`Artemis`: Assessing the similarity of event-interval sequences

Orestis Kostakis Panagiotis Papapetrou Jaakko Hollmén

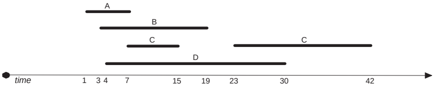
Labs, F-Secure Corporation

Department of Information and Computer Science, Aalto University.

September 6, 2011

Motivation

So far, previous work has focused on the knowledge discovery aspect.



Benefits of comparing:

- existence of a sequence in DB,
- new index structures,
- typical DM tasks,
- recommendation systems.

Background

Event-interval sequences

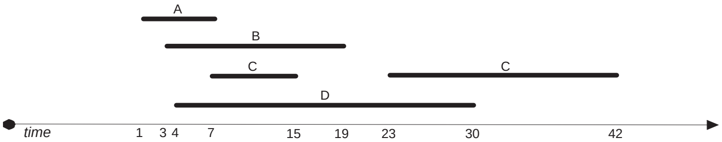


Figure: Size: 5, $\sigma = \{A, B, C, D\}$, $\{(A, 1, 7)$, $(B, 3, 19)$, $(D, 4, 30)$, $(C, 7, 15)$, $(C, 23, 42)\}$.

Distance Functions

Problem Formulation

Given two e-sequences $\mathcal{S}$ and $\mathcal{T}$, define a distance measure D, such that $\forall \mathcal{S}, \mathcal{T}$:

$$
\begin{aligned}
D(\mathcal{S}, \mathcal{T}) &\geq 0 &\quad (1)\\
D(\mathcal{S}, \mathcal{S}) &= 0 &\quad (2)\\
D(\mathcal{S}, \mathcal{T}) &= D(\mathcal{T}, \mathcal{S}) &\quad (3)
\end{aligned}
$$

Reducing to known problems

Sequences of instantaneous events do not depict all the important information:

Problem: Transforming the above arrangements to sequences of instantaneous events would yield the same result: $A_{start}, A_{end}, B_{start}, B_{end}$.

Solution: For each time-point, we must create an *event-vector* which records the number of occurrences of intervals for each label.

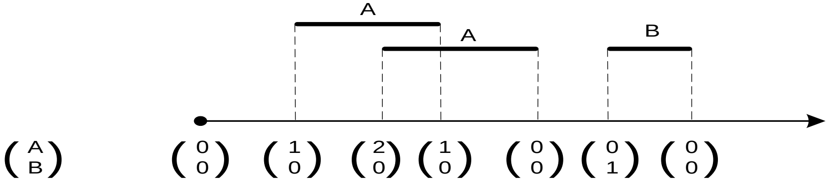


Figure: Encoding arrangements via event-vectors.

Bags of event-vectors can be handled as multi-dimensional time-series. Hence, *Dynamic Time Warping* (DTW) is applicable!

Problem: Vector-based DTW violates the *identity of indiscernibles* (aka Leibniz's law, $\mathcal{A} \neq \mathcal{B} \implies D(\mathcal{A}, \mathcal{B}) > 0$).

The event-vector multisets are: $\{(0), (1), (2), (1), (0)\}$, $\sigma = \{A\}$

Our approach

Focus on the relations between pairs of intervals.

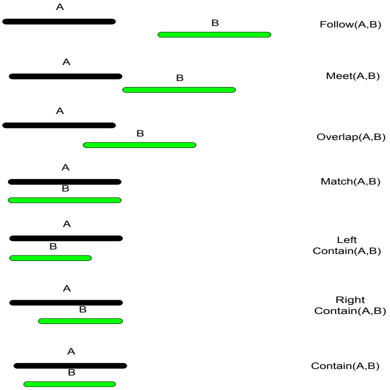
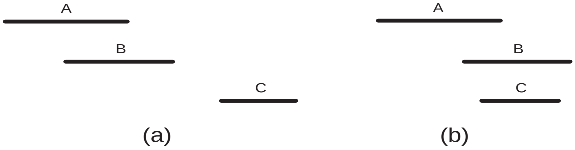


Figure: The relations that we consider; based on Allen's temporal model. Allen, J. F. , 'Maintaining knowledge about temporal intervals', *Communications of the ACM*.

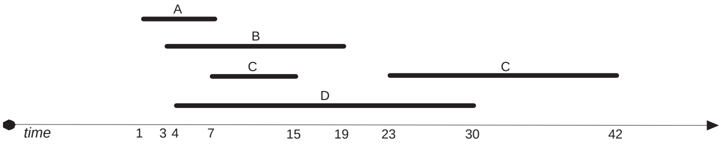
Artemis

Idea: Attempt to find 'corresponding' intervals. Then, derive the overall distance based on the corresponding pairs.



Mapping step: Map each interval to sets of relations.
Matching step: Calculate all pairwise scores. Apply minimum-weight maximum bipartite matching.

Artemis' Mapping step



For $S_i \in \mathcal{S}$ and $S_j \in \mathcal{S}, \forall j \neq i$ in the **same** e-sequence, compute:

- $r_{left}(S_i) = \{r(S_j, S_i) | 1 \leq j < i\}$
- $r_{right}(S_i) = \{r(S_i, S_j) | i < j \leq |S|\}$
- $r_{\varnothing}(S_i) = \{r(\varnothing, S_i)\}$

Additionally: $r_{\varnothing left}(S_i) = r_{left}(S_i) \cup r_{\varnothing}(S_i)$.

Artemis' Matching step

$$Artemis(\mathcal{S}, \mathcal{T}) = \sum_{i=1}^{\max\{|\mathcal{S}|,|\mathcal{T}|\}} d_m(S_i, h(S_i)), \quad S_i \in \mathcal{S}, h(S_i) \in \mathcal{T}.$$

based on the matching h returned by the Hungarian Algorithm, where the interval distances are:

$$d_m(S_i, T_j) = \begin{cases} 1 - \dfrac{|r_{\varnothing left}(S_i) \cap r_{\varnothing left}(T_j)| - |r_{right}(S_i) \cap r_{right}(T_j)|}{\max\{|\mathcal{S}|, |\mathcal{T}|\}}, & \text{if} E_{S_i} = E_{T_j} \\ 1, & \text{if} E_{S_i} \neq E_{T_j} \end{cases}$$

Artemis overview

Problem solved: `Artemis` does not violate the identity of indiscernibles. (Proof is trivial, omitted)

New problem: `Artemis` $\in O(n^3)$, prohibitive for large databases.

New target: Devise a fast lower bounding technique.

Linear-time lower bound for `Artemis`

Given an e-sequences $\mathcal{S}$, we define an $|\sigma|$-dimensional vector $v^{\mathcal{S}}$, that stores, for each event label in σ, the count of event-intervals in $\mathcal{S}$ that share that label.

Theorem

Given $\mathcal{S}$ and $\mathcal{T}$, the lower bound of $Artemis(\mathcal{S}, \mathcal{T})$ is defined as:

$$Artemis_{LB}(\mathcal{S}, \mathcal{T}) = \frac{k}{2} + \left(m - \frac{k}{2}\right)\left(\frac{k}{2m}\right) = k - \frac{k^2}{4m}, \tag{4}$$

where $k = ||v^{\mathcal{S}} - v^{\mathcal{T}}||_1$ and $m = max(|\mathcal{S}|, |\mathcal{T}|)$.

Experimental Setup

Datasets:

- **American Sign Language**
- **Pioneer1** robot sensor data
- **Hepatitis**

Dataset	# of e-sequences	# of intervals	e-sequence size min.	max.	average	$\lvert\sigma\rvert$	# of classes
ASL	873	15675	4	41	18	216	5
Pioneer	160	8949	36	89	56	92	3
Hepatitis	498	53921	15	592	108	147	2

Experimental Setup cnt

Experiments:

A. k-Nearest Neighbor classification
B. Detect identical phrases (ASL dataset)
C. Noise robustness
D. Scalability

In addition, the lower bound was tested for its *tightness*, and its *pruning power* during 1-NN queries

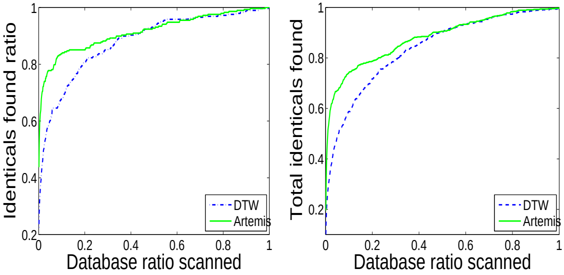
Experiments: k-NN Classification

Dataset	`Artemis` 1-NN	`Artemis` 3-NN	DTW 1-NN	DTW 3-NN
HepData	0.72	0.78	0.74	0.80
Pioneer	0.97	0.97	0.93	0.93
ASL	0.43	0.40	0.43	0.41

Table: *k*-NN classification results.

Conclusion: The results depend on how the class label is encoded into the sequences.

Experiments: Detect identical phrases in ASL dataset.



(a) Identical phrase found within k-NN

(b) Ratio of total identical phrases found within k-NN

Experiments: Noise robustness

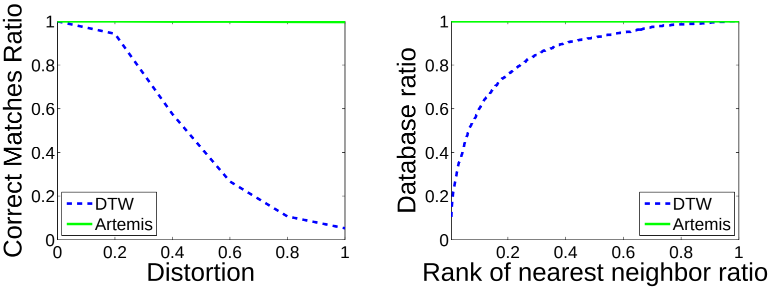
Two types of artificial noise:

- A. Shifts of intervals back or forth.
- B. Swaps of interval labels.

The two methods were compared in terms of:

- A. nearest neighbor retrieval accuracy
- B. rank of nearest neighbor

Results: Noise robustness

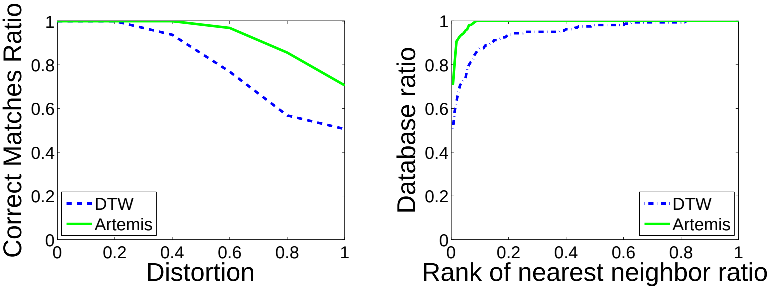


(c) Retrieval Accuracy

(d) Rank of NN. Distortion prob: 1.0, Offset limit: 1.0

Figure: ASL dataset, 'offset' noise.

Results: Noise robustness



(a) Retrieval Accuracy (b) Rank of NN. Swaps prob: 1.0

Figure: Pioneer dataset, 'swaps' noise.

Experiments: Scalability

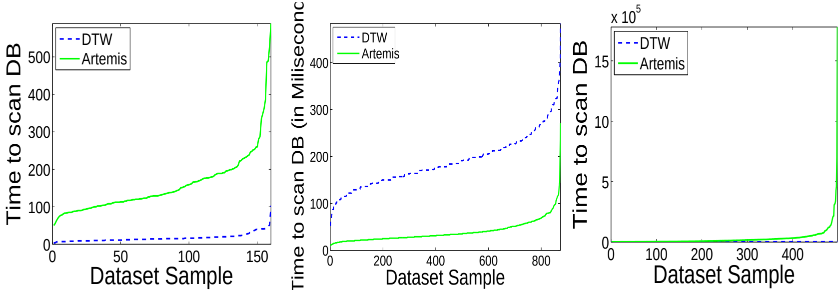
Complexity of each method:

- Vector-based DTW: $O(n \cdot m \cdot |\sigma|)$
- `Artemis`: $O(m^3)$ using hashing, our implementation: $O(m^4)$,

where $n = |\mathcal{A}|$, $m = |\mathcal{B}|$ $(m > n)$.

Time includes transforming each sample in the appropriate form (i.e. bag of event vectors, relation sets) and searching the DB. The samples DB is already in the appropriate form.

Results: Scalability



(a) Pioneer dataset (b) ASL dataset (c) Hepatitis dataset

Experiments: $\texttt{Artemis}_{LB}$

$$\text{Tightness} = \frac{\text{Artemis}_{LB}(\mathcal{S}, \mathcal{T})}{\text{Artemis}(\mathcal{S}, \mathcal{T})} \in [0, 1]$$

Dataset	LB Tightness	1-NN pruning power
ASL	0.8837	0.7931
Hepatitis	0.7166	0.7012
Pioneer	0.6189	0.4855

Table: Lower Bound tightness and pruning power.

Conclusions

- A. We presented 2 methods for comparing event-intervals sequences.
- B. No clear choice for clustering e-sequences. Choice must be application dependent.
- C. `Artemis` is most noise-robust. DTW very fragile.
- D. Promising lower bounding technique.

Directions for future work

- Devise faster distance functions that are metric.
- Determine if `Artemis` satisfies the triangular inequality.
- Devise tighter constant-/linear-time lower bounds for `Artemis`.
- Devise algorithms for on-line comparison of e-sequences.