frac{13}{\max\{56, 61\}} = \frac{13}{61} = 0.21$$

4 Counting All Distinct Subsequences

In this section, we present an efficient technique *to count* the number $\phi(S)$ of all distinct subsequences for a given sequence S. We emphasize the fact that the studied sequences are not *simple sequences* that are discussed in length in the bio-informatics literature for which efficient approaches exist, but rather an ordered list of itemsets. As we will show, this is a highly non-trivial extension as it implies taking into account non-trivial combinatorial aspects. Before stating the main result, we present the intuition behind the proposed counting scheme. Suppose that we extend a given sequence $S = \langle X_1 \cdots X_n\rangle$ with an itemset Y and we observe the relation between $\phi(S)$ and $\phi(S \circ Y)$. Two cases may appear:

1. Y is disjoint with any itemset in S; i.e., for all $i = 1 \ldots n$, $Y \cap S[i] = \emptyset$, then the number of distinct subsequences of $S \circ Y$ equals $|\varphi(S)| \cdot 2^{|Y|}$, since for all $T \in \phi(S)$ and $Y' \in \mathcal{P}_{\geq 1}(Y)$, $T \circ Y'$ is not in $\phi(S)$. For example, $\phi(\langle\{a, b\}\{c\}\rangle \circ \{d, e\}) = 8 \cdot 2^2 = 32$.

2. At least one item of Y appears in an itemset of S; i.e., $\exists i \in [1, n] : Y \cap S[i] \neq \emptyset$. In this case, $|\varphi(S \circ X)|$ is smaller than $|\varphi(S)| \cdot 2^{|Y|}$, because not every combination of a sequence in $\varphi(S)$ with an element from the power set of Y results in a unique subsequence. For example, if $S = \langle\{a, b\}\rangle$ and $Y = \{a, b\}$, the set of all subsequences of S is $\varphi(S) = \{\langle\rangle, \langle\{a\}\rangle, \langle\{b\}\rangle, \langle\{a, b\}\rangle\}$ and the power set of Y is $\mathcal{P}(Y) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$. The sequence $\langle\{a\}\rangle$ can be obtained by either extending the empty sequence $\langle\rangle \in \varphi(S)$ with the itemset $\{a\} \in \mathcal{P}(Y)$, or by extending $\langle\{a\}\rangle \in \varphi(S)$ with $\emptyset \in \mathcal{P}(Y)$.

   Therefore, we need to define a method to remove the repetitions from the count. Formally, $|\varphi(S \circ Y)| = |\varphi(S)| \cdot 2^{|Y|} - R(S, Y)$ where $R(S, Y)$ represents a *correction term* that equals the number of repetitions of subsequences that should be suppressed for a given S concatenated with the itemset Y.

We illustrate the second case with an example.

Example 2. *Consider sequence* S_4 *from our toy data set.* $S_4^2 = \langle\{a\}\{a, b, d\}\rangle$ *is the 2-prefix of* S_4*. Recall from Example 1 that the total number of subsequences of* S_4^2 *is* $\phi(S_4^2) = 15$*. Now suppose that we extend this sequence* S_4^2 *with the itemset* $Y = \{a, b, c\}$*. Clearly, concatenating each sequence from* $\varphi(S_4^2)$ *with each element in the power set of* $\{a, b, c\}$ *will generate some subsequences multiple times. For instance, the subsequence* $\langle\{a\}\{b\}\rangle$ *is generated twice:* $\langle\{a\}\rangle \circ \{b\}$ *and* $\langle\{a\}\{b\}\rangle \circ \emptyset$*. The same applies to other subsequences* $\langle\{a\}\rangle$*,* $\langle\{b\}\rangle$*,* $\langle\{a, b\}\rangle$*,* $\langle\{a\}\{a\}\rangle$ *and* $\langle\{a\}\{ab\}\rangle$*. Thus, making a total of* 6 *subsequences that are counted twice. In this case, the correct number of distinct subsequences for* $S_4^2 \circ Y = \langle\{a\}\{a, b, d\}\{a, b, c\}\rangle$ *is* $|\varphi(S_4^2)| \cdot 2^{|Y|} - R(S_4^2, Y) = 15 \cdot 2^3 - 6 = 114$.

As illustrated by the above example, the actual challenge is the computation of the value of the *correction term* $R(S,Y)$. The general idea is to compensate the repeated concatenation of subsequences from S by the power set of Y. The problem occurs with sequences in $\varphi(S)\circ\mathcal{P}_{\geq 1}(Y)$ that are already in $\varphi(S)$. Suppose T is such a sequence, then T must be decomposable as $T'\circ Y'$, where $T'\in\varphi(S^i)$ for some $i=0\ldots n-1$, and $Y'\subseteq Y\cap S[j]$, for some $j\in i+1\ldots n$. The following definition introduces the *position set* that will capture those positions in S that generate duplicates when compensating for such a sequence T we will consider the last i only such that T' in $\phi(S)$.

Definition 5 (Position set)**.** *Given an itemset Y and a sequence $S=\langle X_1\cdots X_n\rangle$, $L(S,Y)$ is the set of all **maximal positions** where the itemset Y has a maximal intersection with the different itemsets $S[i]$, $i=1\ldots n$. Formally,*

$$L(S,Y)=\{i\mid Y\cap S[i]\neq\emptyset,\ \textit{and}\ i=\max\{j\mid Y\cap S[i]\subseteq Y\cap S[j]\}\}.$$

Notice that if there are multiple positions that generate the same duplicates, we only consider the last one.

Example 3. *Let $S_4=\langle\{a\}\{a,b,d\}\{a,b,c\}\{b,d\}\rangle$ be the studied sequence.*
$L(\langle\rangle,\{a\})=\emptyset,\ L(\langle\{a\}\rangle,\{a,b,d\})=\{1\},\ L(\langle\{a\}\{a,b,d\}\rangle,\{a,b,c\})=\{2\}$,
$L(\langle\{a\}\{a,b,d\}\{a,b,c\}\rangle,\{b,d\})=\{2,3\}$.

The following lemma now formalizes the observation that we only need to consider the sets $S[i]$ for i in the position set.

Lemma 1. *Let S be a sequence, and Y an itemset. Then $\phi(S\circ Y)=\phi(S)\cdot 2^{|Y|}-R(S,Y)$, with*

$$R(S,Y)\ =\ \left|\bigcup_{\ell\in L}\left\{\varphi(S^{\ell-1})\circ\mathcal{P}_{\geq 1}(S[\ell]\cap Y)\right\}\right|$$

Notice, however, that the sets $\varphi(S^{\ell-1})\circ\mathcal{P}_{\geq 1}(S[\ell]\cap Y)$ are not necessarily disjoint; consider, e.g., $S=\langle\{a,b\}\{b,c\}\rangle$ and $Y=\{a,b,c\}$. Then $L=\{1,2\}$, and $\langle\{b\}\rangle$ appears in both $\varphi(S^0)\circ\mathcal{P}_{\geq 1}(S[1]\cap Y)$ and $\varphi(S^1)\circ\mathcal{P}_{\geq 1}(S[2]\cap Y)$. To incorporate this overlap, we compute the cardinality of the union in Lemma 1 using the inclusion-exclusion principle, leading to the following theorem:

Theorem 1. *Let $S=\langle X_1\ldots X_n\rangle$ and Y an itemset. Then,*

$$\phi(S\circ Y)=2^{|Y|}\cdot\phi(S)-R(S,Y) \tag{1}$$

with

$$R(S,Y)=\sum_{K\subseteq L(S,Y)}(-1)^{|K|+1}\left(\phi(S^{\min(K)-1})\cdot\left(2^{|(\bigcap_{j\in K}S[j])\cap Y|}-1\right)\right) \tag{2}$$

Proof. See Appendix. □

We illustrate the counting process with sequence S_4^3. The position set of this sequence is given in Example 3.

$$\begin{aligned}
\phi(\langle\rangle) &= 1\\
\phi(\langle\{a\}\rangle) &= 2^{|\{a\}|}\cdot\phi(\langle\rangle)\ =\ 2\\
\phi(\langle\{a\}\{a,b,d\}\rangle) &= 2^{|\{a,b,d\}|}\phi(\langle\{a\}\rangle)-(2^{|\{a,b,d\}\cap\{a\}|}-1)\cdot\phi(\langle\rangle)\\
&= 2^3\cdot 2-(2^1-1)\cdot 1=15\\
\phi(\langle\{a\}\{a,b,d\}\{a,b,c\}\rangle)&\\
&= 2^{|\{a,b,c\}|}\cdot\phi(\langle\{a\}\{a,b,d\}\rangle)-(2^{|\{a,b,d\}\cap\{a,b,c\}|}-1)\cdot\phi(\langle\{a\}\rangle)\\
&= 2^3\cdot 15-(2^2-1)\cdot 2=114
\end{aligned}$$

5 Counting All Common Subsequences

In this section, we will extend the previous results to count all common distinct subsequences between two sequences S and T. Again, we discuss the basic intuition and then present the main result. Suppose that we extend the sequence S with an itemset Y and we observe the relation between $\varphi(S,T)$ and $\varphi(S \circ Y, T)$, two cases may appear:

1. If no items in Y appear in any itemset of S and T then the concatenation of the itemset Y with the sequence S *has no effect* on the the set $\varphi(S,T)$.

2. If at least an item in Y appears in either one of the sequences S or T (or both) then it can be observed that new common subsequences may appear in $\varphi(S,T)$. As for the counting method of the distinct subsequences of a unique sequence S, repetitions may occur and a generalized correction term for both S and T needs to be defined. Formally,

$$|\varphi(S \circ Y, T)| = |\varphi(S,T)| + A(S,T,Y) - R(S,T,Y)$$

where $A(S,T,Y)$ represents the number of extra common subsequences that should be added and $R(S,T,Y)$ is the correction term.

Similarly to the distinct subsequence problem, the position set will index the positions that generate duplicate sequences. The following lemma formalizes this observation:

Lemma 2. *Let $S = \langle X_1 \dots X_n \rangle$, $T = \langle X'_1 \dots X'_m \rangle$ and Y an itemset.*

$$A(S,T,Y) = \left| \bigcup_{\ell \in L(T,Y)} \left\{ \varphi(S, T^{\ell-1}) \circ \mathcal{P}_{\geq 1}(T[\ell] \cap Y) \right\} \right|$$

$$R(S,T,Y) = \left| \bigcup_{\ell \in L(S,Y)} \left\{ \bigcup_{\ell' \in L(T,Y)} \left\{ \varphi(S^{\ell-1}, T^{\ell'-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap T[\ell'] \cap Y) \right\} \right\} \right|$$

Example 4. *Consider the sequences S_1 and S_2 from our running example. Let $S_1^4 = \langle \{a\}\{a,b\}\{e\}\{c,d\} \rangle$ be the 4-prefix of S_1, and let $S_2^3 = \langle \{a\}\{b,c,d\}\{a,d\} \rangle$ be the 3-prefix of S_2*

Suppose that we extend S_1^4 with the itemset $Y = \{b,d\}$ and count all distinct common subsequences between $S_1^4 \circ \{b,d\}$ and S_2^3.

Notice that the itemset $\{b,d\}$ appears two times in the sequence S_2^3: in the itemsets $\{b,c,d\}$ and $\{a,d\}$. Thus, $L(S_2^3, \{b,d\}) = \{2,3\}$ and $A(S_1^4, S_2^3, \{b,d\}) = |\{\varphi(S_1^4, S_2^1) \circ \mathcal{P}_{\geq 1}(\{b,d\} \cap \{b,c,d\})\} \cup \{\varphi(S_1^4, S_2^2) \circ \mathcal{P}_{\geq 1}(\{b,d\} \cap \{a,d\})\}| = 14$.

Notice also that the itemset $\{b,d\}$ appears two times in S_1^4, in the itemsets $\{a,b\}$ and $\{c,d\}$. Thus $L(S_1^4, \{b,d\}) = \{2,4\}$. In this case, adding the values $A(S_1^4, S_2^3, \{b,d\})$ to $\phi(S_1^4, S_2^3)$ will over count some subsequences. For instance, the subsequences $\langle \{a\}\{b\}\{d\} \rangle$ and $\langle \{b\}\{d\} \rangle$ are counted twice: once in $\varphi(S_1^4, S_2^3)$ and the other when all sequences of the set $\varphi(S_1^4, S_2^3)$ are extended with $\{b,d\} \cap \{a,d\}$. The same remark applies to other subsequences: $\langle \{b\} \rangle$,$\langle \{d\} \rangle$, $\langle \{a\}\{b\} \rangle$ and $\langle \{a\}\{d\} \rangle$. In this case, the correct number of all common distinct subsequences between $S_1^4 \circ \{b,d\}$ and S_2^3 is $|\varphi(S_1^4, S_2^2)| + A(S_1^4, S_2^3, \{b,d\}) - R(S_1^4, S_2^3, \{b,d\})$ where:

$$\begin{aligned} R(S_1^4, S_2^3, \{b,d\}) &= | \quad \{\varphi(S_1^1, S_2^1) \circ \mathcal{P}_{\geq 1}(\{a,b\} \cap \{b,c,d\} \cap \{b,d\})\} \\ &\quad \cup \{\varphi(S_1^1, S_2^2) \circ \mathcal{P}_{\geq 1}(\{a,b\} \cap \{a,d\} \cap \{b,d\})\} \\ &\quad \cup \{\varphi(S_1^3, S_2^1) \circ \mathcal{P}_{\geq 1}(\{c,d\} \cap \{b,c,d\} \cap \{b,d\})\} \\ &\quad \cup \{\varphi(S_1^3, S_2^2) \circ \mathcal{P}_{\geq 1}(\{c,d\} \cap \{a,d\} \cap \{b,d\})\}| \\ &= \quad 6 \end{aligned}$$

Thus,

$$\begin{aligned} \phi(S_1^4 \circ \{b,d\}, S_2^3) &= |\varphi(S_1^4, S_2^2)| + A(S_1^4, S_2^3, \{b,d\}) - R(S_1^4, S_2^3, \{b,d\}) \\ &= 13 + 14 - 6 = 21 \end{aligned}$$

Similarly to Lemma 1 and as illustrated in the above example, the computation of the cardinality of the unions in Lemma 2 implies the usage of the inclusion-exclusion principle. This remark leads to the second theorem:

Theorem 2. *Let $S = \langle X_1 \dots X_n \rangle$, $T = \langle X'_1 \dots X'_m \rangle$ and Y an itemset. Then,*

$$\phi(S \circ Y, T) = \phi(S, T) + A(S, T, Y) - R(S, T, Y) \tag{3}$$

with

$$A(S,T,Y) = \sum_{K \subseteq L(T,Y)} (-1)^{|K|+1} \left(\phi(S, T^{\min(K)-1}) \cdot \left(2^{|(\bigcap_{j \in K} T[j]) \cap Y|} - 1 \right) \right) \tag{4}$$

and

$$R(S,T,Y) = \sum_{K \subseteq L(S,Y)} (-1)^{|K|+1} \left(\sum_{K' \subseteq L(T,Y)} (-1)^{|K'|+1} \cdot f(K, K') \right) \tag{5}$$

where:

$$f(K,K') = \phi(S^{\min(K)-1}, T^{\min(K')-1}) \cdot \left(2^{|(\bigcap_{j \in K} S[j]) \cap \left(\bigcap_{j' \in K'} T[j']\right) \cap Y|} - 1 \right)$$

Proof. See Appendix. □

5.1 Dynamic Programming

Theorem 2 implies a simple dynamic programming algorithm. For two given sequences S and T, such that $|S| = n$ and $|T| = m$, the program produces a $n \times m$ matrix, denoted $\mathcal{M}$, where the $\mathcal{M}_{i,j}$ cell corresponds to all common subsequences between S^i and T^j, $\mathcal{M}_{i,j} = \phi(S^i, T^j)$.

Example 5. *Consider the two sequences $S_1 = \langle \{a\}\{a,b\}\{e\}\{c,d\}\{b,d\} \rangle$ and $S_2 = \langle \{a\}\{b,c,d\}\{a,d\} \rangle$.*
$\phi(S_1, S_2) = 21$ and the set of all common subsequences of S_1 and S_2 is:
$\varphi(S_1, S_2) = \{ \langle\rangle, \langle\{a\}\rangle, \langle\{b\}\rangle, \langle\{c\}\rangle, \langle\{d\}\rangle, \langle\{c,d\}\rangle, \langle\{b\}\{d\}\rangle, \langle\{b,d\}\rangle, \langle\{a\}\{a\}\rangle, \langle\{a\}\{b\}\rangle, \langle\{a\}\{c\}\rangle,$
$\langle\{a\}\{d\}\rangle, \langle\{a\}\{c,d\}\rangle, \langle\{a\}\{b,d\}\rangle, \langle\{c,d\}\{d\}\rangle, \langle\{a\}\{d\}\{d\}\rangle, \langle\{d\}\{d\}\rangle, \langle\{c\}\{d\}\rangle, \langle\{a\}\{b\}\{d\}\rangle,$
$\langle\{a\}\{c\}\{d\}\rangle, \langle\{a\}\{cd\}\{d\}\rangle\}$

	$\emptyset$	$\{a\}$	$\{b,c,d\}$	$\{a,d\}$
$\emptyset$	1	1	1	1
$\{a\}$	1	2	2	2
$\{a,b\}$	1	2	4	5
$\{e\}$	1	2	4	5
$\{c,d\}$	1	2	10	13
$\{b,d\}$	1	2	12	21

Table 2: Matrix for counting all common subsequences between S_1 and S_2

We detail the computation of the cell $\mathcal{M}_{2,1}$ *with the position set* $L(S_2^1, \{a,b\}) = \{1\}$ *and* $L(S_1^1, \{a,b\}) = \{1\}$*:*

$$\begin{aligned}
\mathcal{M}(\{a,b\},\{a\}) &= \phi(\langle\{a\}\{a,b\}\rangle, \langle\{a\}\rangle) \\
&= \mathcal{M}(\{a\},\{a\}) \\
&\quad +(2^{|\{a\}\cap\{a,b\}|}-1)\cdot\mathcal{M}(\{a\},\{\emptyset\}) \\
&\quad -(2^{|\{a\}\cap\{a\}\cap\{a,b\}|}-1)\cdot\mathcal{M}(\{\emptyset\},\{\emptyset\}) \\
&= 2+1-1=2
\end{aligned}$$

The entire computation for $\phi(S_1, S_2)$ *is illustrated in Table 2.*

6 Complexity and Approximability Results

6.1 Complexity

We will now discuss the complexity of computing the number of subsequences in a sequence of itemets and the number of common subsequences in two such sequences using the formulas in Theorems 1 and 2. Essential in this analysis is the size of the position set $L(S,Y)$, which will highly depend on the specific case. It is important to notice that the size of $L(S,Y)$ is bounded by both $2^{|Y|}$ (every index corresponds to a unique subset of Y) and $|S|$ (every index corresponds to a unique position within S). Notice incidentally that the worst case $|L(S^{\ell-1}, S[\ell])| = \ell - 1$ is unlikely to happen for long sequences, as this implies that if we construct the following sequence: $S[1] \cap S[\ell], \ldots, S[\ell-1] \cap S[\ell]$, none of the entries in the sequence is followed by a superset. This would only happen in pathological cases such as $\langle\{a,b,c\}\{a,b,d\}\{a,c,d\}\{b,c,d\}\{a,b\}\{a,c\}\{a,d\}\{b,c\}\{b,d\}\ \{c,d\}\{a\}\{b\}\{c\}\{d\}\{a,b,c,d\}\rangle$. In this case $L(S^{\ell-1}, S[\ell]) = \{1,2,3,4,5,6,7,8,\ 9,10,11,12,13,14,15\}$.

First, we analyze the complexity of the brute-force method consisting of generating all subsequences followed by elimination of duplicates. For a sequence $\langle S_1 ... S_\ell\rangle$, there are $N = \Pi_{i=1}^{\ell} 2^{S[i]}$ subsequences we need to consider; for every position $i = 1 \ldots \ell$, any subset of S_i needs to be considered in combination with every subset of the other positions. Eliminating the duplicates can be done in time $N\log(N)$. The total complexity is hence $\mathcal{O}(N\log(N))$. If we take $C_\ell = \max_{i=1\ldots\ell}|S_i|$, N is upper bounded by 2^{C_ℓ}, and the total complexity of the brute-force method for finding all subsequences of S is bounded by $\mathbf{C}_\ell \mathbf{2}^{\mathbf{C}_\ell}$. When computing the number of subsequences that two sequences S and T have in common, first their respective subsequences are listed, deduplicated, and compared. The complexity of these operations comes down to

$\mathcal{O}(N\log(N) + M\log(M))$, where N is as above, and M is the similar quantity for T. We will show that our methods significantly improve upon these brute-force bounds.

Let's now analyze the complexity of the computation bsased upon the formula in Theorem 2. Suppose that we need to compute the number of subsequences of $\langle S_1...S_\ell\rangle$. Assume that we know the number of subsequences all $S^k, k < \ell$. The number of computations we need to perform if we apply the formula of Theorem 1, to get the number of subsequences of $\langle S_1...S_k\rangle$ is proportional to the size of the powerset of $L(S^{\ell-1}, S[\ell])$; indeed, we need to compute a sum over all subsets of $L(S^{\ell-1}, S[\ell])$. It is easy to see that the size of this set $L(S^{\ell-1}, S[\ell])$ is bounded by $\min(\ell-1, 2^{|S[\ell]|})$. Hence, the total complexity is bounded by $\sum_{k=1}^{\ell} 2^{\min(k-1, 2^{|S[k]|})}$. Hence, $\sum_k = 1^\ell 2^{k-1} = 2^\ell - 1 = \mathcal{O}(2^\ell)$ is an upper bound, which is significantly better than the brute-force method listing all subsequences and removing the duplicates, which has complexity $\Pi_{i=1}^{\ell} 2^{S[i]}$.

The complexity for the computation of the number of common subsequences in Theorem 2 goes along the same lines. Again we will first assume that for two sequences S and T, the number of common subsequences $\phi(S^i, T^j)$ have been computed for all S^i and T^j with (i, j) smaller than $(|S|, |T|)$. Let Y be the last itemset of S; that is, $Y = S[|S|]$, and $S' = S^{|S|-1}$. The main complexity term in the formula in Theorem 2 is in $R(S', T, S[|S|])$; this term dominates the complete expression in terms of computational complexity. The complexity of the double sum is proportional to $2^{|L(S',Y)|}2^{|L(T,Y)|}$, which is bounded by $2^{\min(|S'|, 2^{|Y|})}2^{\min(|T|, 2^{|Y|})} < 2^{|S'||T|}$. So, the total complexity, taking into account that we need to compute the number of common subsequences for all subsequences of S and T as well (cfr. the dynamic programming approach given in 5.1, leads to a total complexity of $\sum_{i=1}^{|S|}\sum_{j=1}^{|T|} 2^{ij} = \mathcal{O}(\min(|S|, |T|)2^{|S||T|})$.

6.2 Approximability results

As stated by Linial and Nissan in [14]: *"Many computational problems may be viewed as asking for the size of a union of a collection of sets. On some instances it turns out that while computing the size of the union is rather difficult, computing the sizes of members in the family, or even of arbitrary intersections thereof is easy. In these cases, the inclusion-exclusion formula may be used to find the size of the union"*. Our similarity measure relies heavily on the inclusion-exclusion principle: on the one hand, the exact computation of the number of all distinct subsequences of a sequence requires the computation of the *correction number*, $R(S, Y)$ in Equation 2, on the other hand the number of common subsequences needs the computation of the addition and correction terms, $A(S, T, Y)$ and $R(S, T, Y)$ in Equations 4 and 5. The computation drawback is the fact that the inclusion-exclusion formula has an exponential number of terms which, as mentioned previously in the complexity subsection, can become a problem with very long sequences and a position set L of big cardinality. This prompted our interest in approximating our similarity measure through the approximation of the inclusion-exclusion formula used in both ACS and ADS computations.

Linial-Nissan Approximation [14, Theorem 2]. Let $A_1, A_2, \ldots, A_N$ be a collection of sets. Suppose that $|\bigcap_{i\in S} A_i|$ is given for every subset $S \subset [N]$ of cardinality $|S| < K$. *How well can* $|\bigcup A_i|$ *be approximated based only on this information?* For any integers K, N there exist (explicitly given) constants $(\alpha_1^{K,N}, \alpha_2^{K,N}, \ldots, \alpha_K^{K,N})$ such that for every collection of sets $A_1, A_2, \ldots, A_N$, the quantity

$$\sum_{|S|\le K} \alpha_{|S|}^{K,N} |\bigcap_{i\in S} A_i|$$

differs from $|\bigcup_{i=1}^{N} A_i|$ by at most a factor of $1 + O(e^{-\frac{2K}{\sqrt{N}}})$ if $K \geq \Omega(\sqrt{N})$ or $O(\frac{N}{K^2})$ if $K \leq O(\sqrt{N})$.

The real numbers $\alpha_1^{K,N}, \alpha_2^{K,N}, \ldots, \alpha_K^{K,N}$ are defined by Linial and Nissan to be the coefficients of the linearly transformed Chebyshev polynomials expressed in terms of the polynomials $\binom{x}{1}, \binom{x}{2}, \ldots, \binom{x}{K}$. The vector $\overrightarrow{\alpha} = (\alpha_1^{K,N}, \alpha_2^{K,N}, \ldots, \alpha_K^{K,N})$ can be calculated very efficiently by solving a set of linear equations. Consider the above polynomial identity for $x = 1, \ldots, K$. The vector of coefficient is calculated as follows [14] :

$$\overrightarrow{\alpha} = \overrightarrow{t} \cdot \mathcal{M}^{-1}$$

where:

- $\mathcal{M}$ is the matrix whose (i, j) entry is $\binom{j}{i}$. The inverse $\mathcal{M}^{-1}(i, j)$ is $(-1)^{i+j}\binom{j}{i}$
- $\overrightarrow{t} = (q_{K,N}(1), q_{K,N}(2), \ldots, q_{K,N}(K))$ is the linearly transformed Chebyshev polynomials.
- $q_{K,N}(x) = 1 - \frac{T_k(\frac{2x-(N+1)}{N-1})}{T_k(\frac{-(N+1)}{N-1})}$
- $T_K(x)$ is a polynomial of degree K and is given by

$$T_K(x) = \frac{(x + \sqrt{x^2-1})^K + (x - \sqrt{x^2-1})^K}{2}$$

In our approximation method, every time that the position set is too big (i.e., $|L| \geq \sigma$ where σ is a user provided size threshold) we compute $\alpha_1^{K,N}, \alpha_2^{K,N}, \ldots, \alpha_k^{K,N}$ with $K = \lceil \sqrt{|L|} \rceil$ and $N = |L|$ and we approximate the inclusion-exclusion formula.

Using Linial and Nissan approximation lead us to Theorems 3 and 4 :

Theorem 3. *Let $S = \langle X_1 \ldots X_n \rangle$ and Y an itemset. Then,*

$$\phi_{LN}(S \circ Y) = 2^{|Y|} \cdot \phi_{LN}(S) - R_{LN}(S, Y) \tag{6}$$

with

$$R_{LN}(S, Y) = \sum_{k=1}^{K} \alpha_k^{K,N} \sum_{\substack{O \subseteq L(S,Y) \\ |O|=k}} \phi_{LN}(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j]) \cap Y|} - 1\right)$$

Where,
$N = |L(S, Y)|$, $K = \lceil \sqrt{|N|} \rceil$ *and* $\alpha_1^{K,N}, \alpha_2^{K,N}, \ldots, \alpha_K^{K,N}$ *are the coefficients Linial and Nissan.*

Theorem 4. *Let $S = \langle X_1 \ldots X_n \rangle$, $T = \langle X'_1 \ldots X'_m \rangle$ and Y an itemset. Then,*

$$\phi_{LN}(S \circ Y, T) = \phi_{LN}(S, T) + A_{LN}(S, T, Y) - R_{LN}(S, T, Y) \tag{7}$$

with

$$A_{LN}(S, T, Y) = \sum_{k'=1}^{K'} \alpha_k^{'K',N'} . \sum_{\substack{O' \subseteq L(T,Y) \\ |O'|=k'}} \phi_{LN}(S, T^{\min(O')-1}) \cdot \left(2^{|(\bigcap_{j \in O'} T[j]) \cap Y|} - 1\right) \tag{8}$$

and

$$R(S,T,Y) = \sum_{k=1}^{K} \alpha_k^{K,N} . \sum_{\substack{O \subseteq L(S,Y) \\ |O|=k}} \left(\sum_{k'=1}^{K'} \alpha_k'^{K',N'} . \sum_{\substack{O' \subseteq L(S,Y) \\ |O'|=k'}} f(O,O') \right) \tag{9}$$

where:

$$f(O,O') = \phi_{LN}(S^{\min(O)-1}, T^{\min(O')-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap \left(\bigcap_{j' \in O'} T[j']\right) \cap Y|} - 1 \right)$$

$N = |L(S,Y)|$, $N' = |L(T,Y)|$, $K = \lceil \sqrt{|L(S,Y)|} \rceil$, $K' = \lceil \sqrt{|L(T,Y)|} \rceil$ *and* $\alpha_1^{K,N}, \alpha_2^{K,N}, \ldots, \alpha_K^{K,N}$, $\alpha_1'^{K',N'}, \alpha_2'^{K',N'}, \ldots, \alpha_{K'}'^{K',N'}$ *are the coefficients Linial and Nissan.*

Example 6. *Consider* $S = \langle \{a,c,d,e,f,g,h,i,j,k\}\{a,b,d,e,f,g,h,i,j,k\}\{a,b,c,e,f,g,h,i,j,k\}\{a,b,c,d,f,g,h,i,j,k\}\{a,b,c,d,e,g,h,i,j,k\}\{a,b,c,d,e,f,h,i,j,k\}\{a,b,c,d,e,f,g,i,j,k\}\{a,b,c,d,e,f,g,h,j,k\}\{a,b,c,d,e,f,g,h,i,k\}\{a,b,c,d,e,f,g,h,i,j,k\}\rangle$.

The number of distinct subsequences for S^9 *is* $\phi(S^9) = 1\,233\,117\,889\,207\,727\,097\,068\,621\,596$ *and the position set for the last itemset is* $L(S^9, S[10]) = \{1,2,3,4,5,6,7,8,9\}$*, with the normal exact computation the final number of distinct subsequences is:*

$\phi(S^{10}) = 2^{|\{a,b,c,d,e,f,g,h,i,j,,k\}|} . \phi(S^9) - R(S^9, S[10]) = 2\,524\,192\,319\,208\,217\,367\,699\,468\,407\,013.$

Notice that the inclusion-exclusion formula used for the computation of $R(S^9, S[10])$ *contains* $\sum_{i=1}^{9} \binom{9}{i}$ *terms as following:*

$$\begin{aligned}
R(S^9, S[10]) \;=\; & +1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=1}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right) \\
& -1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=2}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right) \\
& +1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=3}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right) \\
& -1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=4}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right) \\
& +1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=5}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right) \\
& -1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=6}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right) \\
& +1. \left(\sum_{\substack{O \subseteq L(S^9, S[10]) \\ |O|=7}} \phi(S^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j \in O} S[j]\right) \cap S[10]|} - 1 \right) \right)
\end{aligned}$$

$$
\begin{aligned}
& -1.\Bigg(\sum_{\substack{O\subseteq L(S^9,S[10])\\|O|=8}} \phi(S^{\min(O)-1})\cdot\left(2^{|(\bigcap_{j\in O} S[j])\cap S[10]|}-1\right)\Bigg)\\
& +1.\Bigg(\sum_{\substack{O\subseteq L(S^9,S[10])\\|O|=9}} \phi(S^{\min(O)-1})\cdot\left(2^{|(\bigcap_{j\in O} S[j])\cap S[10]|}-1\right)\Bigg)\\
= \quad & 1\,233\,117\,889\,207\,727\,097\,068\,621\,595
\end{aligned}
$$

To do the Linial-Nissan approximation, remark that:

$$
\begin{aligned}
N &= |L| = 9\\
K &= \lceil\sqrt{N}\rceil = \lceil\sqrt{9}\rceil = 3
\end{aligned}
$$

The coefficients Linial and Nissan are:

$$
\begin{aligned}
\overrightarrow{\alpha} &= (\alpha_1^{3,9},\alpha_2^{3,9},\alpha_3^{3,9})\\
&= \overrightarrow{t}\cdot\mathcal{M}^{-1}
\end{aligned}
$$

The Matrix $\mathcal{M}^{-1}$ is:

$$
\mathcal{M}^{-1} = \begin{pmatrix} 1 & -2 & 3\\ 0 & 1 & -3\\ 0 & 0 & 1\end{pmatrix}
$$

The linearly transformation Chebyshev polynomials are:

$$
\begin{aligned}
\overrightarrow{t} &= (q_{3,9}(1),q_{3,9}(2),q_{3,9}(3))\\
&= (0.75,1.13,1.24)
\end{aligned}
$$

Where:

$$
\begin{aligned}
q_{3,9}(1) &= 1-\frac{T_3(\frac{2-(9+1)}{9-1})}{T_3(\frac{-(9+1)}{9-1})} = 1-\frac{T_3(-1)}{T_3(-\frac{10}{8})} = 1-\frac{-1}{-4,06} = 0.75\\
q_{3,9}(2) &= 1-\frac{T_3(\frac{4-(9+1)}{9-1})}{T_3(\frac{-(9+1)}{9-1})} = 1-\frac{T_3(-\frac{6}{8})}{T_3(-\frac{10}{8})} = 1-\frac{0.56}{-4,06} = 1.13\\
q_{3,9}(3) &= 1-\frac{T_3(\frac{6-(9+1)}{9-1})}{T_3(\frac{-(9+1)}{9-1})} = 1-\frac{T_3(-\frac{4}{10})}{T_3(-\frac{10}{8})} = 1-\frac{1}{-4,06} = 1.24
\end{aligned}
$$

Finally, the coefficients Linial and Nissan are:

$$\begin{aligned}
\overrightarrow{\alpha} &= (\alpha_1^{3,9}, \alpha_2^{3,9}, \alpha_3^{3,9}) \\
&= \overrightarrow{t} \cdot \mathcal{M}^{-1} \\
&= \begin{pmatrix} 0.75 & 1.13 & 1.24 \end{pmatrix} . \begin{pmatrix} 1 & -2 & 3 \\ 0 & 1 & -3 \\ 0 & 0 & 1 \end{pmatrix} \\
&= \begin{pmatrix} 0.75 & -0.36 & 0.1 \end{pmatrix}
\end{aligned}$$

After solving the associated system of linear equations, $\alpha_1^{3,9} = 0.75$; $\alpha_2^{3,9} = -0.36$; $\alpha_3^{3,9} = 0.1$.

The approximated correction term is

$$\begin{aligned}
R_{LN}(S^9, S[10]) &= \sum_{k=1}^{3} \alpha_k^{3,9} \sum_{\substack{O \subseteq L(S,Y) \\ |O|=k}} \phi_{LN}(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j]) \cap S[10]|} - 1\right) \\
&= \alpha_1^{3,9} . \sum_{\substack{O \subseteq L(S,Y) \\ |O|=1}} \phi_{LN}(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j]) \cap S[10]|} - 1\right) \\
&+ \alpha_2^{3,9} . \sum_{\substack{O \subseteq L(S,Y) \\ |O|=2}} \phi_{LN}(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j]) \cap S[10]|} - 1\right) \\
&+ \alpha_3^{3,9} . \sum_{\substack{O \subseteq L(S,Y) \\ |O|=3}} \phi_{LN}(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j]) \cap S[10]|} - 1\right) \\
&= 929\,812\,789\,770\,157\,650\,420\,104\,121{,}90.
\end{aligned}$$

Notice here that the formula contains only $\sum_{i=1}^{3} \binom{9}{i}$ *terms, which is already a significant computation gain. Finally,*

$$\begin{aligned}
\phi_{LN}(S^{10}) &= 2^{|\{a,b,c,d,e,f,g,h,i,j,,k\}|} \cdot \phi_{LN}(S^9) - R_{LN}(S^9, S[10]) \\
&= 2\,524\,495\,624\,307\,654\,937\,146\,116\,924\,486{,}1.
\end{aligned}$$

Position Set Tail Pruning Approximation . The second approximation technique takes root from the simple observation that the first elements of the position set L have almost no impact on the final result of the ACS or ADS because of the repeated multiplications by a power of 2 (recall the Theorems 1 and 2). Simply put, the first elements of a very large position set L are just negligible with respect to the final result of the inclusion-exclusion computation as they are dominated by the last elements that have a bigger multiplicand. Thus a very simple approximation technique is to only take into account the last (or the k last) element of the position set L when doing the computations (i.e. discard the tail). We define an index $\ell_{(S,Y)}$ is the maximal positions where the itemset Y has an intersection with the different itemsets $S[i]$, $i = 1 \ldots n$. Formally

$$\ell_{(S,Y)} = max\{i; S[i] \cap Y \neq \emptyset\}$$

Using Position Set Tail Pruning Approximation lead us to Theorems 5 and 6 :

Theorem 5. *Let $S = \langle X_1 \dots X_n \rangle$ and Y an itemset. Then,*

$$\phi_{TP}(S \circ Y) = 2^{|Y|} \cdot \phi_{TP}(S) - \phi_{TP}(S^{\ell_{(S,Y)}-1})(2^{|S[\ell_{(S,Y)}] \cap Y|} - 1) \tag{10}$$

Where:

$\phi_{TP}(S)$ is the position set tail pruning approximation value of all distinct subsequences for a given sequence S

Theorem 6. *Let $S = \langle X_1 \dots X_n \rangle$, $T = \langle X'_1 \dots X'_m \rangle$ and Y an itemset. Then,*

$$\begin{aligned} \phi_{TP}(S \circ Y, T) &= \phi_{TP}(S, T) + \phi_{TP}(S, T^{\ell_{(T,Y)}-1}) \cdot (2^{|(T[\ell_{(T,Y)}] \cap Y|} - 1) \\ &- \phi_{TP}(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1}) \cdot (2^{|(T[\ell_{(T,Y)}] \cap S[\ell_{(S,Y)}] \cap Y|} - 1) \end{aligned}$$

Where:

$\phi_{TP}(S, T)$ is the position set tail pruning approximation value of all common subsequences between two sequences S and T.

We prove that in the worst case (pathological sequences) the error induced by this approximation remains bounded with respect to $2^{|Y|}$ where Y is the last itemset in the position set L.

Proof. See Appendix. □

Example 7. *Consider the sequence S in Example 6. The position set tail pruning approximation starts by taking the maximal position in S^9 where the itemset $S[10]$ has an intersection. In this case $\ell_{S^9,S[10]}$ is equal to 9. Thus, the approximated correction term is*

$$R_{TP}(S^9, S[10]) = \phi_{TP}(S^8)(2^{|s[9] \cap S[10]|} - 1) = 1\,232\,514\,898\,438\,572\,496\,636\,423\,623$$

and

$$\phi_{TP}(S^{10}) = 2^{|\{a,b,c,d,e,f,g,h,i,j,,k\}|} \cdot \phi_{TP}(S^9) - R(S^9, S[10]) = 2\,524\,192\,922\,198\,986\,522\,299\,900\,604\,985$$

.

7 Experiments

In this section we empirically evaluate our similarity measure on synthetic and real-world datasets. Our approach is implemented in Java. The goal of these experiments is to show the usefulness of our proposed similarity measure and all the analysis are run over a MacBook Pro with a 2.5GHz Intel Core i5, 4GB of RAM Memory running OS X 10.6.8.

Patients	*Trajectories*
$Patient_1$	$\langle\{54, CHU_{nancy}, C34, ZBQK\}\{57, CL_{metz}, Z51, ZBQK\}\rangle$
$Patient_2$	$\langle\{54, CHU_{nancy}, I70, ZBQK, GFFA\}\{67, CL_{strasbourg}, Z51, GFFA\}\rangle$
$Patient_3$	$\langle\{75, CH_{paris}, C34, ZBQK\}\{57, CL_{metz}, Z51, GFFA, GLLD\}\rangle$

Table 3: Healthcare trajectories of 4 patients.

7.1 Healthcare Trajectory Clustering

Our first batch of experiments was conducted with healthcare data from the PMSI[1], a French nationwide hospital information system. In this system, each hospital stay leads to the collection of a minimal and standardized set of administrative and medical data. Although they are essentially used for payment purposes, data from the PMSI can also serve the exploration of patients journeys through several hospitalizations and feed a decision support system, helping healthcare managers for strategic planning and organization of the healthcare system. Such a goal cannot be reached without a recomposition and a better understanding of the so called healthcare trajectories.

In a healthcare trajectory, every hospitalization can be described by the healthcare institution where it takes place, its main cause (diagnosis) and a set of medical and surgical procedures underwent by the patient. For example $\{Moselle, Metz\ regional\ hospital,\ lung\ cancer,\ chest\ radiography\}$ represents a stay in the regional hospital of Metz, in the administative area of Moselle [2] for a lung cancer where the patient underwent a chest radiography. A patient trajectory can thus be seen as a sequence of itemsets, each itemset representing one hospitalization. Computing similarity between patient healthcare trajectories will open the way for patients clustering.

Our dataset contains 828 patients suffering from lung cancer and living in the Lorraine region, in the east of France. In the PMSI, information is coded using controlled vocabularies. In particular, diagnoses are coded with the International Classification of Diseases (ICD10)[3]and medical procedures with the French nomenclature for procedures (CCAM[4]). Table 3 shows an example of care trajectories for 3 patients. For example, $Patient_1$ has two hospitalizations. He was admitted in the University Hospital of Nancy(coded as CHU_{nancy}), in Meurthe-et-Moselle (54) for a Lung cancer ($C34$), and underwent a chest Radiography ($ZBQK$). Then, he was hospitalized in a private clinic in Metz(CL_{metz}), Moselle (57), for a chemotherapy session ($Z51$) where he also had a chest radiography. Figure 1 shows the distribution of the length of healthcare trajectories in our dataset, the median length beeing 11 stays.

Our similarity measure is used to build a similarity matrix between patient trajectories. A hierarchical clustering procedure is then applied on the matrix using the *hclust method* from the R software [20]. The number of clusters is set to 4 based on a priori knowledge from our experts. To assess the quality of our similarity measure, we describe each cluster with *"representative"* trajectories. To do so, we first extract frequent closed sequential patterns from our dataset by applying CloSpan [21] with a minimal support of 10%. Then, the support of the obtained patterns is computed in each of the 4 different clusters. Patterns having the highest variation of support between clusters were detected using a chi-squared measure (χ^2). Patterns with a high χ^2 and a high support in a given cluster can be seen as a distinguishing feature of that cluster. After discussing these results with our medical expert, two criteria appeared to be related

[1]Programme de Médicalisation des Systèmes d'Information
[2]one of the 101 departments of France
[3]http://apps.who.int/classifications/apps/icd/icd10online/
[4]http://www.ameli.fr/accueil-de-la-ccam/index.php

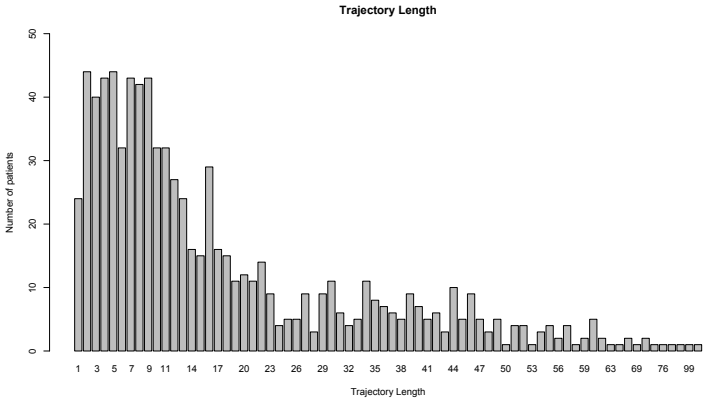


Figure 1: Distribution length of the patient trajectories

with the results of the clustering process, the *place of hospitalization* and the *length of the care trajectories*. We describe in the following the different clusters built with our similarity measure and its associated medical explanations.

Cluster 1 The pattern $\langle\{54, C34, GFFA\}\rangle$ has a high χ^2 and a high support in Cluster 1. Patients in that cluster underwent a pneumonectomy ($GFFA$) in a Meurthe-et-Moselle (department 54) hospital. They usually have a short trajectory (median is 6 stays).

Cluster 2 The pattern $\langle\{57, C34, GFFA\}\rangle$ is frequent in cluster 2 (support is around 80%) but not in the others. It contains patients having underwent a pneumonectomy ($GFFA$) in a Moselle (department 57) hospital. Cluster 2 is characterized by longer patterns with repeated stays in the departement of Moselle, such as $\langle\{57\}\{57\}\{57\}\{57\}\{57\}\{57\}\rangle$. Patients in that cluster have a median trajectory length of 13.

Cluster 3 The pattern $\langle\{54, Z51\}\{54, Z51\}\{54, Z51\}\{54, Z51\}\{54, Z51\}\{54, Z51\}\{54, Z51\}\rangle$ is more represented in Cluster 3 (support is approximately 95%) than in any other cluster. It represents patients who have repeated chemotherapy sessions in Meurthe-et-Moselle. The median trajectory length in that cluster is 37.

Cluster 4 This cluster is similar to cluster 3 (chemotherapy sessions) but with stays occuring in various places, especially in the bordering region of Alsace.

As can be seen, the clustering is based on a combination of different trajectory lengths and precise diagnoses or procedures such as pneumonectomy or chemotherapies. Because our similarity measure is only based on the number of common subsequences, we were able to build clusters that were close to the knowledge that doctors and experts have on patients trajectories in the Lorraine region. Furthermore, for our experts, these results are very encouraging as they correspond to the two main modalities in care for lung cancer: (i) surgery only or (ii) chemothery with (or without) surgery. They also highlight some important geographical characteristics in care trajectories.

The runtime for building the similarity matrix for 828 patient trajectories is about 25 minutes. We applied the two proposed approximations: the Linial-Nissan approximation and the Position set tail pruning approximation. With the Linial-Nissan approximation, the computation time is

	C_1	C_2	C_3	C_4	Average
Linial-Nissan Approximation	0.056	0.066	0.13	0.17	0.1
Position Set Tail Pruning Approximation	0.04	0.073	0.07	0.2	0.09

Table 4: Entropy of clusters obtained with the Linial-Nissan and Position set tail pruning approximations.

around 10 minutes to build the similarity matrix. With the position set tail pruning approximation, it takes about 2 minutes to build the same similarity matrix. To assess the quality of the approximations, we compare the clusters obtained using the two approximations with the clusters obtained by using the similarity measure without any approximation. We use the Shannon entropy to evaluates how well the clusters, obtained by using the two approximation solutions, matches with the original clusters. Table 4 shows the entropy of the clusters obtained with the two approximations. The smaller the entropy of a cluster, the more homogeneous the cluster is (i.e., it contains similar objects). The average entropy for clustering with the Linial-Nissan approximation is 0.1 and with the position set tail pruning approximation is 0.09. This result highlights the fact that approximating our similarity measure still yields good and competitive conclusions.

7.2 Experiments on Synthetic Datasets

In the following, we study the scalability of our measure computation. We assess the different runtimes with respect to three different parameters:

- The average number of itemsets in a sequence.
- The average number of items in each itemset of a sequence
- The total number of sequences that are processed through the similarity computation.

We carry out our experiments on our three propositions: the *normal* similarity measure and the two proposed approximations.

Figures 2, 3 and 4 represent the evolution of the running time of 499 500 ($\frac{n\times(n-1)}{2}$) comparisons over 1 000 sequences w.r.t the average number of items in each itemset and the average number of itemsets in each sequence. We run this test on several types of sequences: sequences with itemsets of cardinality 5, 10, 15, 20, 25 and with several lengths : 5, 10, 15 and 20 itemsets. As expected, the plots on Figure 2 show that the execution time for calculating the similarity matrix without any approximation takes a long time due to the complexity of the inclusion-exclusion formula. With the Linial-Nissan approximation, the execution time is greatly reduced as seen on Figure 3. Finally, the graphs on Figure 4 show that the position set tail pruning approximation is more time efficient. The execution time does not change significantly when the cardinality of the itemsets or the length of the sequences increase. Figure 5 presents the comparison of runtime for calculating the similarity matrix for our approach and its approximations. In addition, we compare our approach with two well-known measures: the longest common subsequence and the edit distance. The plots show that our similarity measure, along with the position set tail pruning approximation, takes the same runtime as the edit distance or the longest common subsequence computations. In addition, the similarity matrices computed with our similarity measure and the different approximation methods are similar as can be seen on Figure 6.

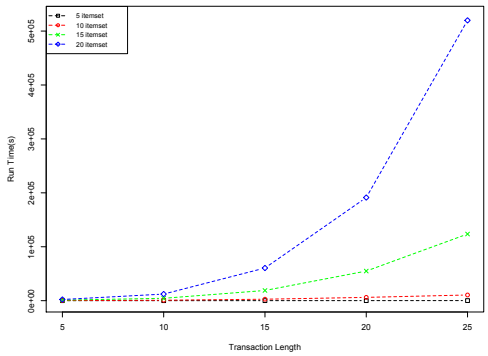


Figure 2: Runtime for calculating the similarity matrix of 1 000 sequences based on the sequences and itemsets lengths

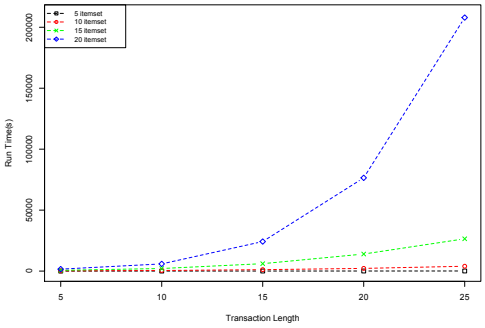


Figure 3: Runtime for calculating of similarity matrix (Linial-Nissan approximation) of 1 000 sequences w.r.t the sequences and itemsets lengths

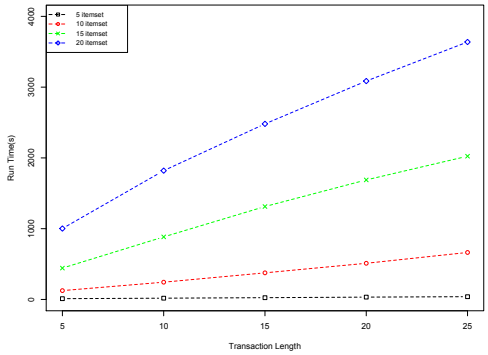


Figure 4: Runtime for calculating of similarity matrix (position set tail pruning approximation) of 1 000 sequences w.r.t the sequences and itemsets lengths

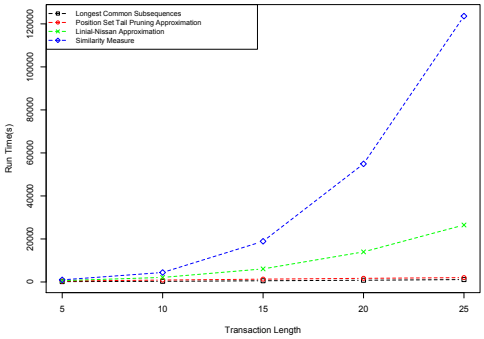
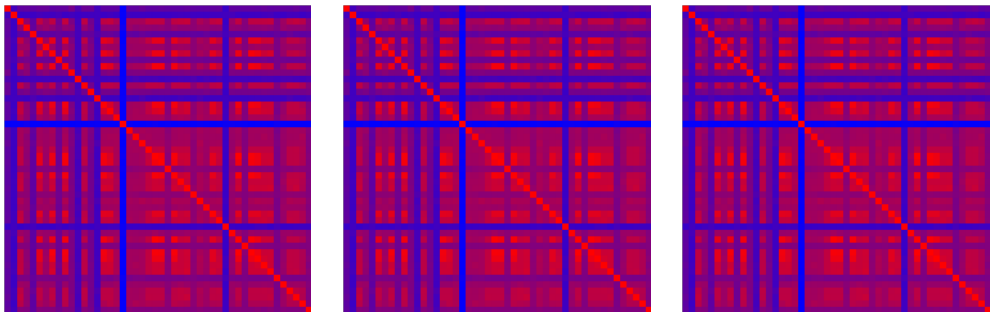


Figure 5: Comparison runtime for calculating the similarity matrix of 1 000 sequences with a sequence length of 15 and different values of itemset lengths'



(a) Without approximations (b) Linial-Nissan approximation (c) Position set tail pruning approximation

Figure 6: Matrix similarity generated from a sample of 50 sequences from synthetic datasets

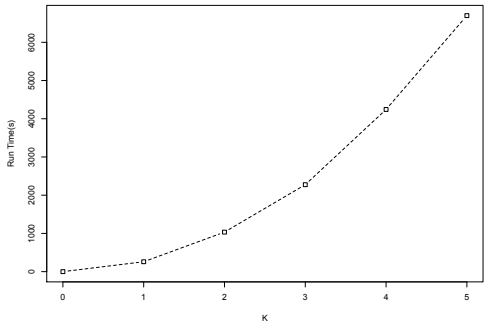


Figure 7: Runtime for the Linial-Nissan approximation with varying k and the sequences and itemsets length is 15

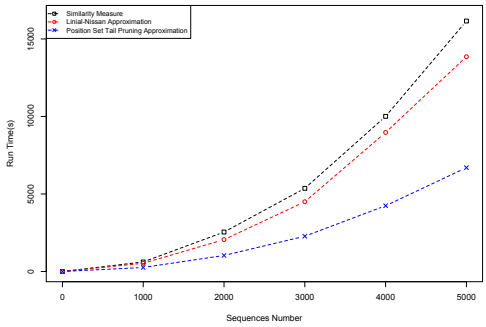


Figure 8: Time for calculating the similarity matrix based on the number of sequences

The plot on Figure 7 presents the impact of the different values of the parameter k on the runtime for the Linial-Nissan approximation. This Figure shows that the calculation time increases by a almost a factor of two, but remains acceptable, when we increase the value of k.

Finally, Figure 8 shows the time needed to compute the similarity matrix when the number of sequences increases. In each case, there is $\frac{n\times(n-1)}{2}$ similarity comparisons where n is the number of sequences in the data set. We run this test over sequences with 10 itemsets on average and with 10 items in each itemset. For 5 000 sequences (i.e. 12 497 500 similarity comparisons), the execution time for our similarity measure is about 4 hours (16 000 seconds), about 3 hours (14 000 seconds) for the Linial-Nissan approximation and about 1,5 hours (6 000 seconds) for the position set tail pruning approximation.

These experiments highlight the fact that our measure is efficient in term of runtime for a large panel of sequences with different varying parameters.

8 Conclusion

In this paper, we study the problem of counting all common subsequences between two sequences of itemsets. We present theoretical results and an efficient dynamic programming algorithm (ACS) to count the number of common subsequences between two sequences. This solution allows us to define in a simple and intuitive manner a similarity measure between two sequences S and T. In addition, we propose two approximation methods to speed up the computation for long sequence like biological sequences. This similarity has been successfully applied for the analysis of real-world healthcare and synthetic data sets.

References

[1] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," in *Proceedings of the 1994 ACM SIGMOD international conference on Management of data*, ser. SIGMOD '94. New York, NY, USA: ACM, 1994, pp. 419–429. [Online]. Available: http://doi.acm.org/10.1145/191839.191925

[2] C. Sander and R. Schneider, "Database of homology-derived protein structures and the structural meaning of sequence alignment," *Proteins*, vol. 1, no. 9, pp. 56–68, 1991.

[3] C. Chothia and M. Gerstein, "Protein evolution. how far can sequences diverge?" *Nature*, vol. 6617, no. 385, pp. 579–581, 1997.

[4] Q. Yang and H. H. Zhang, "Web-log mining for predictive web caching," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, p. 2003, 2003.

[5] J. Serrà, H. Kantz, X. Serra, and R. G. Andrzejak, "Predictability of music descriptor time series and its application to cover song detection," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 20, no. 2, pp. 514–525, 2012.

[6] C. Leslie, E. Eskin, and W. S. Noble, "The spectrum kernel: a string kernel for svm protein classification." *Pacific Symposium On Biocomputing*, vol. 575, no. 50, pp. 564–575, 2002. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/11928508

[7] T. Xiong, S. Wang, Q. Jiang, and J. Z. Huang, "A new markov model for clustering categorical sequences," in *Proceedings of the 2011 IEEE 11th International Conference on Data Mining*, ser. ICDM '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 854–863. [Online]. Available: http://dx.doi.org/10.1109/ICDM.2011.13

[8] V. Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[9] J. Herranz, J. Nin, and M. Sole, "Optimal symbol alignment distance: A new distance for sequences of symbols," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, pp. 1541–1554, 2011.

[10] E. Keogh, "Exact indexing of dynamic time warping," in *Proceedings of the 28th international conference on Very Large Data Bases*, ser. VLDB '02. VLDB Endowment, 2002, pp. 406–417. [Online]. Available: http://dl.acm.org/citation.cfm?id=1287369.1287405

[11] H. Wang and Z. Lin, "A novel algorithm for counting all common subsequences," in *Proceedings of the 2007 IEEE International Conference on Granular Computing*, ser. GRC

'07. Washington, DC, USA: IEEE Computer Society, 2007, pp. 502–. [Online]. Available: http://dx.doi.org/10.1109/GRC.2007.15

[12] S. Wodak and J. Janin, "Structural basis of macromolecular recognition." *Adv Protein Chem*, vol. 61, pp. 9–73, 2002.

[13] D. S. Hirschberg, "A linear space algorithm for computing maximal common subsequences," *Commun. ACM*, vol. 18, no. 6, pp. 341–343, Jun. 1975. [Online]. Available: http://doi.acm.org/10.1145/360825.360861

[14] N. Linial and N. Nisan, "Approximate inclusion-exclusion," *Combinatorica*, vol. 10, no. 4, pp. 349–365, 1990.

[15] J. Oncina and M. Sebban, "Learning stochastic edit distance: Application in handwritten character recognition," *Pattern Recogn.*, vol. 39, no. 9, pp. 1575–1587, Sep. 2006. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2006.03.011

[16] F. Muzaffar, B. Mohsin, F. Naz, and L. F. Jawed, "Dsp implementation of voice recognition using dynamic time warping algorithm," *IEEE Explore*, pp. 1–7, 2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4382877

[17] M. J. Z. Karlton Sequeira, "Admit: Anomaly-base data mining for intrusions," in *8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul 2002.

[18] M. Vlachos, M. Hadjieleftheriou, D. Gunopulos, and E. J. Keogh, "Indexing multidimensional time-series with support for multiple distance measures," in *KDD*, L. Getoor, T. E. Senator, P. Domingos, and C. Faloutsos, Eds. ACM, 2003, pp. 216–225.

[19] C. Elzinga, S. Rahmann, and H. Wang, "Algorithms for subsequence combinatorics," *Theor. Comput. Sci.*, vol. 409, no. 3, pp. 394–404, 2008. [Online]. Available: http://dx.doi.org/10.1016/j.tcs.2008.08.035

[20] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2012, ISBN 3-900051-07-0. [Online]. Available: http://www.R-project.org/

[21] X. Yan, J. Han, and R. Afshar, "Clospan: Mining closed sequential patterns in large datasets," in *In SDM*, 2003, pp. 166–177.

Proof of Lemma 1

Let $T = \langle T_1, \ldots, T_m \rangle$ be a sequence that is counted multiple times; i.e., $T \in (\varphi(S) \circ \mathcal{P}_{\geq 1}(Y)) \cap \varphi(S)$. Clearly $T_m \in \mathcal{P}_{\geq 1}(Y)$ as otherwise T would not have been in $\varphi(S) \circ \mathcal{P}_{\geq 1}(Y)$. Let k denote $\max\{j | T_m \subseteq S[j]\}$. Since $T \in \varphi(S)$, such k must exist. Then, $k \in L(S, Y)$, since k is the largest index for which $S[k] \cap Y$ includes T_m. Therefore, $T \in \varphi(S^{k-1}) \circ \mathcal{P}_{\geq 1}(S[k] \cap Y)$ for a $k \in L(S, Y)$. □

Proof of Theorem 1

The proof is a simple application of the inclusion-exclusion principle to compute the cardinality of the union of Lemma 1:

$$R(S,Y) = \left| \bigcup_{\ell \in L(S,Y)} \left\{ \varphi(S^{\ell-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap Y) \right\} \right|$$

$$R(S,Y) = \sum_{K \subseteq L(S,Y)} (-1)^{|K|+1} \left| \bigcap_{\ell \in K} \left\{ \varphi(S^{\ell-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap Y) \right\} \right|$$

The proof is completed by the following two observations:

$$\begin{aligned} set_K &:= \bigcap_{\ell \in K} \left\{ \varphi(S^{\ell-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap Y) \right\} \\ &= \varphi(S^{\min(K)-1}) \circ \mathcal{P}_{\geq 1}((\cap_{k \in K} S[k]) \cap Y) \end{aligned}$$

Indeed; any sequence of length m in set_K has $T^{m-1} \in S^{\min(K)-1}$, and $T_m \in \mathcal{P}_{\geq 1}(S[k] \cap Y)$, for all $k \in K$. And, the second observation:

$$|set_K| = \phi(S^{\min(K)-1}) \cdot \left(2^{|(\cap_{k \in K} S[k]) \cap Y|} - 1 \right)$$

□

Proof of Theorem 2

1. No items in Y appear in any itemset of S and T, in this case the set of all common distinct subsequences between $S \circ Y$ and T is exactly the same set of all common distinct subsequences between S and T. Hence, $\phi(S \circ Y, T) = \phi(S, T)$.

2. If at least an item in Y appears in either one of the sequences S or T (or both), then $\varphi(S \circ Y, T)$ is expressed as the union of the set of all common distinct subsequences between S and T with the set of added sequences $\mathcal{A}$ *without* the set of repeated sequences $\mathcal{R}$. Formally,

$$\varphi(S \circ Y, T) = \varphi(S, T) \cup \mathcal{A} \backslash \mathcal{R} \tag{11}$$

with

$$\mathcal{A} = \left\{ \bigcup_{\ell' \in L(T,Y)} \varphi(S, T^{\ell'-1}) \circ \mathcal{P}_{\geq 1}(T[\ell'] \cap Y) \right\} \tag{12}$$

and

$$\mathcal{R} = \left\{ \bigcup_{\ell \in L(S,Y)} \left\{ \bigcup_{\ell' \in L(T,Y)} \varphi(S^{\ell-1}, T^{\ell'-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap T[\ell'] \cap Y) \right\} \right\} \tag{13}$$

Notice that because these three sets are disjoint, the cardinality of $\varphi(S \circ Y, T)$ can be simply expressed as $|\varphi(S \circ Y, T)| = |\varphi(S,T)| + |\mathcal{A}| - |\mathcal{R}|$. Using the inclusion-exclusion principle, $|\mathcal{A}|$, denoted as $A(S,T,Y)$ can be written as,

$$\begin{aligned} A(S,T,Y) &= \left| \bigcup_{\ell \in L(T,Y)} \left\{ \varphi(S, T^{\ell-1}) \circ \mathcal{P}_{\geq 1}(T[\ell] \cap Y) \right\} \right| \\ &= \sum_{K \subseteq L(T,Y)} (-1)^{|K|+1} \, |set_K| \end{aligned}$$

where

$$set_K = \bigcap_{\ell \in K} \left\{ \varphi(S, T^{\ell-1}) \circ \mathcal{P}_{\geq 1}(T[\ell] \cap Y) \right\}$$

$A(S,T,Y)$ is completed by the following two observations:

$$\begin{aligned} set_K &:= \bigcap_{\ell \in K} \left\{ \varphi(S, T^{\ell-1}) \circ \mathcal{P}_{\geq 1}(T[\ell] \cap Y) \right\} \\ &= \varphi(S, T^{\min(K)-1}) \circ \mathcal{P}_{\geq 1}((\cap_{k \in K} T[k]) \cap Y) \end{aligned}$$

And, the second observation:

$$|set_K| = \phi(S, T^{\min(K)-1}) \cdot \left(2^{|(\cap_{k \in K} T[k]) \cap Y|} - 1 \right)$$

$A(S,T,Y)$ can be written as,

$$A(S,T,Y) = \sum_{K \subseteq L(T,Y)} (-1)^{|K|+1} \cdot \phi(S, T^{\min(K)-1}) \cdot \left(2^{\left| \left(\bigcap_{j \in K} X'_j \right) \cap Y \right|} - 1 \right) \tag{14}$$

The same inclusion-exclusion reasoning applies to the cardinality of $\mathcal{R}$, denoted $R(S,T,Y)$

$$\begin{aligned} R(S,T,Y) &= \left| \left\{ \bigcup_{\ell \in L(S,Y)} \left\{ \bigcup_{\ell' \in L(T,Y)} \mathcal{D}_{\ell,\ell'} \right\} \right\} \right| \\ &= \sum_{K \subseteq L(S,Y)} (-1)^{|K|+1} \cdot \sum_{K' \subseteq L(T,Y)} (-1)^{|K'|+1} \cdot \left| set_{K,K'} \right| \end{aligned}$$

and

$$set_{K,K'} = \bigcap_{\ell \in K} \bigcap_{\ell' \in K'} \varphi(S^{\ell-1}, T^{\ell'-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap T[\ell'] \cap Y)$$

The final result follows after noticing that,

$$set_{K,K'} = \bigcap_{\ell \in K} \bigcap_{\ell' \in K'} \varphi(S^{\ell-1}, T^{\ell'-1}) \circ \mathcal{P}_{\geq 1}(S[\ell] \cap T[\ell'] \cap Y)$$

$$set_{K,K'} = \varphi(S^{\min(K)-1}, T^{\min(K')-1}) \circ \mathcal{P}_{\geq 1}\Big((\cap_{k \in K} S[k]) \cap (\cap_{k' \in K'} T[k']) \cap Y\Big)$$

$R(S,T,Y)$ can be written as,

$$R(S,T,Y) = \sum_{K \subseteq L(S,Y)} (-1)^{|K|+1} \cdot \sum_{K' \subseteq L(T,Y)} (-1)^{|K'|+1} \cdot D(S,T,Y,K,K') \tag{15}$$

where

$$D(S,T,Y,K,K') = \phi(S^{\min(K)-1}, T^{\min(K')-1}) \cdot 2^{\left|(\bigcap_{j \in K} X_j) \cap \left(\bigcap_{j' \in K'} X'_{j'}\right) \cap Y\right|} - 1$$

□

Proof of Theorem 5

The proof presents range of the error for the position set tail pruning approximation value of the all distinct subsequences for a given sequence $S \circ Y$.

The Best Case : In this case, L(S,Y) contains only one item, that means:

$$L(S,Y) \quad = \quad \{\ell_{(S,Y)}\}$$

The exact value of all distinct subsequences of $S \circ Y$, $\phi(S \circ Y)$ is:

$$\phi(S \circ Y) = \phi(S) \cdot 2^{|Y|} - \phi(S^{\ell_{(S,Y)}})(2^{|S[\ell_{(S,Y)}] \cap Y|} - 1) \tag{16}$$

The approximation value of all distinct subsequences of $S \circ Y$, $\phi_{TP}(S \circ Y)$ is:

$$\phi_{TP}(S \circ Y) = \phi_{TP}(S) \cdot 2^{|Y|} - \phi_{TP}(S^{\ell_{(S,Y)}})(2^{|S[\ell_{(S,Y)}] \cap Y|} - 1) \tag{17}$$

The error is the difference between the exact value and the approximation value. It is the difference between the Equation 16 and 17, as following:

$$error_{ADS} = \big(\phi_{TP}(S) - \phi(S)\big) \cdot 2^{|Y|} - \big(\phi_{TP}(S^{\ell_{(S,Y)}}) - \phi(S^{\ell_{(S,Y)}})\big)(2^{|S[\ell_{(S,Y)}] \cap Y|} - 1)$$

The Worst Case : In this case, L(S,Y) contains $\ell_{(S,Y)}$ items, that means:

$$L(S,Y) \quad = \quad \{1, \ldots, \ell_{(S,Y)}\}$$

The exact value of all distinct subsequences of $S \circ Y$, $\phi(S \circ Y)$ is:

$$\phi(S \circ Y) = \phi(S) \cdot 2^{|Y|} - \sum_{O \subseteq L(S,Y)} (-1)^{|O|+1} \left(\phi(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j]) \cap Y|} - 1\right)\right) \tag{18}$$

The approximation value of all distinct subsequences of $S \circ Y$, $\phi_{TP}(S \circ Y)$ is:

$$\phi_{TP}(S \circ Y) = \phi_{TP}(S) \cdot 2^{|Y|} - \phi_{TP}(S^{\ell_{(S,Y)}})(2^{|S[\ell_{(S,Y)}]\cap Y|} - 1) \tag{19}$$

The error is the difference between the exact value and the approximation value. It is the difference between the Equation 18 and 19, as following:

$$\begin{aligned} error_{ADS} &= \big(\phi_{TP}(S) - \phi(S)\big) \cdot 2^{|Y|} - \big(\phi_{TP}(S^{\ell_{(S,Y)}}) - \phi(S^{\ell_{(S,Y)}})\big)\big(2^{|S[\ell_{(S,Y)}]\cap Y|} - 1\big) \\ &+ \sum_{\substack{O \subseteq L(S,Y) \\ O \neq \{\ell_{(S,Y)}\}}} (-1)^{|O|+1} \left(\phi(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j])\cap Y|} - 1\right)\right) \end{aligned}$$

Range of the error for the approximation value is:

$$\begin{aligned} error_{ADS} \in \Bigg[& \big(\phi_{TP}(S) - \phi(S)\big) \cdot 2^{|Y|} - \big(\phi_{TP}(S^{\ell_{(S,Y)}}) - \phi(S^{\ell_{(S,Y)}})\big)(2^{|S[\ell_{(S,Y)}]\cap Y|} - 1) \\ & , \big(\phi_{TP}(S) - \phi(S)\big) \cdot 2^{|Y|} - \big(\phi_{TP}(S^{\ell_{(S,Y)}}) - \phi(S^{\ell_{(S,Y)}})\big)\big(2^{|S[\ell_{(S,Y)}]\cap Y|} - 1\big) \\ & + \sum_{\substack{O \subseteq L(S,Y) \\ O \neq \{\ell_{(S,Y)}\}}} (-1)^{|O|+1} \left(\phi(S^{\min(O)-1}) \cdot \left(2^{|(\bigcap_{j \in O} S[j])\cap Y|} - 1\right)\right) \Bigg] \end{aligned}$$

□

Proof of Theorem 6

We present here range of the error for the position set tail pruning approximation of all distinct common subsequences between $S \circ Y$ and T.

The Best Case : In this case, L(S,Y) and L(T,Y) contain only one item and , that means:

$$\begin{aligned} L(S,Y) &= \{\ell_{(S,Y)}\} \\ L(T,Y) &= \{\ell_{(T,Y)}\} \end{aligned}$$

The exact value of all distinct common subsequences between $S \circ Y$ and T, $\phi(S \circ Y, T)$ is:

$$\begin{aligned} \phi(S \circ Y, T) &= \phi(S,T) + \phi(S, T^{\ell_{(T,Y)}-1}).\big(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\big) \\ &\quad -\phi(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1}).\big(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\big) \end{aligned} \tag{20}$$

The approximation value of all distinct common subsequences between $S \circ Y$ and T, $\phi_{TP}(S \circ Y, T)$ is:

$$\begin{aligned} \phi_{TP}(S \circ Y, T) &= \phi_{TP}(S,T) + \phi_{TP}(S, T^{\ell_{(T,Y)}-1}).\big(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\big) \\ &- \phi_{TP}(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1}).\big(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\big) \end{aligned} \tag{21}$$

The error is the difference between the exact value and the approximation value. It is the difference between the Equation 20 and 22, as following:

$$
\begin{aligned}
error_{ACS} &= \Big(\phi(S,T) - \phi_{TP}(S,T)\Big) + \Big(\phi(S, T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S, T^{\ell_{(T,Y)}-1})\Big)\Big(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\Big) \\
&- \Big(\phi(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1})\Big)\Big(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\Big)
\end{aligned}
$$

The Worst Case : In this case, L(S,Y) contains $\ell_{(S,Y)}$ items and L(S,T) contains $\ell_{(T,Y)}$ items, that means:

$$
\begin{aligned}
L(S,Y) &= \{1, \ldots, \ell_{(S,Y)}\} \\
L(T,Y) &= \{1, \ldots, \ell_{(T,Y)}\}
\end{aligned}
$$

The exact value of all distinct common subsequences between $S \circ Y$ and T, $\phi(S \circ Y, T)$ is:

$$
\begin{aligned}
&\phi(S \circ Y, T) = \phi(S,T) + \sum_{O \subseteq L(T,Y)} (-1)^{|O|+1} \left(\phi(S, T^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j\in O} T[j]\right)\cap Y|} - 1\right)\right) \\
&- \sum_{O \subseteq L(S,Y)} (-1)^{|O|+1} \left(\sum_{O' \subseteq L(T,Y)} (-1)^{|O'|+1} \cdot \phi(S^{\min(O)-1}, T^{\min(O')-1}) \cdot \left(2^{\left|\left(\bigcap_{j\in O} S[j]\right)\cap\left(\bigcap_{j'\in O'} T[j']\right)\cap Y\right|} - 1\right.\right.
\end{aligned}
$$

The approximation value of all distinct common subsequences between $S \circ Y$ and T, $\phi_{TP}(S \circ Y, T)$ is:

$$
\begin{aligned}
\phi_{TP}(S \circ Y, T) &= \phi_{TP}(S,T) + \phi_{TP}(S, T^{\ell_{(T,Y)}-1}).\left(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\right) \\
&- \phi_{TP}(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1}).\left(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\right)
\end{aligned}
$$

The error is the difference between the exact value and the approximation value. It is the difference between $\phi(S \circ Y, T)$ and $\phi_{TP}(S \circ Y, T)$, as following:

$$
\begin{aligned}
error_{ACS} &= \Big(\phi(S,T) - \phi_{TP}(S,T)\Big) + \Big(\phi(S, T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S, T^{\ell_{(T,Y)}-1})\Big).\left(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\right) \\
&- \Big(\phi(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S^{\ell_{(S,Y)}-1}, T^{\ell_{(T,Y)}-1})\Big).\left(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\right) \\
&+ \sum_{\substack{O \subseteq L(T,Y) \\ O \neq \{\ell_{(T,Y)}\}}} (-1)^{|O|+1} \left(\phi(S, T^{\min(O)-1}) \cdot \left(2^{|\left(\bigcap_{j\in O} T[j]\right)\cap Y|} - 1\right)\right) \\
&- \sum_{O \subseteq L(S,Y)} (-1)^{|O|+1} \sum_{\substack{O' \subseteq L(T,Y) \\ O \neq \{\ell_{(S,Y)}\} \\ O' \neq \{\ell_{(T,Y)}\}}} (-1)^{|O'|+1} \cdot \phi(S^{o-1}, T^{o'-1}) \cdot \left(2^{\left|\left(\bigcap_{j\in O} S[j]\right)\cap\left(\bigcap_{j'\in O'} T[j']\right)\cap Y\right|} - 1\right)
\end{aligned}
$$

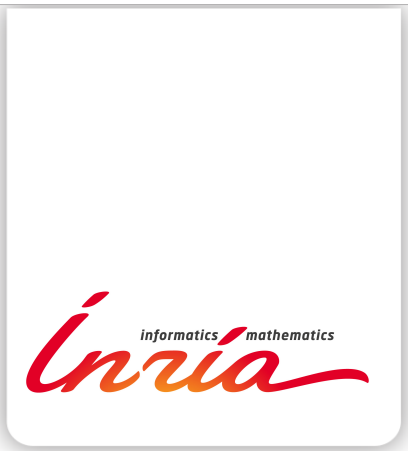
where o is $\min(O)$ and o' is $\min(O')$.

Range of the error for the approximation value is:

$$
\begin{aligned}
error_{ACS} \in \Bigg[& \\
& \Big(\phi(S,T) - \phi_{TP}(S,T)\Big) + \Big(\phi(S,T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S,T^{\ell_{(T,Y)}-1})\Big)\Big(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\Big) \\
- & \Big(\phi(S^{\ell_{(S,Y)}-1},T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S^{\ell_{(S,Y)}-1},T^{\ell_{(T,Y)}-1})\Big)\Big(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\Big) \\
& , \\
& \Big(\phi(S,T) - \phi_{TP}(S,T)\Big) + \Big(\phi(S,T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S,T^{\ell_{(T,Y)}-1})\Big).\big(2^{|T[\ell_{(T,Y)}]\cap Y|} - 1\big) \\
- & \Big(\phi(S^{\ell_{(S,Y)}-1},T^{\ell_{(T,Y)}-1}) - \phi_{TP}(S^{\ell_{(S,Y)}-1},T^{\ell_{(T,Y)}-1})\Big).\big(2^{|S[\ell_{(S,Y)}]\cap T[\ell_{(T,Y)}]\cap Y|} - 1\big) \\
+ & \sum_{\substack{O\subseteq L(T,Y)\\ O\neq\{\ell_{(T,Y)}\}}} (-1)^{|O|+1}\Big(\phi(S,T^{o-1})\cdot\Big(2^{|(\bigcap_{j\in O}T[j])\cap Y|} - 1\Big)\Big) \\
- & \sum_{O\subseteq L(S,Y)} (-1)^{|O|+1} \sum_{\substack{O'\subseteq L(T,Y)\\ O\neq\{\ell_{(S,Y)}\}\\ O'\neq\{\ell_{(T,Y)}\}}} (-1)^{|O'|+1}\cdot\phi(S^{o-1},T^{o'-1})\cdot\big(2^{\left|(\bigcap_{j\in O}S[j])\cap(\bigcap_{j'\in O'}T[j'])\cap Y\right|} - 1\big) \\
\Bigg] &
\end{aligned}
$$

where o is $\min(O)$ and o' is $\min(O')$. □

Contents



RESEARCH CENTRE
NANCY – GRAND EST

615 rue du Jardin Botanique
CS20101
54603 Villers-lès-Nancy Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399