Summarizing Process Traces for Analysis Tasks: An Intuitive and User-controlled Approach

Phuong Nguyen[1], Vatche Isahagian[2], Vinod Muthusamy[2] and Aleksander Slominski[2]

[1]*Google*
[2]*IBM Research*

Abstract

Domains such as business processes and workflows require working with multi-dimensional ordered objects. There is a need to analyze this data for operational insights. For example, in business processes, users are interested in clustering process traces to discover per-cluster process models that are less complex. Such applications require the ability to measure the similarity between data objects. However, measuring the similarity between sequence-based data is computationally expensive. We present an intuitive and user-controlled approach to summarize sequence-based multi-dimensional data. Our summarization schemes provide a trade-off between the quality and efficiency of analysis tasks. We also derive an error model for summary-based similarity under an edit-distance constraint. Evaluation results over real-world datasets show the effectiveness of our methods.

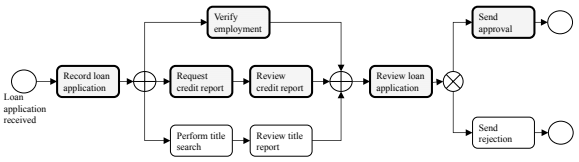
1. Introduction

Many application domains produce data in the form of multi-dimensional sequence of objects. For example, in business processes, an underlying process model is represented as a directed acyclic graph of activities, the traces generated from the execution of the model are regarded as instances of the underlying model. Each trace consists of a sequence of activities sorted by time, where each activity in the trace appears in the process model and may be repeated[1]. Figure 1 shows an example of a loan application process model, along with a sequence of activities—with multi-dimensional attributes—that represent a possible execution trace of the model. For example, an activity can contain information about the responsible person and department, the person who performs the activity, and the group to which she belongs. As an example from another domain, Figure 2 shows a sample trace of a semiconductor manufacturing workflow, where activities are sequenced and have multi-dimensional attributes, such as the sector where the activity is performed and the person responsible for it.

There is a need to get operational insights from such datasets. For example, in business process management, discovered models are often complex and difficult to comprehend [1], so users cluster process traces and applying process discovery algorithms [2] on each cluster. These latter models tend to be both less complex and more accurate since there is less diversity

PMAI@IJCAI22: International IJCAI Workshop on Process Management in the AI era, July 23, 2022, Vienna, Austria
pvnguye2@illinois.edu (P. Nguyen); vatchei@ibm.com (V. Isahagian); vmuthus@us.ibm.com (V. Muthusamy); aslom@us.ibm.com (A. Slominski)

[1] *Trace*, *process trace*, and *sequence* interchangeably refer to an instance of a process.



Timestamp,Responsible,Department,Trace,Resource,**Activity**,Group
09/22/15 10:16 AM,Resource21,General,Trace-11,Resource21,**Record loan application**,Group 1
09/26/15 08:10 AM,Resource21,General,Trace-11,Resource10,**Request credit report**,Group 4
10/01/15 03:05 PM,Resource21,General,Trace-11,Resource21,**Review credit report**,Group 1
10/15/15 10:00 AM,Resource21,General,Trace-11,Resource15,**Verify employment**,Group 2
10/20/15 12:30 PM, Resource21,General,Trace-11,Resource15,**Review loan application**,Group 1
10/31/15 04:30 PM,Resource21,General,Trace-11,Resource21,**Send approval**,Group 4

Figure 1: Loan application process and a sample trace.

Activity	Sector	Responsible
Pull wafers	CONTROL	A
Record Wafer IDs / Attach Wafer ID Map	CONTROL	A
Initiator Coordination	INIT ATTN	A
lamp degas in chamber E	METAL	B
sputter etch in chamber D	METAL	B
low stress 10 kW TaN - Rs monitor	METAL	B
lamp degas in chamber E	METAL	B
sputter etch in chamber D	METAL	B
low stress 10 kW TaN	METAL	B
Wafer transport to 1-2 from 7-2	DIEL	C
200C 1000W Lo OH oxide	DIEL	C
transport from 1-2 to 5-2	DIEL	C

Figure 2: Sample trace of a semiconductor manufacturing workflow.

among the traces within a cluster. In another example, scientists are interested in querying the provenance of workflow executions to look for executions similar to the one in their query.

Analyzing multi-dimensional sequence data poses a number of challenges. The first is computational complexity. For example, using edit-distance to capture the similarity between sequences [3] is computationally expensive since edit-distance is quadratic to the sequence length and business processes sequences in can consist of hundreds of items. This is especially challenging when dealing with large datasets and in applications such as traces clustering, where a lot of similarity computations need to be calculated. This complexity can lead to delays that affect interactive applications, such as similarity search, where users interact directly with the application and expect results in a timely manner. The second challenge is to combine multi-dimensional attributes of data with the sequential structure between data objects into a unified approach. Edit-distance, for example, only considers the number of operations to transform one trace into another.

We employ summarization schemes to enable efficient analysis of multi-dimensional data under edit-distance constraints. We focus on analysis tasks that are based on edit-distance because it is a widely used measure for similarity. Sections 2 and 3 introduce the key approach: instead of performing the analysis on the original high-dimensional data, which is computationally expensive, we transform the data into a summary or embedding space that has fewer dimensions, so that the same analysis can be computed more efficiently. Section 4 introduces our topic-summarization schemes to incorporate the multi-dimensional attributes of data items into the analysis and produce summaries that capture the semantics of process traces, while enabling the flexible trade-off between quality and efficiency of analysis tasks on summaries.In Section 5, we develop an error model for the edit-distance measure in the summary space to provide some guarantees for the results of analysis tasks on summaries. Finally, Section 6 shows the effectiveness of our summarization scheme on a number of datasets.

2. Trace Summarization Approach

We assume the existence of an original dataset that consists of a set of process traces or logs of workflow executions. Running an analysis, which would typically be computationally expensive

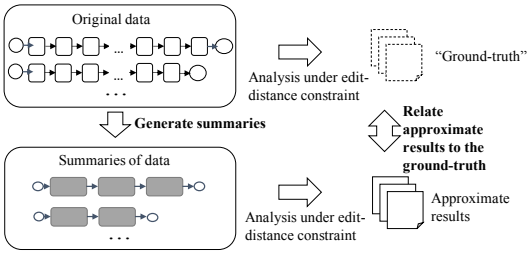


Figure 3: Overview of our approach.

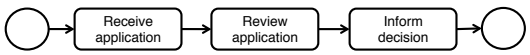


Figure 4: Topic-based representation of the process in Figure 1.

due to the high-dimensionality of the data, provides results which are deemed as exact or "ground truth" answer. As shown in Figure 3, our approach is to transform the original data into a new summary space with fewer dimensions, thus avoiding the computationally expensive analysis on original data. The output of any analysis in the summary space is an approximation of the "ground truth". To show the practicality of our proposed approach, we need to address the following challenges: (1) How to generate summaries of data in a controlled and intuitive manner, and (2) Relate the approximate results on summaries to the results on original data?

To address these two challenges, we define sequential-order-preserving summarization and introduce a summarization scheme that is intuitive and give users more control over the resulting summaries. We also present an error model for summary-based similarity measure under edit-distance constraint and show that it provides guarantees over the results of clustering and similarity search tasks.

3. Definitions

A multidimensional set $\mathscr{O}$ is a set of objects $\mathbb{O}$ and a set of associated attributes $\mathbb{A} = (\mathcal{A}_1, \mathcal{A}_2, ..., \mathcal{A}_{|\mathbb{A}|})$: $\mathscr{O} = \langle \mathbb{O}, \mathbb{A} \rangle$, each object $o \in \mathbb{O}$ is defined as a tuple: $o = (\mathcal{A}_1(o), \mathcal{A}_2(o), ..., \mathcal{A}_{|\mathbb{A}|}(o))$, in which each i-th dimension corresponds to the value of attribute $\mathcal{A}_i$ of o, denoted as $\mathcal{A}_i(o)$.

A *Multidimensional Sequence* $\mathbf{p}$ of size m on a multidimensional set $\mathscr{O}$ is defined as an ordered set of m objects in $\mathscr{O}$: $\mathbf{p} = (p_1, p_2, ..., p_m), p_i \in \mathscr{O}, 1 \leq i \leq m$. We denote $\iota_{\mathbf{p}}(p)$ as the *index*, or position, of an object p in a sequence $\mathbf{p}$. In the above definition, $\iota_{\mathbf{p}}(p_i) = i, \forall 1 \leq i \leq m$.

For example, Figure 2 presents a sequence of objects defined on a multidimensional set with three attributes: *Activity*, *Sector*, and *Responsible*.

Our interest is in different forms of summarization of multidimensional sequences to improve efficiency of sequence analysis. Before defining summarization of sequences, we define the notion of many-to-one mapping of objects between multidimensional sets as an object mapping function f from an original multidimensional set $\mathscr{O}$ to a summary set $\mathscr{S}$, $f : \mathscr{O} \rightarrow \mathscr{S}$, so that for each $p \in \mathscr{O}, \exists! s \in \mathscr{S} : s = f(p)$.

Definition 1. *A f-summarization of a sequence $\mathbf{p}$ on $\mathscr{O}$ is defined as a summary sequence $\mathbf{s}$ on $\mathscr{S}$, denoted as $\mathbf{s} = f(\mathbf{p})$, where each object $p \in \mathbf{p}$ is replaced by its many-to-one mapping f: $s = f(p)$, while retaining the same index $\iota_{\mathbf{s}}(s) := \iota_{\mathbf{p}}(p)$.*

A summarization of a sequence is said to preserve the sequential relationship from the original sequence if it satisfies the following definition:

Definition 2. *A f-summarization of a sequence $\mathbf{p}$, denoted as $\mathbf{s} = f(\mathbf{p})$, is a sequential preserving summarization of $\mathbf{p}$ if: $\forall p, p' \in \mathbf{p}$, if $\iota_{\mathbf{p}}(p) < \iota_{\mathbf{p}}(p')$, then $\iota_{\mathbf{s}}(s) \leq \iota_{\mathbf{s}}(s')$, with $s = f(p), s' = f(p')$.*

By retaining the indices of objects in the original sequence, f-summarization (*c.f, definition 1*) preserves sequential relationships, which is vital in improving the efficiency of sequence analysis. Therefore, we define the notion of *reduced f-summarization*, in which adjacent duplicate objects in the summary sequence are collapsed to reduce the size of a summarized sequence.

Definition 3. *A reduced f-summarization of a sequence $\mathbf{p}$ on $\mathscr{O}$ is defined as a sequence $\mathbf{s}$ on $\mathscr{S}$, denoted as $\mathbf{s} = f^*(\mathbf{p})$, where each object $p \in \mathbf{p}$ is replaced by its f-based mapping $s = f(p)$ in $\mathbf{s}$ and, $\forall p_i, p_{i+1} \in \mathbf{p}, 1 \leq i \leq |\mathbf{p}| - 1$, if $p_i = p_{i+1}$, then $\iota_{\mathbf{s}}(p_i) = \iota_{\mathbf{p}}(p_{i+1})$.*

Theorem 1. *A reduced f-summarization is sequence preserving.*

Proof. Omitted due to space constraints. Available in [4]. □

4. Topic-Based Summarization

To incorporate the multidimensional attributes of a sequence's data items, we begin by outlining Attribute-based summarization as an f-summarization[2] where f is a mapping on $\mathscr{O}$. This scheme provides an intuitive way for users to choose attributes as a summarization criteria and produces summaries that are easy to interpret. It does not give users control over the average length of summarized sequences, which we refer to as *resolution*. This is because attribute values are static and already defined with the original data.

Longer summarized sequences are more expensive to analyze, but attribute-based summarization offers little control in the sequence length. We seek a way for users to trade-off between efficiency and accuracy of data analysis. For example, for similarity search, users might tolerate false positives (e.g., 0.9 false positive rate) for faster response (e.g., results within 5 seconds). We observe that business processes can often be represented by higher-level process models of fewer dimensions. Figure 4 shows an example of a more abstract version of the process model in Figure 1, where each activity corresponds to multiple activities in Figure 1.

We propose a *topic-based summarization* technique that captures the many-to-one mapping from the original sequences to one with fewer dimensions, where each *topic* is an abstract representation of a set of original dimensions. Since the topics are implicit from the original sequences, we first perform dimensionality reduction on the original sequences to transform the original dimensions to topics. Then, we define the notion of topic-based summarization using the new representation.

Algorithm 1 highlights the main steps in the topic-based summarization process. Before applying dimension reduction techniques to the original sequences (Line 3), it is important to have an appropriate data representation for sequences (Line 2). We begin by selecting an attribute of the original sequences and transform multidimensional sequences to the appropriate attribute-based summarization. It is often intuitive to pick the attribute with the most number

[2]Unless explicitly stated, a summarization will refer to reduced summarization.

Algorithm 1 Topic summarization process steps

1: **procedure** GENERATEKTOPICSUMMARIZATION($\mathbf{S}$, κ, $\mathscr{O}$, λ)
2: $\mathbf{M}$ = Generate_Vectors($\mathbf{S}$) (by Equation 1)
3: $\mathbf{M}'$, $\mathbf{W}$ = Dim_Reduction($\mathbf{M}$, κ)
4: *// Calculate pairwise similarities* $\theta(a_i, a_j)$
5: **for** each pair $(a_i, a_j) \in \mathscr{O}$ **do**
6: $\mathcal{S}_{ij} = CalcSim(a_i, a_j)$
7: *// Perform hierarchical clustering*
8: $\mathcal{H} = \texttt{hierarchical\_clustering}(\mathcal{S})$
9: *// Flatten the hierarchy* $\mathcal{H}$
10: $\mathbb{C} = \texttt{flatten\_hierarchy}(\mathcal{H}, \kappa)$
11: return $\mathbb{C}$

of dimensions as this attribute likely captures the most essential information about the objects in the original multidimensional set. For example, in Figure 2, *Activity* is the attribute with the most number of dimensions and it is also the base attribute to represent sequences, while other attributes, such as *Sector* and *Responsible*, provide supporting information for *Activity*.

We then represent each sequence $\mathbf{p}$ as a numeric vector $(\vartheta_1, \vartheta_2, ..., \vartheta_{|\mathcal{A}^*|})$, where $\mathcal{A}^*$ is the base attribute set that sequences are transformed to in the first step and $|\mathcal{A}^*|$ is the number of dimensions on $\mathcal{A}^*$. We measure ϑ_i for $\mathbf{p}$ in a way that captures both the local importance of each dimension and its specificity to a sequence. To capture the local importance, we use the frequency of the i-th dimension in $\mathbf{p}$, denoted as $\texttt{tf}^i_{\mathbf{p}}$, that is defined by the number of items in $\mathbf{p}$ whose values equal the i-th dimension of $\mathcal{A}^*$, denoted as a_i. To capture the specificity, we use the popularity of a dimension across all sequences: $\texttt{df}_i = |\{\mathbf{p} \in \mathbb{S} | a_i \in \mathbf{p}\}|$, where $\mathbb{S}$ is the set of all sequences. Intuitively, the higher $\texttt{df}_i$ is, the more popular the i-th dimension is and thus, the less specificity it is to a sequence. The formulation of ϑ_i is as follows:

$$\vartheta_i = \begin{cases} (1 + log(\texttt{tf}^i_{\mathbf{p}})) \times log(\frac{|\mathbb{S}|}{\texttt{df}_i}) & \text{if } a_i \in \mathbf{p} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

After representing sequences as vectors, the set of sequences $\mathbb{S}$ can be represented as a matrix $\mathbf{M}$, whose size is $|\mathbb{S}| \times |\mathcal{A}^*|$ where each row corresponds to a vector representation of a sequence in $\mathbb{S}$. With this matrix representation, we can apply off-the-shelf dimension reduction techniques on $\mathbf{M}$, such as non-negative matrix factorization (NMF), principle component analysis (PCA), or singular value decomposition (SVD), among others (Line 3). The results of these techniques can be presented as two matrices $\mathbf{M}'$ and $\mathbf{W}$. $\mathbf{M}$, whose size equals $|\mathbb{S}| \times k$ with k being the number of new dimensions (i.e., $k = |\mathscr{S}|$), represents the original sequences on the summary space. $\mathbf{W}$, whose size equals $|\mathscr{O}| \times k$, represents the original dimensions on the new dimensions, or topics (i.e., each row is a vector representing the distribution of an original dimension over the set of new dimensions).

After dimensionality reduction, we produce a many-to-one mapping from the original dimensions to topics (Line 6). Two dimensions a_i, a_j in the original space are likely to be in the same topic if their corresponding vectors in $\mathbf{W}$ have high similarity (e.g., using Cosine similarity). In addition, a_i and a_j are likely to be in the same topic if they frequently appear next to each other in a sequence (i.e., they represent two closely related activities in the underlying process model). From these insights, we model the problem of finding an optimal many-to-one mapping

from the original dimensions to topics as a constrained optimization problem:

$$\begin{aligned} \underset{f}{\text{argmax}} \quad & \lambda \cdot \sum_{f(a_i)=f(a_j)} \theta(a_i, a_j) + (1-\lambda) \cdot \sum_{(a_i,a_j)} \omega(a_i, a_j)\theta(a_i, a_j) \\ \text{subject to} \quad & f : \mathscr{O} \to \mathscr{S} \\ & \forall a_i, a_j \in \mathscr{O}, \text{if } f(a_i) \neq f(a_j), \text{then } a_i \neq a_j. \\ & |\mathscr{S}| = k. \end{aligned} \tag{2}$$

where $\theta(a_i, a_j)$ is the similarity between dimensions a_i and a_j based on their corresponding representation in $\mathbf{W}$, $\omega(a_i, a_j)$ is the number of times a_i and a_j are adjacent in input sequence set $\mathbb{S}$, and λ is used to bias towards similarity between dimensions or the number of adjacent appearances. We now can formally define the notion of topic summarization as follows:

Definition 4. *(k-**Topic Summarization**) A k-topic summarization of sequences from original multidimensional set $\mathscr{O}$ to a summary set $\mathscr{S}$ is defined as a reduced f-summarization, where the mapping f is the solution of the optimization problem defined in (2).*

Finding an optimal k-topic summarization is NP-hard (a variant of the set partitioning problem). We take a greedy heuristic approach similar to the agglomerative clustering algorithm (Line 8). It starts by treating each original dimension as a singleton cluster, then merging nearby pairs of dimensions until all clusters have been merged into a single cluster. This step creates a hierarchy where each leaf node is a dimension and the root is the single cluster of the last merge. Because we want a partition of disjoint k clusters as the new dimensions, the next step is to cut the hierarchy at some point to obtain the desirable number of clusters. To find the cut (Line 10), we find the minimum similarity threshold so that the distance between any two dimensions in the same cluster is no more than that threshold and there are at most k clusters.

5. Error Model for Edit-Distance on Summaries

We seek to relate the approximate results of analysis tasks on the summary space to those on the original space. Since a similarity measure underlies a lot of analysis tasks, such as similarity search and traces clustering, we focus on the relationship between the similarity of sequences on the summary space with that on the original space under edit-distance constraint: $\texttt{ed}(\mathbf{p}, \mathbf{q})$ & $\texttt{ed}(f(\mathbf{p}), f(\mathbf{q}))$, where ed is the edit-distance function and f is a summarization function. We select edit-distance as the similarity measure because it captures both the structural similarity (i.e., whether two sequences consist of data items in similar order) and content-based similarity (i.e., whether two sequences share similar set of data items) between sequences. Furthermore, edit-distance's results, presented as a chain of edit operators to transform a sequence to the other, can be easily interpreted by users, which makes it widely popular in practice.

In terms of the relationship between $\texttt{ed}(\mathbf{p}, \mathbf{q})$ and $\texttt{ed}(f(\mathbf{p}), f(\mathbf{q}))$, we are interested in the contractive property.

Definition 5. *Given a summarization f, we said that the edit-distance measure satisfies the contractive property on f if* $\texttt{ed}(\mathbf{p}, \mathbf{q}) \geq \texttt{ed}(f(\mathbf{p}), f(\mathbf{q})), \forall \mathbf{p}, \mathbf{q}$.

The contractive property guarantees that performing edit-distance based similarity search on the summary space using f will yield results with 100% recall [5]. Specifically, given

a query sequence $\mathbf{p}$ and an edit-distance threshold χ, the similarity search task needs to find all sequences in the sequence set $\mathbb{S}$ that have edit-distance with $\mathbf{p}$ smaller or equal than χ: $\mathbb{S}^* = \{\mathbf{q} \in \mathbb{S} | \texttt{ed}(\mathbf{p}, \mathbf{q}) \leq \chi\}$. If the contractive property holds for a summarization f, it is sufficient to find all sequences $\mathbf{q}$ that satisfy the threshold χ on the summary space: $\bar{\mathbb{S}} = \{\mathbf{q} \in \mathbb{S} | \texttt{ed}(f(\mathbf{p}), f(\mathbf{q})) \leq \chi\}$. Because if $\texttt{ed}(\mathbf{p}, \mathbf{q}) \leq \chi$, then $\texttt{ed}(f(\mathbf{p}), f(\mathbf{q})) \leq \chi$; we can guarantee that if $\mathbf{q} \in \mathbb{S}^*$, then $\mathbf{q} \in \bar{\mathbb{S}}$ (i.e., 100% recall).

While the contractive property does not hold *in general* for edit-distance between summarized sequences, we show that it holds under certain circumstances. The first of which is when f is a *non-reduced many-to-one*.

Theorem 2. *If f is a non-reduced many-to-one summarization on $\mathscr{O}$, as defined in definition 1, then we have:* $\texttt{ed}(\mathbf{p}, \mathbf{q}) \geq \texttt{ed}(f(\mathbf{p}), f(\mathbf{q}))$, $\forall \mathbf{p}, \mathbf{q}$ *on* $\mathscr{O}$.

Proof. Omitted due to space constraints. Available in [4]. □

For *reduced many-to-one summarization* f, we are able to derive rules to indicate whether the contractive property holds for edit-distance of a particular pair of sequences $\mathbf{p}, \mathbf{q}$.

Theorem 3. *Given two sequences $\mathbf{p}, \mathbf{q}$ in the original space $\mathscr{O}$, if f is a reduced many-to-one summarization on $\mathscr{O}$, as defined in definition 3, then:*

- *If $\Gamma_{\mathbf{p},\mathbf{q}} \geq \Lambda_{f(\mathbf{p}),f(\mathbf{q})}$, then we have $\texttt{ed}(\mathbf{p}, \mathbf{q}) \geq \texttt{ed}(f(\mathbf{p}), f(\mathbf{q}))$; or edit-distance on summary space by f **satisfies** the contractive property.*
- *If $\Gamma_{f(\mathbf{p}),f(\mathbf{q})} > \Lambda_{\mathbf{p},\mathbf{q}}$, then we have $\texttt{ed}(\mathbf{p}, \mathbf{q}) < \texttt{ed}(f(\mathbf{p}), f(\mathbf{q}))$; or edit-distance on summary space by f **does not** satisfy the contractive property.*

where $\Lambda_{\mathbf{p},\mathbf{q}} = max(|\mathbf{p}|, |\mathbf{q}|)$ and $\Gamma_{\mathbf{p},\mathbf{q}} = ||\mathbf{p}| - |\mathbf{q}||$, with $|\mathbf{p}|$ being the length of $\mathbf{p}$.

Proof. Omitted due to space constraints. Available in [4]. □

While Theorem 3 does not cover all cases, we empirically show that the number of sequence pairs whose edit-distances on reduced many-to-one summarization that violate the contractive property is very small. Thus, it has a high recall for similarity search task.

6. Evaluation

We evaluate the effectiveness and efficiency of our summarization schemes on two analysis tasks: trace similarity search and traces clustering.

Datasets: We use datasets from multiple domains: the `Lithography` dataset (596 traces with 1066 types of activities, each having multi-dimensional attributes) is from a real semiconductor manufacturing process, the `BPIC` 2015 dataset (1199 traces with 289 activity types) is from a building permit application process, and the `BANK` dataset (2000 traces with 113 activity types) consists of synthetically generated logs from a large bank transaction process. Evaluations were conducted on a 2.7GHz quad-core Intel Core i7 machine with 16GB of RAM [3].

[3] Results for `BPIC` experiments are available at [4]. The `Lithography` dataset is production dataset provided by IBM and is private. Other datasets is available at https://data.4tu.nl/repository/collection:all.

	k=2	k=5	k=10	k=20	k=50	k=100
Topic	0.000%	0.003%	0.006%	0.007%	0.010%	0.014%
Random	0.002%	0.010%	0.007%	0.021%	0.027%	0.033%

Figure 5: Similarity false negatives: percentage of sequence pairs in the `Lithography` dataset where edit-distance in the summary space violates the contractive property.

Summarization schemes: We compare results of analysis tasks using our proposed summarization schemes (i.e., `Topic` and `Attribute`), `Random` summarization, which randomly maps an original dimension to a new dimension in the summary space, and with the analysis results on the original space. Although $Random$-based summaries lack interpretability, as shown in [6], a random summarization scheme on sequence graph can yield good results. We vary the number of dimensions k in the summary space used by `Random` and `Topic` and vary the attributes used by `Attribute`.

6.1. Evaluation Results on Similarity Tasks

The contractive property holds for most of the cases, as seen in Figure 5 which shows the percentage of sequence pairs in the `Lithography` dataset, out of over 177,000 pairs, whose edit-distances violate the contractive property in the summary space using $Topic$ and $Random$ summarization over different number of summary dimensions k. Since the recall rate is high, we focus on the false positive rate of the similarity search results.

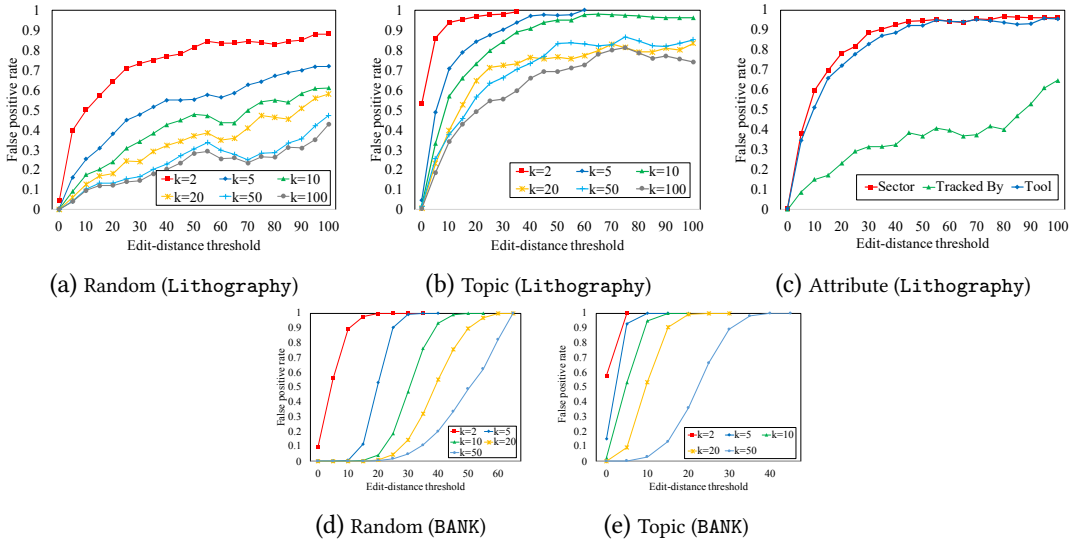
Evaluation metrics: Given an edit distance threshold χ, the false positive metric tells us that, out of all sequence pairs that satisfy $\texttt{ed}(f(\mathbf{p}), f(\mathbf{q})) \leq \chi$ on the summary space, how many of them actually satisfy the threshold in the original space: $\texttt{ed}(\mathbf{p}, \mathbf{q}) \leq \chi$.

Effectiveness: Figure 6 shows the effectiveness of the summarization schemes on the similarity search task for the `Lithography`, and `BANK` datasets[4]. The y-axis reports the false positive results, while the x-axis corresponds to different edit-distance thresholds. As expected (Figure 6a, 6b, 6d, 6e), the higher the number of dimensions in the summary space (denoted by k), the better the result (i.e., lower false positive rates). That is because, with more dimensions in the summary space, summaries of sequences more resemble the original sequences. Thus, there is little difference between edit-distances on the summary space and in the original space.

Comparing the summarization schemes on the same number of dimensions, `Random` outperforms `Topic` (at the cost of interpretability and efficiency, as we will show later). For `Attribute` (Figure 6c), since we cannot control the number of dimensions (as it depends on the attribute data), the quality of the results also depend on the chosen attribute. Specifically, the $TrackedBy$[5] attribute outperforms $Sector$ and $Tool$. This is in part because there are more dimensions on $TrackedBy$'s summary space, and thus the summaries on the $TrackedBy$ space more resemble the original sequences. $Sector$ and $Tool$ produce similar results, since similar $Tool$s are often used in the same $Sector$.

[4]We only evaluate `Attribute` summarization on the `Lithography` dataset because this dataset's attributes provide better semantics compared with `BANK`.

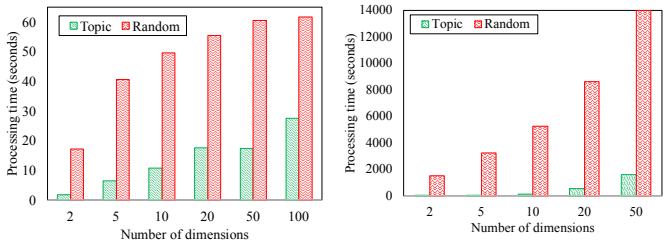
[5]Three main activity attributes are used on the `Lithography` data: $TrackedBy$ represents the person in charged of the activity; $Sector$ represents the area/department where the activity is taken, and $Tool$ represents the tool used to perform the activity.



(a) Random (Lithography) (b) Topic (Lithography) (c) Attribute (Lithography)

(d) Random (BANK) (e) Topic (BANK)

Figure 6: False positive rates by different summarization schemes on similarity search task using the Lithography, and BANK datasets.



(a) (Lithography) (b) (Bank)

Figure 7: Efficiency comparison of processing time between Random and Topic summarizations using the Lithography, and Bank datasets.

Efficiency: To evaluate the efficiency of the summarization schemes, we vary the number of dimensions k in the summary space and measure the time to calculate the edit-distance between all pairs of sequences. We see in Figure 7, that for both Random and Topic, larger k, which leads to longer longer sequences in the summary space, results in longer processing time. For similar values of k, Topic outperforms Random, which verifies Topic's ability to capture the semantic relationship between the original dimensions, and thus significantly reduces the size of sequences in the summary space, as well as the processing time. More importantly, even at different values of k where we observed similar effectiveness of results by Random and Topic (e.g., $k = 2$ with Random and $k = 10$ with Topic on the Lithography dataset in Figure 6), Topic is still much more efficient than Random.

6.2. Evaluation Results on Traces Clustering

Evaluation metrics: We evaluate the clustering results using process-specific metrics [3]: weighted average conformance fitness, and weighted average structure complexity. While

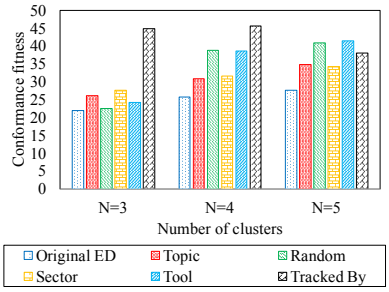


Figure 8: Conformance fitness comparison.

Approach	N=3			N=4			N=5		
	Arcs	Places	Trans.	Arcs	Places	Trans.	Arcs	Places	Trans.
Original ED	3930	**1505**	1964	3419	1328	1709	3508	**1356**	1753
Topic	3855	1474	1927	3261	1269	1630	2685	1061	1342
Random	**3959**	1500	**1979**	**3733**	**1418**	**1866**	**3552**	1351	**1775**
Sector	3697	1429	1848	3043	1195	1521	2775	1094	1387
Tool	3722	1441	1860	**2792**	**1110**	**1396**	2758	1104	1379
Tracked By	**3482**	**1357**	**1741**	2827	1121	1413	**2650**	**1051**	**1325**

Figure 9: Traces clustering results' structural complexity comparison. (Green and red boxes denote best and worst results, respectively.)

the process model's conformance fitness quantifies the extent to which the discovered model can accurately reproduce the recorded traces, the structure complexity quantifies whether the clustering results produce process models that are simple and compact. Given a summarization scheme, we first transform all sequences to the summary space, and then perform traces clustering (using hierarchical clustering) with edit-distance as the similarity measure. Then, a process model is generated for each cluster using the Heuristic mining algorithm [7] and then converted to the Petri-Net model for conformance analysis. Given the Petri-net model, we use two publicly available plugins from the ProM framework [8] for fitness and structural complexity analysis: The Conformance Checker Plugin is used to measure the fitness of the generated process models and the Petri-Net Complexity Analysis Plugin is used to analyze the structural complexity of the process models. After fitness and complexity scores are calculated for each cluster, the final scores are calculated as the average score over all clusters, weighted by the cluster size.

Effectiveness of summarization schemes: Figure 8 highlights the *conformance fitness* of the clustering results in the summary space by different summarization schemes[6] on the `Lithography` dataset. Surprisingly, using summarization schemes not only helps improve the efficiency of the clustering task (as we showed earlier in the efficiency evaluation), but also helps produce clusters with process models of higher fitness, compared with the clustering results in the original space. The trend is similar when varying the number of clusters N. That is because measuring trace similarity on the summary space helps remove noise that often exists when measuring similarity using the original representation. Among summarization schemes, `Attribute` helps produce clustering results of higher conformance fitness (especially when using the $TrackedBy$ attribute). That is because `Attribute` summarizations capture better the semantic relationship between traces (e.g., traces are similar if the corresponding sequences of $Sector$, $Tool$, or $TrackedBy$ are similar).

In terms of the structural complexity (Figure 9), `Attribute` summarizations outperform other summarization schemes, again due to its ability to capture semantic relationships between traces, producing clusters whose process models capture traces with similar semantics, and thereby having simpler model structures. On the other hand, `Random` , unable to capture the semantic relationships between traces, is the worst performer.

[6]We use $k = 2$ for `Random`, and $k = 20$ for `Topic`, as these are similarly effective for similarity search.

In both conformance fitness and structural complexity tests, `Topic` summarization approaches `Attribute`. Unlike `Attribute` summarization, which does not give users control over the resolution of the summaries, `Topic` summarization provides a qualitative advantage in offering a tunable parameter, k, to trade-off between the effectiveness and efficiency in the analysis task.

7. Related Work

Subsequence mapping and sequence retrieval is an active area of research. One common approach is to summarize original sequences using q-grams [9, 10] and measure the similarity between two sets of q-grams. DRESS [9] uses the most frequent codewords as references to identify a set candidate matches of a query. MinSearch [10] partitions strings into a hierarchy of substrings and builds an index comprised of a set of hash tables, so that strings having common substrings and thus small edit distance are grouped into the same hash table. These methods do not preserve the sequential relationship between data items from the original sequences, and do not consider sequences of multi-dimensional attributes of each data item.

Graph similarity and mining focuses on transforming the original graph – based on graph substructures *e.g.* trees [11], branches [12] – to a compact representation before measuring similarity. Recent techniques [13] make use of disjoint substructures of graphs to capture structural differences between graphs. Theses graphs lose their representation and interpretability after being transformed into substructure representation.

Embedding methods [14][15][16] improve efficiency of similarity search on complex data. Few of the embedding approaches guarantee properties of similarity measure on the embedding space, such as contractive property. For example, it may require that the similarity measure between data on the embedding space to be from a specific family of measure (e.g., Minkowski metric). Furthermore, techniques that transform original sequences into vector-based representation do not maintain the sequential relationship between data items on the new representation.

There has been a significant amount of research on various topics related to *graph summarization*. We refer the reader to the following surveys [17, 18]. OLAP [18] enables interpretable summaries of original graph at various resolutions as aggregate graphs. Chen et al. [6] show that random summaries are capable of mining frequent graph patterns and effectively reduce the size of original graph. In this work, besides using explicit attributes, we leverage the implicit topics as summarization criteria. We also show that, different from general graphs, random summarization on sequences, although produces good effectiveness, suffers from efficiency.

Efforts to address *scalability issues* in business process analysis focuses either on process model discovery of complex traces [19], or the use of vector space-based dimensional reduction to improve the performance of traces clustering [20]. Our focus is on improving efficiency of traces clustering and similarity search under edit-distance constraint.

8. Conclusions

We introduce a method to perform efficient analysis on sequence-based multi-dimensional data using intuitive and user-controlled summarizations. We define a topic summarization scheme that offer flexible trade-off between quality and efficiency of analysis tasks and derive an error

model for summary-based similarity under an edit-distance constraint. The approach was found to be both effective and efficient based on evaluations on real-world process datasets.

References

[1] A. K. A. De Medeiros, et al., Process mining based on clustering: A quest for precision, in: BPM, 2007.

[2] W. Van der Aalst, et al., Workflow mining: Discovering process models from event logs, TKDE (2004).

[3] J. Bose, et al., Context aware trace clustering: Towards improving process mining results, in: SDM, 2009.

[4] P. Nguyen, et al., Summarized: Efficient framework for analyzing multidimensional process traces under edit-distance constraint, arXiv:1905.00983 (2019).

[5] P. Papapetrou, et al., Reference-based alignment in large sequence databases, VLDB (2009).

[6] C. Chen, et al., Mining graph patterns efficiently via randomized summaries, VLDB (2009).

[7] A. Weijters, W. M. van Der Aalst, A. A. De Medeiros, Process mining with the heuristics miner-algorithm, Technische Universiteit Eindhoven, Tech. Rep. WP (2006).

[8] B. F. Van Dongen, et al., The prom framework: A new era in process mining tool support, in: Conference on Application and Theory of Petri Nets, 2005.

[9] A. Kotsifakos, et al., Dress: dimensionality reduction for efficient sequence search, KDD (2015).

[10] H. Zhang, Q. Zhang, Minsearch: An efficient algorithm for similarity search under edit distance, in: KDD, 2020.

[11] W. Zheng, et al., Graph similarity search with edit distance constraint in large graph databases, in: CIKM, 2013.

[12] Z. Li, et al., An efficient probabilistic approach for graph similarity search, in: ICDE, 2018.

[13] J. Kim, D.-H. Choi, C. Li, Inves: Incremental partitioning-based verification for graph similarity search., in: EDBT, 2019, pp. 229–240.

[14] M. Espadoto, et al., Toward a quantitative survey of dimension reduction techniques, IEEE transactions on visualization and computer graphics (2019).

[15] C. Faloutsos, K.-I. Lin, FastMap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets, 1995.

[16] J. T.-L. Wang, et al., Metricmap: an embedding technique for processing distance-based queries in metric spaces, IEEE Transactions on Systems, Man, and Cybernetics (2005).

[17] Y. Liu, et al., Graph summarization methods and applications: A survey, ACM (CSUR) (2018).

[18] a. o. Queiroz-Sousa, A review on olap technologies applied to information networks, ACM Transactions on Knowledge Discovery from Data (TKDD) 14 (2019) 1–25.

[19] S. J. Leemans, et al., Scalable process discovery with guarantees, in: Conference on Enterprise, Business-Process and Information Systems Modeling, 2015.

[20] M. Song, et al., A comparative study of dimensionality reduction techniques to enhance trace clustering performances, Expert Systems with Applications (2013).