

# Big Sequence Management: on Scalability

Karima Echihabi, Kostas Zoumpalifanos, Themis Palpanas  
Mohammed V University, Harvard University & Université de Paris, Université de Paris & French University Institute (UIF)

IEEE International Conference on Big Data (IEEE BigData), December 2020



1

## Acknowledgements

- thanks for slides to
  - Michail Vlachos
  - Eamonn Keogh
  - Panagiotis Papapetrou
  - George Kollios
  - Dimitrios Gunopoulos
  - Christos Faloutsos
  - Panos Karras

BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

2

## Introduction, Motivation

BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

3

## Data series

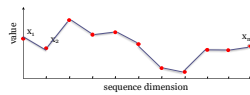
cifn 4

BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

4

## Data series

- Sequence of points ordered along some dimension

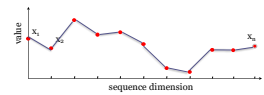


BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

5

## Data series

- Sequence of points ordered along some dimension

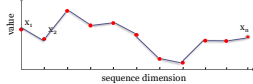


BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

6

## Data series

- Sequence of points ordered along some dimension



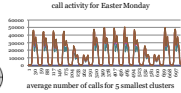
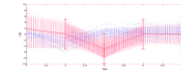
cifn 7

BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

7

## Telecommunications

- analysis of call activity patterns
- Telecom Italia

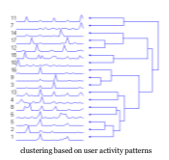
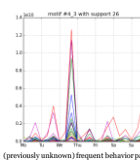


BigData, Zoumpalifanos, Palpanas - 1000 BigData 2020

8

## Home Networks

- temporal usage behavior analysis of home networks
- Portugal Telecom



9

Data Centers

- cloud utilization/operation/health monitoring



10

Neuroscience

- functional Resonance Magnetic Imaging (fMRI) data
  - primary experimental tool of neuroscientists
  - reveal how different parts of brain respond to stimuli



11

Neuroscience

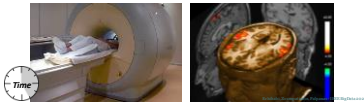
- functional Resonance Magnetic Imaging (fMRI) data
  - primary experimental tool of neuroscientists
  - reveal how different parts of brain respond to stimuli



12

Neuroscience

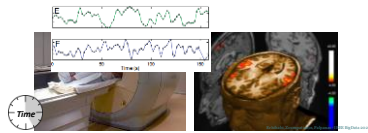
- functional Resonance Magnetic Imaging (fMRI) data
  - primary experimental tool of neuroscientists
  - reveal how different parts of brain respond to stimuli



13

Neuroscience

- functional Resonance Magnetic Imaging (fMRI) data
  - primary experimental tool of neuroscientists
  - reveal how different parts of brain respond to stimuli



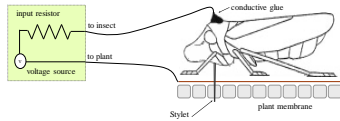
14

Entomology



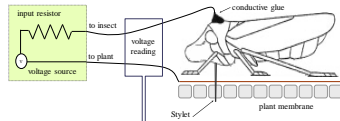
15

Entomology

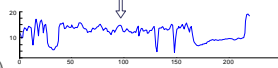


16

Entomology



17



18



19

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



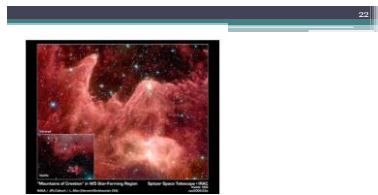
20

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



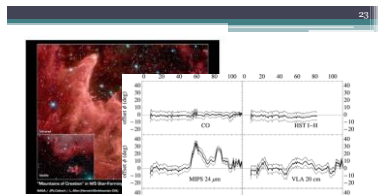
21

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



22

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



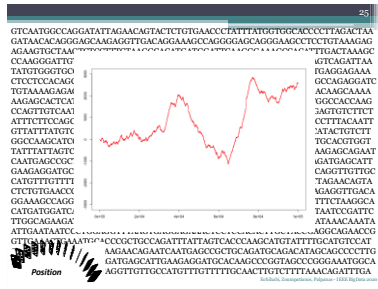
23

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



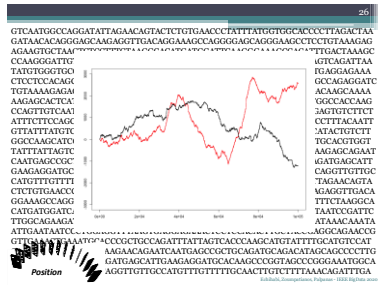
24

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



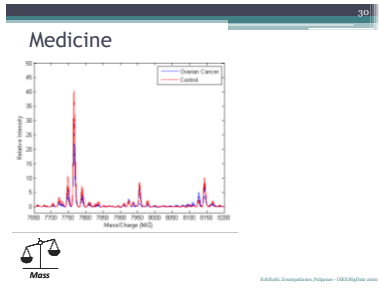
25

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



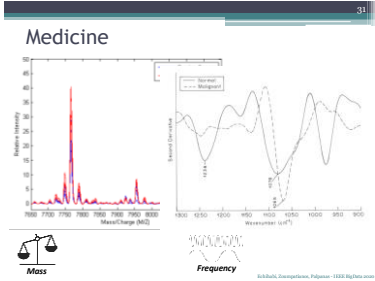
26

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



30

Elshahhat, Desimpsonen, Pajunen - 1933 (English) view



31



32



33



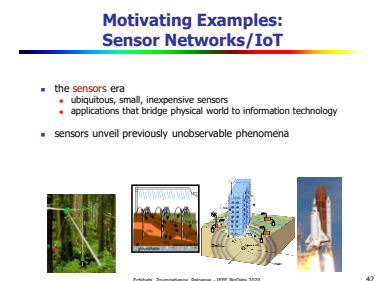
34



35



36



42

### Data as a Set

### Data as a Sequence

- streaming data
  - window of interest
    - landmark window
    - sliding window (shifting window)
- may treat streaming data as a set, or as a sequence
  - depends on whether sequence is important

Eidhubi, Zoumpalierou, Palama - IEEE BigData 2020

43

### Data Series Anomalies Problem

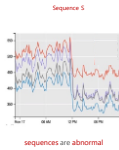
- develop anomaly detection techniques based on sequences (data series), not on individual values
  - individual values can be normal, but their sequence can be abnormal!

Eidhubi, Zoumpalierou, Palama - IEEE BigData 2020

44

### Data Series Anomalies Problem

- develop anomaly detection techniques based on sequences (data series), not on individual values

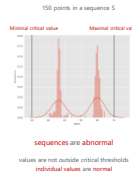


Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

45

### Data Series Anomalies Problem

- develop anomaly detection techniques based on sequences (data series), not on individual values
  - individual values can be normal, but their sequence can be abnormal!



Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

46

### Data Series (Signal) Processing Data Series Management

- lots of literature on data series processing
  - periodicity detection
  - data series modeling and forecasting
    - ARMA, ARIMA
  - outlier detection
  - focuses on next value

- instead, we will focus on
  - sequences as first class citizens
  - very large collections of data series
  - fast and scalable similarity search

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

47

### Objectives

- get introduced to the data series data type
  - characteristics, properties, peculiarities
- learn about
  - data series representations
  - data series similarity matching
  - data series indexing
  - systems for data series management
  - challenges and open problems

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

48

### Data Series Representations

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

49

### Introduction

- lots of work on data series representations

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

50

### Introduction

- lots of work on data series representations
  - techniques for representing/storing data series

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

51

### Introduction

- lots of work on data series representations
  - techniques for representing/storing data series
- main goal
  - summarize data series
  - render subsequent processing more efficient

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

52

### Outline

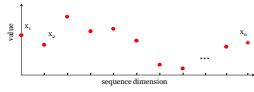
- terminology and definitions
- motivation
- pre-processing tasks
- data series representation techniques

Erlinda, Dumpsites, Pagina: 1000 Pagina: 1000

53

Data series

- Sequence of points ordered along some dimension



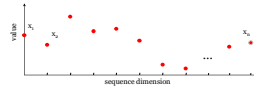
- terminology: we will use interchangeably
  - data series, time series, data sequence, sequence

© IBM Corp. 2014. All rights reserved.

54

Data series

- Sequence of points ordered along some dimension



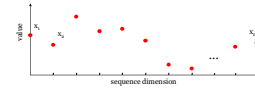
- number of data series values, n
  - length, or dimensionality

© IBM Corp. 2014. All rights reserved.

55

Data series

- Sequence of points ordered along some dimension



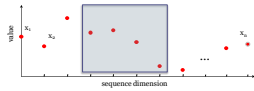
- subsequence
  - subset of contiguous values

© IBM Corp. 2014. All rights reserved.

56

Data series

- Sequence of points ordered along some dimension



- subsequence
  - subset of contiguous values
  - eg, subsequence of length (dimensionality) 4

© IBM Corp. 2014. All rights reserved.

57

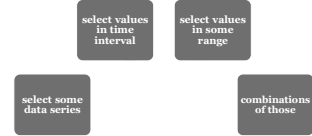
Outline

- terminology and definitions
  - motivation
- pre-processing tasks
- data series representation techniques

© IBM Corp. 2014. All rights reserved.

64

Simple Query Answering



© IBM Corp. 2014. All rights reserved.

65

Analysis Tasks



© IBM Corp. 2014. All rights reserved.

66

Analysis Tasks

- analyze evolution of values across x-dimension
- identify trends

© IBM Corp. 2014. All rights reserved.

67

Analysis Tasks

- analyze evolution of values across x-dimension
- identify trends
- treat data series as a first class citizen
  - analyze each data series as a single object
  - process all n-dimensions at once

© IBM Corp. 2014. All rights reserved.

68

Analysis Tasks  
Subsequences

- often times the data series are very long
  - $n \gg 1$
  - streaming data series

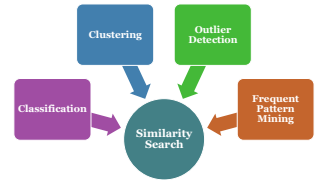
69

Analysis Tasks  
Subsequences

- often times the data series are very long
  - $n \gg 1$
  - streaming data series
- we then chop the long sequence in subsequences
  - e.g., using sliding window, or shifting window
  - pick carefully length of subsequence
    - should contain patterns of interest
- and process each subsequence separately

70

Complex Analytics



71

Complex Analytics



72

Complex Analytics



73

Motivation

- effective representation techniques to the rescue!
  - can significantly reduce the processing time
  - typically much smaller than original/raw data series

74

Motivation

- effective representation techniques to the rescue!
  - can significantly reduce the processing time
  - typically much smaller than original/raw data series
- will learn how to compute and use these representations

75

Motivation

- effective representation techniques to the rescue!
  - can significantly reduce the processing time
  - typically much smaller than original/raw data series
- will learn how to compute and use these representations
- these representations can further be used for indexing

76

Motivation

- effective representation techniques to the rescue!
  - can significantly reduce the processing time
  - typically much smaller than original/raw data series
- will learn how to compute and use these representations
- these representations can further be used for indexing
- all **guarantee correct, exact results!**

77

Outline

- terminology and definitions
- motivation
- **pre-processing tasks**
- data series representation techniques

78

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization

- data series encode trends
- usually interested in identifying similar trends

79

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

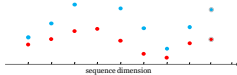
Pre-Processing  
z-Normalization

- data series encode trends
- usually interested in identifying similar trends
- but **absolute** values may mask this similarity

80

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization

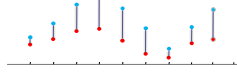


- two data series with similar trends

81

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization

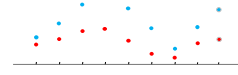


- two data series with similar trends
- but large distance...

82

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization

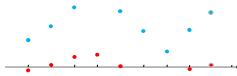


- **zero mean**
  - compute the mean of the sequence
  - subtract the mean from every value of the sequence

83

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization

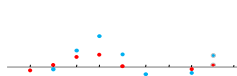


- **zero mean**
  - compute the mean of the sequence
  - subtract the mean from every value of the sequence

84

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization

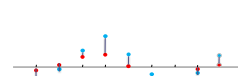


- **zero mean**
  - compute the mean of the sequence
  - subtract the mean from every value of the sequence

85

© Holthaus, Demegortian, Palamara - 1998 Rights reserved

Pre-Processing  
z-Normalization



- **zero mean**
  - compute the mean of the sequence
  - subtract the mean from every value of the sequence

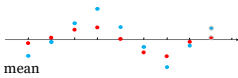
86

© Holthaus, Demegortian, Palamara - 1998 Rights reserved



87

### Pre-Processing z-Normalization

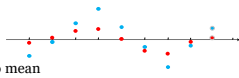


- zero mean
  - compute the standard deviation of the sequence
  - divide every value of the sequence by the stddev
- standard deviation one

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

88

### Pre-Processing z-Normalization

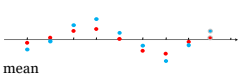


- zero mean
  - compute the standard deviation of the sequence
  - divide every value of the sequence by the stddev
- standard deviation one

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

89

### Pre-Processing z-Normalization

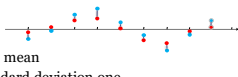


- zero mean
  - compute the standard deviation of the sequence
  - divide every value of the sequence by the stddev
- standard deviation one

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

90

### Pre-Processing z-Normalization



- zero mean
- standard deviation one

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

91

### Pre-Processing z-Normalization

- when to z-normalize
  - interested in trends

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

92

### Pre-Processing z-Normalization

- when to z-normalize
  - interested in trends
- when not to z-normalize
  - interested in absolute values

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

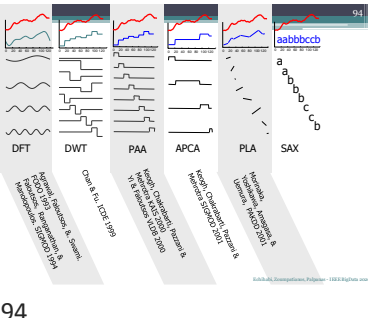
93

### Outline

- terminology and definitions
- motivation
- pre-processing tasks
- data series representation techniques

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

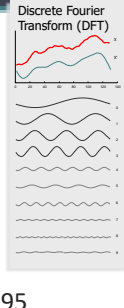
94



© 2004, Simon Steiner, Palgrave - 1000 Rights.com

95

### Discrete Fourier Transform (DFT)



**Basic Idea:** Represent the time series as a linear combination of sines and cosines

Transform the data from the time domain to the frequency domain

Highlight the periodicities but keep only the first  $n/2$  coefficients

Why  $n/2$  coefficients?  
 ✓ Because they are symmetric

Excellent free Fourier Primer  
 Hagg Shalaby, The Fourier Transform - a Primer, Technical Report CS-95-37, Department of Computer Science, Brown University, 1995.  
<http://www.ncsl.nist.gov/CB/Research/Postdoc/Shalaby/>

© 2004, Simon Steiner, Palgrave - 1000 Rights.com

### Discrete Fourier Transform...recap

Pros and Cons of DFT as a time series representation

**Pros:**

- Good ability to compress most natural signals
- Fast, off the shelf DFT algorithms exist  $O(\log(n))$

**Cons:**

- Difficult to deal with sequences of different lengths

96

### Discrete Wavelet Transform (DWT)

**Basic Idea:** Represent the time series as a linear combination of wavelet basis functions, but keep only the first  $M$  coefficients

Obtained from a single prototype wavelet  $\psi(t)$  called *mother wavelet* by *dilatations* and *shifting*:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right)$$

where  $a$  is the scaling parameter and  $b$  is the shifting parameter

Excellent free Wavelets Primer  
Stolnitz, E., DeRose, T., & Salesin, D. (1995). *Wavelets for computer graphics A primer: IEEE Computer Graphics and Applications.*

97

### Discrete Wavelet Transform (DWT)

Pros and Cons of DWT as a time series representation

**Pros:**

- Good ability to compress stationary signals
- Can be computed in linear time

**Cons:**

- Signals must have a length  $n = 2^{\text{some\_integer}}$
- Works best if  $N$  is  $= 2^{\text{some\_integer}}$ ; Otherwise wavelets approximate the left side of signal at the expense of the right side

98

### Piecewise Aggregate Approximation (PAA)

**Basic Idea:** Represent the time series as a sequence of box basis functions, each box being of the same length

**Computation:**

- $X$ : time series of length  $n$
- Can be represented in the  $N$ -dimensional space as:

$$\bar{X}_i = \frac{1}{n} \sum_{j=\phi(i-1)+1}^{\phi(i)} x_j$$

Keogh, Chakrabarti, Pazzani & Mehrotra, KDD (2000)  
Byoung-Kee Yi, Christos Faloutsos, VLDB (2000)

99

### Piecewise Aggregate Approximation (PAA)

Pros and Cons of PAA as a time series representation.

**Pros:**

- Extremely fast to calculate
- As efficient as other approaches (empirically)
- Support queries of arbitrary lengths
- Can support any Minkowski metric
- Supports non Euclidean measures
- Supports weighted Euclidean distance
- Simple/ Intuitive!

**Cons:**

- If visualized directly, looks aesthetically unpleasing

100

### Piecewise Linear Approximation (PLA)

**Basic Idea:** Represent the time series (size  $n$ ) as a sequence of straight lines (size  $N$ )

Lines could be **connected**  $\Rightarrow > N/2$  lines allowed

Lines could be **disconnected**  $\Rightarrow > N/3$  lines allowed

Empirical evidence on dozens of datasets suggests that **disconnected** is better

Also only **disconnected** allows a lower bounding Euclidean approximation

Karl Friedrich Gauss  
1777 - 1855

Each line segment has

- length
- left\_weight
- right\_weight (right\_weight can be inferred by looking at the next segment)

Each line segment has

- length
- left\_weight
- right\_weight

101

### Piecewise Linear Approximation (PLA)

Pros and Cons of PLA as a time series representation

**Pros:**

- Good ability to compress natural signals
- Fast linear time algorithms for PLA exist
- Able to support some interesting non-Euclidean similarity measures
- Already widely accepted in some communities (i.e., biomedical)

**Cons:**

- Not (currently) "indexable" by any data structure (but does allow fast sequential scan)

102

### Adaptive Piecewise Constant Approximation (APCA)

**Basic Idea:** Represent the time series as a sequence of box basis functions, each box being of the *different* length

- High quality of APCA noted by many researchers
- Can be indexed\*!

Unfortunately, it is non-trivial to understand and implement and thus has only been re-implemented once or twice

\*K. Chakrabarti, E. J. Keogh, S. Mehrotra, M. J. Pazzani: Locally adaptive dimensionality reduction for indexing large time series databases. ACM Trans. Database Syst. 27(2): 188-228 (2002)

103

### Adaptive Piecewise Constant Approximation (APCA)

Pros and Cons of APCA as a time series representation

**Pros:**

- Fast to calculate  $O(n)$
- More efficient than other approaches
- Supports queries of arbitrary lengths
- Supports non Euclidean measures
- Support fast exact queries, and even faster approximate queries on the same data structure

**Cons:**

- Somewhat complex implementation
- If visualized directly, looks aesthetically unpleasing

104

109

### SAX Representation

- Symbolic Aggregate approximation (SAX)
  - (1) Represent data series  $T$  of length  $n$  with  $m$  segments using Piecewise Aggregate Approximation (PAA)
    - $T$  typically normalized to  $\mu = 0, \sigma = 1$
    - $PAA(T, m) = \vec{T} = \vec{t}_1, \dots, \vec{t}_m$
    - where  $\vec{t}_i = \frac{1}{m} \sum_{j=2^{i-1}+1}^{2^i} T_j$
  - (2) Discretize into a vector of symbols
    - Breakpoints map to small alphabet  $\alpha$  of symbols

109

136

### Similarity Search

136

137

### Distance Measures

- similarity search is based on measuring distance between sequences
- dozens of distance measures have been proposed
  - lock-step
    - Minkowski, Manhattan, Euclidean, Maximum, DISSIM, ...
  - sliding
    - Normalized Cross-Correlation, SBD, ...
  - elastic
    - DTW, LCSS, MSM, EDR, ERP, Swale, ...
  - kernel-based
    - KDTW, GAK, SINK, ...
  - embedding
    - GRAIL, RWS, SPIRAL, ...

137

138

### Distance Measures

- similarity search is based on measuring distance between sequences
- dozens of distance measures have been proposed
  - lock-step
    - Minkowski, Manhattan, Euclidean, Maximum, DISSIM, ...
  - sliding
    - Normalized Cross-Correlation, SBD, ...
  - elastic
    - DTW, LCSS, MSM, EDR, ERP, Swale, ...
  - kernel-based
    - KDTW, GAK, SINK, ...
  - embedding
    - GRAIL, RWS, SPIRAL, ...

138

139

### Euclidean Distance

139

140

### Euclidean Distance

140

141

### Euclidean Distance

- Euclidean distance
  - pair-wise point distance
$$ED(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

141

142

### Correlation

- measures the degree of relationship between data series
  - indicates the degree and direction of relationship

142

143

### Correlation

- measures the degree of relationship between data series
  - indicates the degree and direction of relationship
- direction of change
  - positive correlation
    - values of two data series change in same direction
  - negative correlation
    - values of two data series change in opposite directions

143

Correlation

- measures the degree of relationship between data series
  - indicates the degree and direction of relationship
- direction of change
  - positive correlation
    - values of two data series change in same direction
  - negative correlation
    - values of two data series change in opposite directions
- linear correlation
  - amount of change in one data series bears constant ratio of change in the other data series

Edinburgh, Southampton, Palgrave - 1988 Right/Repro

144

Correlation

- measures the degree of relationship between data series
  - indicates the degree and direction of relationship
- direction of change
  - positive correlation
    - values of two data series change in same direction
  - negative correlation
    - values of two data series change in opposite directions
- linear correlation
  - amount of change in one data series bears constant ratio of change in the other data series
- useful in several applications

Edinburgh, Southampton, Palgrave - 1988 Right/Repro

145

Pearson's Correlation Coefficient

- used to see linear dependency between values of data series of equal length, n

$$PC = \frac{1}{n-1} \sum_{i=1}^n \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

Edinburgh, Southampton, Palgrave - 1988 Right/Repro

146

Pearson's Correlation Coefficient

- used to see linear dependency between values of data series of equal length, n

$$PC = \frac{1}{n-1} \sum_{i=1}^n \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

- where  $\bar{x}$  is the mean:  $\bar{x} = \frac{1}{n-1} \sum_{i=1}^n x_i$
- and  $s_x$  is the standard deviation:  $s_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2}$

Edinburgh, Southampton, Palgrave - 1988 Right/Repro

147

Pearson's Correlation Coefficient

- used to see linear dependency between values of data series of equal length, n

$$PC = \frac{1}{n-1} \sum_{i=1}^n \left( \frac{x_i - \bar{x}}{s_x} \right) \left( \frac{y_i - \bar{y}}{s_y} \right)$$

- takes values in [-1,1]
  - 0 - no correlation
  - 1, 1 - inverse/direct correlation
- there is a statistical test connected to PC, where null hypothesis is the no correlation case (correlation coefficient = 0)
  - test is used to ensure that the correlation similarity is not caused by a random process

Edinburgh, Southampton, Palgrave - 1988 Right/Repro

148

PC and ED

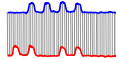
- Euclidean distance:  $ED = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$
- In case of Z-normalized data series (mean = 0, stddev = 1):
  - $PC = \frac{1}{n-1} \sum_{i=1}^n x_i \cdot y_i$  and  $ED^2 = 2n(n-1) - 2 \sum_{i=1}^n x_i y_i$
- so the following formula is true:  $ED^2 = 2(n-1)(n-PC)$
- direct connection between ED and PC for Z-normalized data series
  - if ED is calculated for normalized data series, it can be directly used to calculate the p-value for statistical test of Pearson's correlation instead of actual PC value.

Edinburgh, Southampton, Palgrave - 1988 Right/Repro

149

Distance Measures: LCSS against Euclidean, DTW

- Euclidean
  - rigid

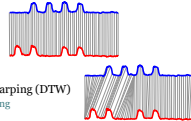


Edinburgh, Southampton, Palgrave - 1988 Right/Repro

150

Distance Measures: LCSS against Euclidean, DTW

- Euclidean
  - rigid
- Dynamic Time Warping (DTW)
  - allows local scaling

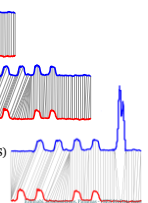


Edinburgh, Southampton, Palgrave - 1988 Right/Repro

151

Distance Measures: LCSS against Euclidean, DTW

- Euclidean
  - rigid
- Dynamic Time Warping (DTW)
  - allows local scaling
- Longest Common SubSequence (LCSS)
  - allows local scaling
  - ignores outliers



152

153

## Similarity Matching

- given a data series collection  $D$  and a query data series  $q$ , return the data series from  $D$  that are the most similar to  $q$ 
  - there exist different flavors of this basic operation
- basis for most data series analysis tasks

153

154

## Similarity Matching Nearest Neighbor (NN) Search

- given a data series collection  $D$  and a query data series  $q$ , return the data series from  $D$  that has the smallest distance to  $q$

154

155

## Similarity Matching Nearest Neighbor (NN) Search

- given a data series collection  $D$  and a query data series  $q$ , return the data series from  $D$  that has the smallest distance to  $q$
- result set contains one data series

155

156

## Similarity Matching Nearest Neighbor (NN) Search

- **serial scan**
  - compute the distance between  $q$  and every  $d_i \in D$
  - return  $d_i$  with the smallest distance to  $q$

156

157

## Similarity Matching Nearest Neighbor (NN) Search

- **serial scan**
  - $bsf = \text{Inf}$  // best so far distance
  - for every  $d_i \in D$ 
    - compute distance,  $dist$ , between  $d_i$  and  $q$
    - if this  $dist$  less than  $bsf$  then  $bsf = dist$
  - return  $d_i$  corresponding to  $bsf$

157

158

## Similarity Matching k-Nearest Neighbors (kNN) Search

- given a data series collection  $D$  and a query data series  $q$ , return the  $k$  data series from  $D$  that have the  $k$  smallest distances to  $q$

158

159

## Similarity Matching k-Nearest Neighbors (kNN) Search

- given a data series collection  $D$  and a query data series  $q$ , return the  $k$  data series from  $D$  that have the  $k$  smallest distances to  $q$
- result set contains  $k$  data series

159

160

## Similarity Matching k-Nearest Neighbors (kNN) Search

- **serial scan**
  - compute the distance between  $q$  and every  $d_i \in D$
  - return the  $k$   $d_i$  with the  $k$  smallest distances to  $q$

160

161

## Similarity Matching k-Nearest Neighbors (kNN) Search

- **serial scan**
  - $kbsf = \text{Null}$  // best so far max-heap of  $k$  elements
  - for every  $d_i \in D$ 
    - compute distance,  $dist$ , between  $d_i$  and  $q$
    - if this  $dist$  less than  $\max$  of  $kbsf$  then insert  $dist$  in  $kbsf$
  - return  $k$   $d_i$  corresponding to  $k$  elements in  $kbsf$

161

### Similarity Matching $\epsilon$ -Range Search

- given a data series collection D and a query data series q, return all data series from D that are within distance  $\epsilon$  from q

162

### Similarity Matching $\epsilon$ -Range Search

- given a data series collection D and a query data series q, return all data series from D that are within distance  $\epsilon$  from q
- result set contains [?] data series

163

### Similarity Matching $\epsilon$ -Range Search

- serial scan
  - compute the distance between q and every  $d_i \in D$
  - return all  $d_i$  with distance less than  $\epsilon$  to q

164

### Similarity Matching $\epsilon$ -Range Search

- serial scan
  - res = {} // empty result set
  - for every  $d_i \in D$ 
    - compute distance, dist, between  $d_i$  and q
    - if this dist less than  $\epsilon$  then insert dist in res
  - return all  $d_i$  corresponding to elements in res

165

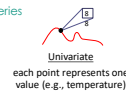
### Problem Variations

Series

166

### Problem Variations

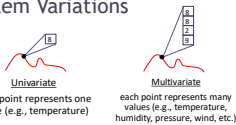
Series



167

### Problem Variations

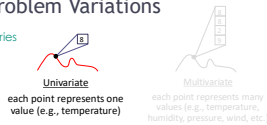
Series



168

### Problem Variations

Series



169

### Problem Variations

Queries



170

ciN 171

### Problem Variations

Queries

**Whole matching**

Entire query  
Entire candidate

**Subsequence matching**

Entire query  
A subsequence of a candidate

© 2013 Morgan Kaufmann Publishers. All rights reserved.

171

ciN 172

### Problem Variations

Queries

**Whole matching**

Entire query  
Entire candidate

**Subsequence matching**

Entire query  
A subsequence of a candidate

© 2013 Morgan Kaufmann Publishers. All rights reserved.

172

ciN 173

### Problem Variations

Distances

- Euclidean Distance (ED)
- Dynamic Time Warping (DTW)
- Longest Common Subsequence (LCSS)
- Edit Distance
- And more...

© 2013 Morgan Kaufmann Publishers. All rights reserved.

173

ciN 174

### Methods

Similarity Search Methods

**Approximate** (PVLDB'20)

No guarantee

**ε-Approximate**

© 2013 Morgan Kaufmann Publishers. All rights reserved.

174

ciN 175

### Methods

Similarity Search Methods

**Approximate** (PVLDB'20)

No guarantee

**ε-Approximate**

© 2013 Morgan Kaufmann Publishers. All rights reserved.

175

ciN 176

### Methods

Similarity Search Methods

**Approximate** (PVLDB'20)

$0 < \delta < \epsilon, \epsilon > 0$

$\delta, \epsilon$  guarantee

No guarantee

**δ-ε-Approximate\***

**ε-Approximate**

© 2013 Morgan Kaufmann Publishers. All rights reserved.

\* result is within distance  $(1+\epsilon)$  of the exact answer with probability  $\delta$

176

ciN 177

### Methods

Similarity Search Methods

**Approximate** (PVLDB'20)

$0 < \delta < \epsilon, \epsilon > 0$

$\delta, \epsilon$  guarantee

No guarantee

**δ-ε-Approximate\***

$\delta < \epsilon, \epsilon$  guarantee

**Probabilistic**

**ε-Approximate**

© 2013 Morgan Kaufmann Publishers. All rights reserved.

\* result is within distance  $(1+\epsilon)$  of the exact answer with probability  $\delta$

177

ciN 178

### Methods

Similarity Search Methods

**Approximate** (PVLDB'20)

$0 < \delta < \epsilon, \epsilon > 0$

$\delta, \epsilon$  guarantee

No guarantee

**δ-ε-Approximate\***

$\delta < \epsilon, \epsilon$  guarantee |  $\delta < \epsilon, \epsilon$  guarantee

**Probabilistic**

**ε-Approximate**

© 2013 Morgan Kaufmann Publishers. All rights reserved.

\* result is within distance  $(1+\epsilon)$  of the exact answer with probability  $\delta$

178

ciN 179

### Methods

Similarity Search Methods

**Approximate** (PVLDB'20)

$0 < \delta < \epsilon, \epsilon > 0$

$\delta, \epsilon$  guarantee

No guarantee

**δ-ε-Approximate\***

$\delta < \epsilon, \epsilon$  guarantee |  $\delta < \epsilon, \epsilon$  guarantee

**Probabilistic**

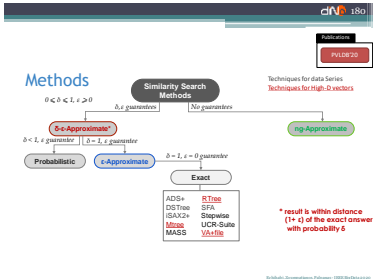
**ε-Approximate**

**Exact**

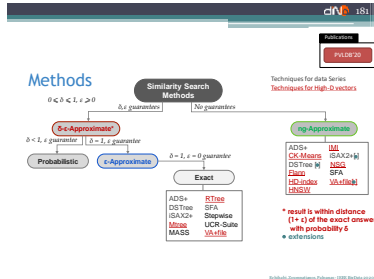
© 2013 Morgan Kaufmann Publishers. All rights reserved.

\* result is within distance  $(1+\epsilon)$  of the exact answer with probability  $\delta$

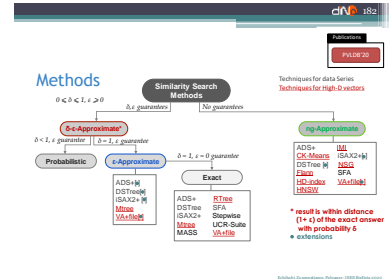
179



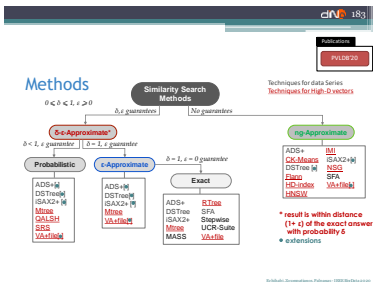
180



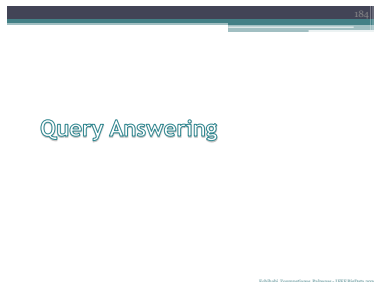
181



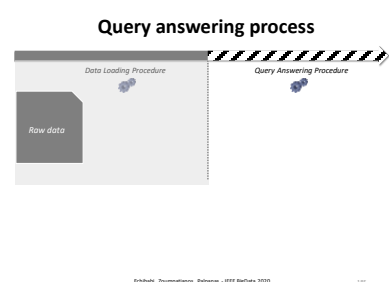
182



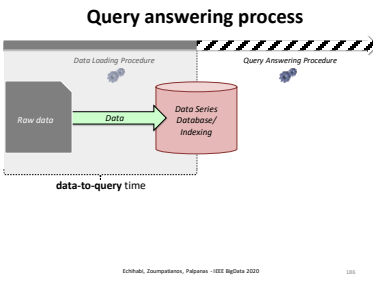
183



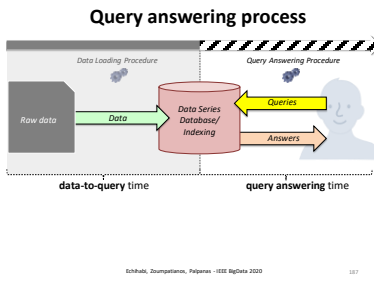
184



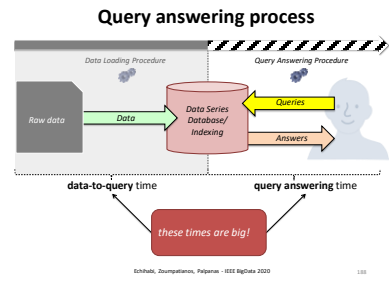
185



186



187



188



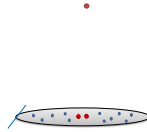
Similarity Search via Serial Scan



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

189

Similarity Search via Serial Scan



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

190

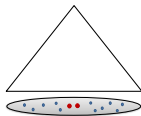
Similarity Search via Serial Scan



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

191

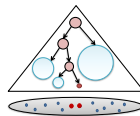
Similarity Search via Indexing



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

192

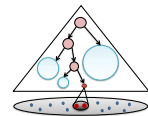
Similarity Search via Indexing



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

193

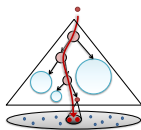
Similarity Search via Indexing



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

194

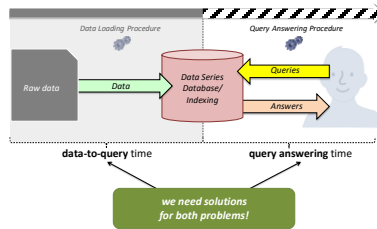
Similarity Search via Indexing



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

195

Query answering process



Eshhaki, Zoumpatianos, Palpanas - IEEE BigData 2020

196

Similarity Matching Fast Euclidean Distance

- similarity matching requires many distance computations
- can significantly slow down processing
- because of large number of data series in the collection
- because of high dimensionality of each data series

197

198

## Similarity Matching Fast Euclidean Distance

- similarity matching requires many distance computations
  - can significantly slow down processing
    - because of large number of data series in the collection
    - because of high dimensionality of each data series
- in case of Euclidean Distance, we can speedup processing by
  - smart implementation of distance function
  - early abandoning

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

198

199

## Similarity Matching Fast Euclidean Distance

- similarity matching requires many distance computations
  - can significantly slow down processing
    - because of large number of data series in the collection
    - because of high dimensionality of each data series
- in case of Euclidean Distance, we can speedup processing by
  - smart implementation of distance function
  - early abandoning
- result in **considerable** performance improvement

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

199

200

## Similarity Matching Fast Euclidean Distance

- smart implementation of distance function

$$ED(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

200

201

## Similarity Matching Fast Euclidean Distance

- smart implementation of distance function
  - do **not** compute the square root (of the Euclidean Distance)

$$ED(X, Y) = \sum_{i=1}^n (x_i - y_i)^2$$

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

201

202

## Similarity Matching Fast Euclidean Distance

- smart implementation of distance function
  - do **not** compute the square root (of the Euclidean Distance)

$$ED(X, Y) = \sum_{i=1}^n (x_i - y_i)^2$$

- does not alter the results
- saves precious CPU cycles

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

202

203

## Similarity Matching Fast Euclidean Distance

- early abandoning
  - **stop** the distance computation as soon as it exceeds the value of bsf

$$ED(X, Y) = \sum_{i=1}^m (x_i - y_i)^2, \quad m \leq n$$

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

203

204

## Similarity Matching Fast Euclidean Distance

- early abandoning
  - **stop** the distance computation as soon as it exceeds the value of bsf

$$ED(X, Y) = \sum_{i=1}^m (x_i - y_i)^2, \quad m \leq n$$

- does not alter the results
- avoids useless computations

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

204

## GEMINI Framework

- Raw data: original full-dimensional space
- Summarization: reduced dimensionality space
- Searching in original space *costly*
- Searching in reduced space *faster*:
  - Less data, indexing techniques available, lower bounding

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

205

205

## GEMINI Framework

- Raw data: original full-dimensional space
- Summarization: reduced dimensionality space
- Searching in original space *costly*
- Searching in reduced space *faster*:
  - Less data, indexing techniques available, lower bounding
- Lower bounding enables us to
  - *prune search space*: throw away data series based on reduced dimensionality representation
  - *guarantee correctness* of answer
    - no false negatives
    - false positives filtered out based on raw data

©Mehdi B. Shamsi, University of Pennsylvania, 1998, Rights Reserved.

206

206

### GEMINI Framework

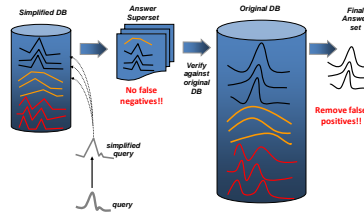
GEMINI Solution: Quick filter-and-refine:

- extract  $m$  features (numbers, e.g., average)
- map to point in  $m$ -dimensional feature space
- organize points
- retrieve the answer using a NN query
- discard false positives

© 2016, DeepLearning.AI. All rights reserved.

208

### Generic Search using Lower Bounding



© 2016, DeepLearning.AI. All rights reserved.

209

### GEMINI: contractiveness

- GEMINI works when:

$$D_{\text{feature}}(F(x), F(y)) \leq D(x, y)$$

- Note that, the closer the feature distance to the actual one, the better

© 2016, DeepLearning.AI. All rights reserved.

210

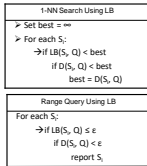
### Lower Bounding

We can speed up similarity search by using a lower bounding function

- $D$ : distance measure
- LB: lower bounding function s.t.:  $LB(Q, S_i) \leq D(Q, S_i)$

**Intuition**

- ✓ Try to use a cheap lower bounding calculation as often as possible
- ✓ Do the expensive, full calculations when absolutely necessary



© 2016, DeepLearning.AI. All rights reserved.

211

### Lower Bounding

we want to find the 1-NN to our query data series,  $Q$

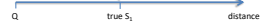


© 2016, DeepLearning.AI. All rights reserved.

212

### Lower Bounding

we compute the distance to the first data series in our dataset,  $D(S_1, Q)$  this becomes the best so far (BSF)



© 2016, DeepLearning.AI. All rights reserved.

213

### Lower Bounding

we compute the distance  $LB(S_2, Q)$  and it is greater than the BSF we can safely prune it, since  $D(S_2, Q) \geq LB(S_2, Q)$



© 2016, DeepLearning.AI. All rights reserved.

214

### Lower Bounding

we compute the distance  $LB(S_3, Q)$  and it is smaller than the BSF we have to compute  $D(S_3, Q) \geq LB(S_3, Q)$ , since it may still be smaller than BSF



© 2016, DeepLearning.AI. All rights reserved.

215

### Lower Bounding

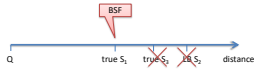
it turns out that  $D(S_3, Q) \geq \text{BSF}$ , so we can safely prune  $S_3$



© 2016, DeepLearning.AI. All rights reserved.

216

Lower Bounding

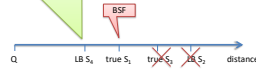


© 2010, University of Pennsylvania. All rights reserved.

217

Lower Bounding

we compute  $LB(S_4, Q)$  and it is smaller than the BSF  
we have to compute  $DS(S_4, Q) \geq LB(S_4, Q)$ , since it may still be smaller than BSF

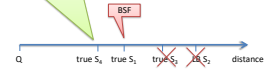


© 2010, University of Pennsylvania. All rights reserved.

218

Lower Bounding

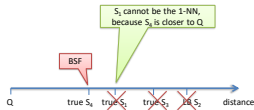
It turns out that  $DS(S_4, Q) < BSF$ , so  $S_4$  becomes the new BSF



© 2010, University of Pennsylvania. All rights reserved.

219

Lower Bounding

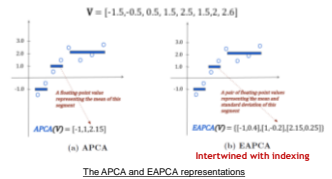


© 2010, University of Pennsylvania. All rights reserved.

220

Data Series Indexing

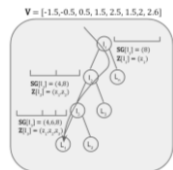
DSTree Summarization



© 2010, University of Pennsylvania. All rights reserved.

222

DSTree Indexing



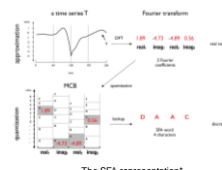
- Each node contains
  - # vectors
  - segmentation SG
  - synopsis Z

- Each Leaf node also :
  - stores its raw vectors in a separate disk file

© 2010, University of Pennsylvania. All rights reserved.

223

Symbolic Fourier Approximation (SFA) Summarization



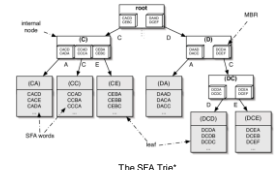
The SFA representation

https://www2.informatik.hu-berlin.de/~schaefer/papers/sfa\_classification.pptx

© 2010, University of Pennsylvania. All rights reserved.

224

SFA Indexing



The SFA Trie

https://www2.informatik.hu-berlin.de/~schaefer/papers/sfa\_classification.pptx

© 2010, University of Pennsylvania. All rights reserved.

225

ciN 226

### iSAX

- based on iSAX representation, which offers a bit-aware, quantized, multi-resolution representation with variable granularity

$[6, 6, 3, 0] = [110, 110, 0111, 000]$   
 $[3, 3, 1, 0] = [11, 11, 011, 00]$   
 $[1, 1, 0, 0] = [1, 1, 0, 0]$

227

226

ciN 227

### iSAX

228

227

ciN 228

229

228

ciN 229

### iSAX2+

- implements bulk loading strategy for iSAX:
- does not move around (read/write) raw data of data series and its approximation **unless necessary**
- intuition for proposed solution:
  - iSAX grows fast at the beginning of bulk loading, its shape stabilizing well before the end of the process
  - several data series end up in leaf nodes that never need to split
  - implement lazy splitting:
    - move raw data to leaf node the first time
    - if leaf node splits, do not move raw data until the end of index building process

230

229

ciN 230

### ADS+

- novel paradigm for building a data series index
  - does not build entire index and then answer queries
  - starts answering queries by building the part of the index needed by those queries
- still guarantees correct answers
- intuition for proposed solution
  - builds index using only iSAX summaries; uses large leaf size
  - postpones leaf materialization to query time
    - only materialize (at query time) leaves needed by queries
  - parts that are queried more are refined more
    - use smaller leaf sizes (reduced leaf materialization and query answering costs)

231

230

ciN 297

### Extensions...

- Coconut: current solution for limited memory devices and streaming time series
  - bottom-up, succinct index construction based on sortable summarizations

232

297

ciN 298

### Extensions...

- Coconut: current solution for limited memory devices and streaming time series
  - bottom-up, succinct index construction based on sortable summarizations
  - outperforms state-of-the-art in terms of index space, index construction time, and query answering time

233

298

ciN 299

### Extensions...

- Coconut: current solution for limited memory devices and streaming time series
  - bottom-up, succinct index construction based on sortable summarizations
  - outperforms state-of-the-art in terms of index space, index construction time, and query answering time

234

299

ciN 300

### Coconut-LSM

### Extensions...

- Coconut: current solution for limited memory devices and streaming time series
  - bottom-up, succinct index construction based on sortable summarizations
  - outperforms state-of-the-art in terms of index space, index construction time, and query answering time

235

300



Parallelization/Distribution

- DPiSAX:** current solution for distributed processing (Sparrow)
  - balances work of different worker nodes
  - performs 2 orders of magnitude faster than centralized solution
- ParIS:** current solution for modern hardware
  - masks out the CPU cost
  - answers exact queries in the order of a few secs
    - 3 orders of magnitude faster than single-core solutions

310

Parallelization/Distribution

**k-NN Classification**

Time (Seconds)

Number of nearest neighbors

- DPiSAX:** current solution for distributed processing (Sparrow)
  - balances work of different worker nodes
  - performs 2 orders of magnitude faster than centralized solution
- ParIS:** current solution for modern hardware
  - masks out the CPU cost
  - answers exact queries in the order of a few secs
    - 3 orders of magnitude faster than single-core solutions

311

Parallelization/Distribution

**k-NN Classification**

Time (Seconds)

Number of nearest neighbors

- DPiSAX:** current solution for distributed processing (Sparrow)
  - balances work of different worker nodes
  - performs 2 orders of magnitude faster than centralized solution
- ParIS:** current solution for modern hardware
  - masks out the CPU cost
  - answers exact queries in the order of a few secs
    - 3 orders of magnitude faster than single-core solutions

312

Parallelization/Distribution

**k-NN Classification**

classifying 100K objects using a 100GB dataset goes down from several days to few hours!

Time (Seconds)

Number of nearest neighbors

- DPiSAX:** current solution for distributed processing (Sparrow)
  - balances work of different worker nodes
  - performs 2 orders of magnitude faster than centralized solution
- ParIS:** current solution for modern hardware
  - masks out the CPU cost
  - answers exact queries in the order of a few secs
    - > 1 order of magnitude faster than single-core solutions
- MESSI:** current single-node parallel solution + in-memory data
  - answers exact queries at interactive speeds: ~30msec on 100GB
- SING:** current single-node parallel solution + GPU + in-memory data
  - answers exact queries at interactive speeds: ~32msec on 100GB

313

Parallelization/Distribution

- DPiSAX:** current solution for distributed processing (Sparrow)
  - balances work of different worker nodes
  - performs 2 orders of magnitude faster than centralized solution
- ParIS:** current single-node parallel solution
  - masks out the CPU cost
  - answers exact queries in the order of a few secs
    - > 1 order of magnitude faster than single-core solutions
- MESSI:** current single-node parallel solution + in-memory data
  - answers exact queries at interactive speeds: ~30msec on 100GB
- SING:** current single-node parallel solution + GPU + in-memory data
  - answers exact queries at interactive speeds: ~32msec on 100GB

314

iSAX Index Family

Timeline: 2006, 2010, 2014, 2015, 2017, 2018, 2019, 2020

- Basic Index:** iSAX
- Basic Index Loading:** iSAX 2.0, iSAX+, iSAX2+
- Adaptive:** iSAX+, iSAX2+
- Distributed:** DPiSAX
- Multi-Core, Multi-Socket, SIMD:** ParIS, ParIS+, MESSI
- Graphics Processing Units (GPU):** SING
- Sortable Summarizations, Streaming Data Series:** iSAX-Tree / iSAX-Stream, iSAX-Stream
- Variable-Length Queries:** iSAX-Stream

315

Experimental Comparison: Exact Query Answering Methods

316

How do these methods compare?

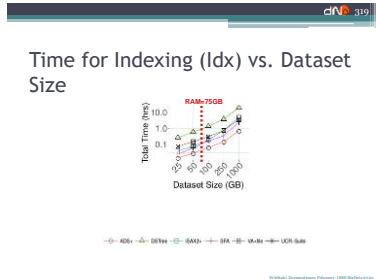
- several methods proposed in last 3 decades
- never carefully compared to one another
- we now present results of extensive experimental comparison

317

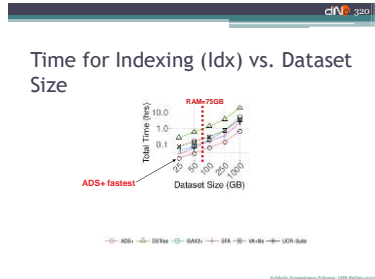
Experimental Framework

- Hardware**
  - HDD and SSD
- Datasets**
  - Synthetic (25GB to 1TB) and 4 real (100 GB)
- Exact Query Workloads**
  - 100 - 10,000 queries
- Performance measures**
  - Time, # disk accesses, footprint, pruning, Tightness of Lower Bound (TLB), etc.
- C/C++ methods (4 methods reimplemented from scratch)**
  - Procedure:**
    - Step 1: Parametrization
    - Step 2: Evaluation of individual methods
    - Step 3: Comparison of best methods

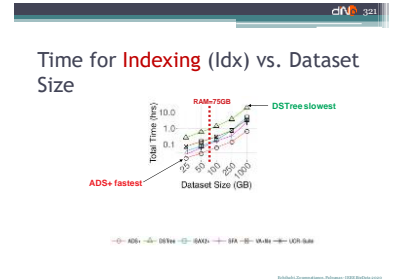
318



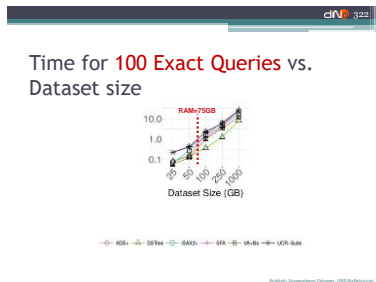
319



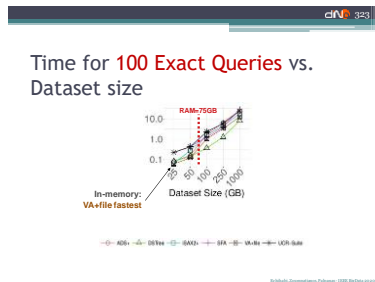
320



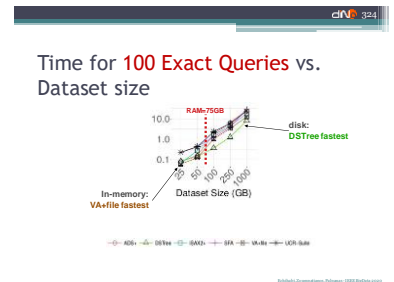
321



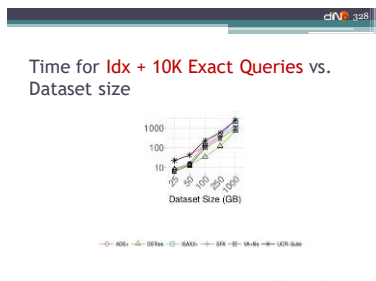
322



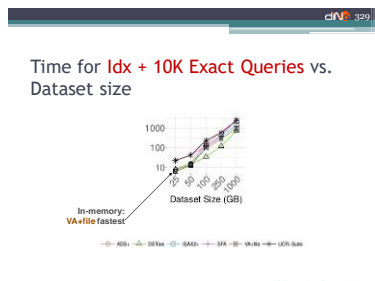
323



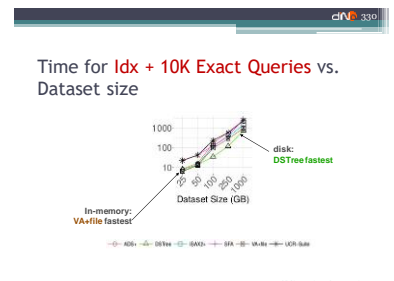
324



328

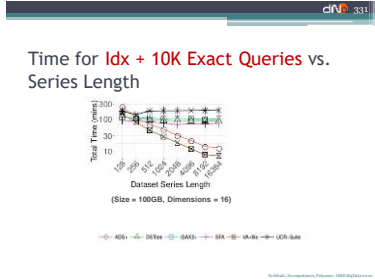


329

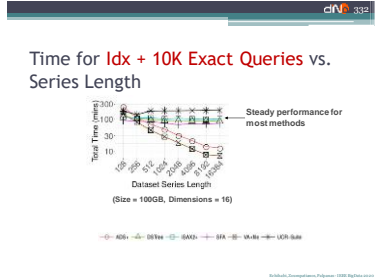


330

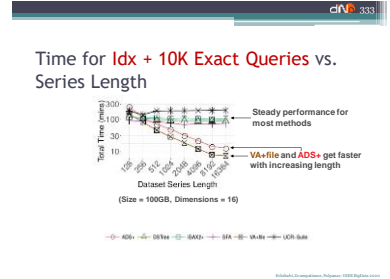




331



332



333

### Unexpected Results

- Some methods do not scale as expected (or not at all!)
- Brought back to the spotlight two older methods VA+file and DSTree
  - Our reimplementations outperform by far the original ones
- Optimal parameters for some methods are different from the ones reported in the original papers
- Tightness of Lower Bound (TLB) does not always predict performance

334

### TLB does not always predict performance

335

### TLB does not always predict performance

The TLB measures the quality of a summarization (higher is better)

336

### TLB does not always predict performance

The TLB measures the quality of a summarization (higher is better)

$$TLB = \frac{dist(Query.candidate) \text{ in reduced space}}{dist(Query.candidate) \text{ in original space}} \leq 1$$

337

### TLB does not always predict performance

The TLB measures the quality of a summarization (higher is better)

$$0 \leq TLB = \frac{dist(Query.candidate) \text{ in reduced space}}{dist(Query.candidate) \text{ in original space}} \leq 1$$

worst  best

338

### TLB does not always predict performance

The TLB measures the quality of a summarization (higher is better)

$$0 \leq TLB = \frac{dist(Query.candidate) \text{ in reduced space}}{dist(Query.candidate) \text{ in original space}} \leq 1$$

worst  best

339

TLB does not always predict performance

The TLB measures the quality of a summarization (higher is better)  $0 \leq TLB = \frac{dist(Query.candidate) \text{ in reduced space}}{dist(Query.candidate) \text{ in original space}} \leq 1$

worst best

DS Tree and ISAX2+ have similar TLB

ISAX2+ is slower than DS Tree

340

TLB does not always predict performance

The TLB measures the quality of a summarization (higher is better)  $0 \leq TLB = \frac{dist(Query.candidate) \text{ in reduced space}}{dist(Query.candidate) \text{ in original space}} \leq 1$

worst best

DS Tree and ISAX2+ have similar TLB

ISAX2+ is slower than DS Tree

No bias, same data and same implementation framework

341

Insights

- Results are sensitive to:
  - Parameter tuning
  - Hardware setup
  - Implementation
  - Workload selection
- Results identify methods that would benefit from modern hardware

342

Time Series Management Systems

343

Storing Time-Series

Multiple options. By popularity:

344

Storing Time-Series: File-System

Multiple different formats implemented for various applications

345

Storing Time-Series: DBMS

**Illustra (1993) → IBM Informix (Time-Series DataBlade):**

- Users need to define a time-series sub-type, which have a datetime as the first column in the definition
- Can encode both regular and irregular time-series (fixed of variable intervals)
- Can describe meta-data
- Supports: running aggregates, prev, next value reasoning, horizontal and vertical mathematical operations, lags, etc.

**Shore → SEQ**

- Custom Time-Series Data Type
- Various time-series operators (order, correlation, etc.)

**Oracle:**

- Introduced Time-Series functionality in Oracle8
- Now merged into the main product.
- It is in the form of time-series analytics functions (e.g., forecasting)

346

Storing Time-Series: DBMS

**Illustra (1993) → IBM Informix (Time-Series DataBlade):**

- Users need to define a time-series sub-type, which have a datetime as the first column in the definition
- Can encode both regular and irregular time-series (fixed of variable intervals)
- Can describe meta-data
- Supports: running aggregates, prev, next value reasoning, horizontal and vertical mathematical operations, lags, etc.

**Shore → SEQ**

- Custom Time-Series Data Type
- Various time-series operators (order, correlation, etc.)

**Oracle:**

- Introduced Time-Series functionality in Oracle8
- Now merged into the main product.
- It is in the form of time-series analytics functions (e.g., forecasting)

Most people use DBMSs merely for storing and retrieving time-series. All analysis is performed externally.

347

Storing Time-Series: Specialized Time-Series DBs

348

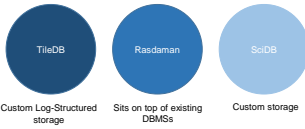
### Storing Time-Series: ArrayDBs



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 349

349

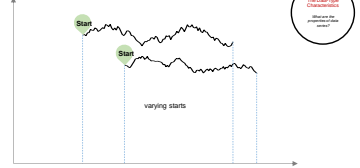
### Storing Time-Series: ArrayDBs



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 350

350

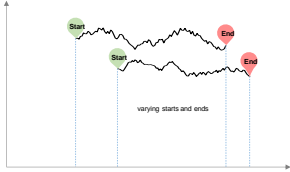
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 351

351

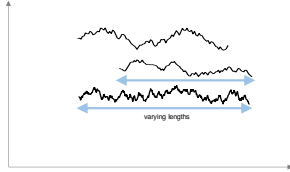
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 352

352

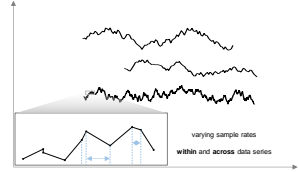
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 353

353

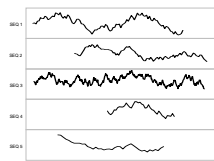
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 354

354

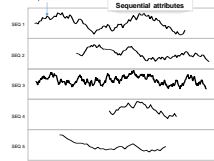
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 355

355

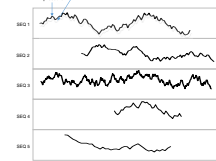
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 356

356

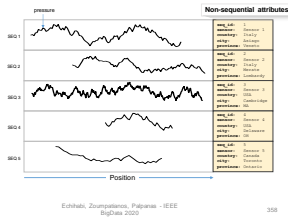
### Time-Series Characteristics



Ethiabi, Zoumpatianos, Paloukas - IEEE BigData 2020 357

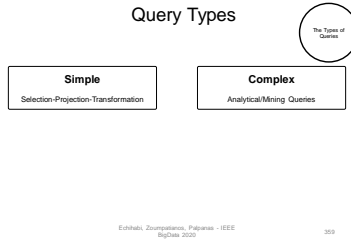
357

### Time-Series Characteristics



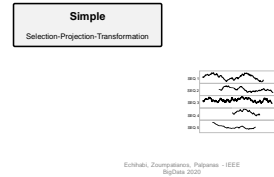
358

### Query Types



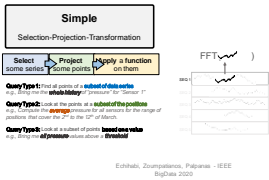
359

### Query Types



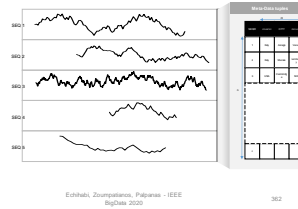
360

### Query Types



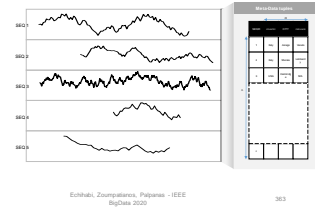
361

### Storage



362

### Storage



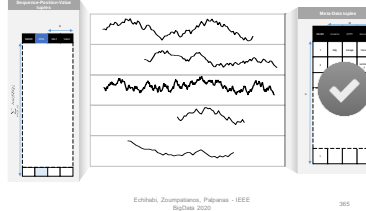
363

### Storage



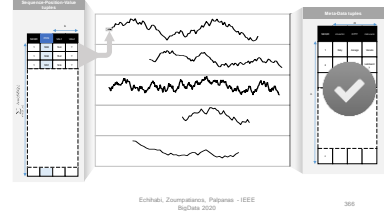
364

### Schema

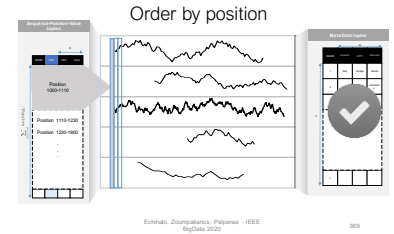
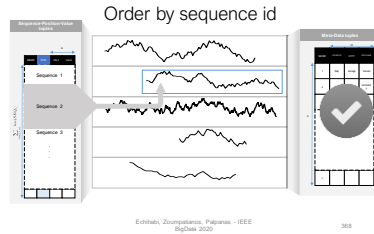
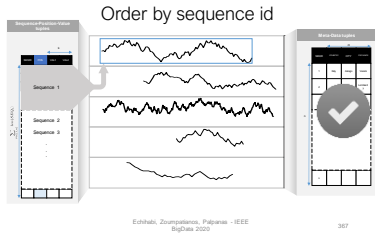


365

### Schema



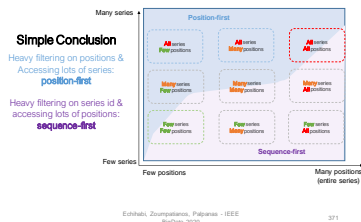
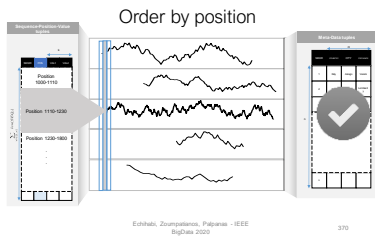
366



367

368

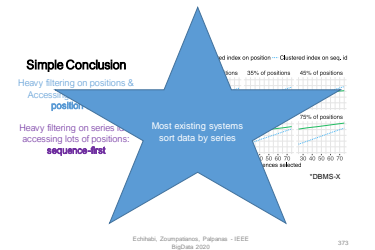
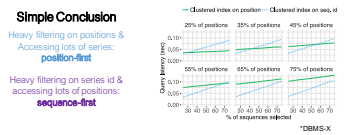
369



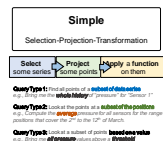
370

371

372

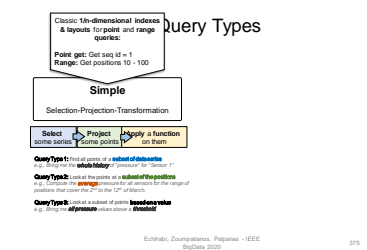


Query Types



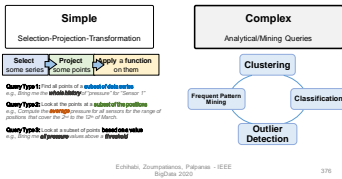
373

374



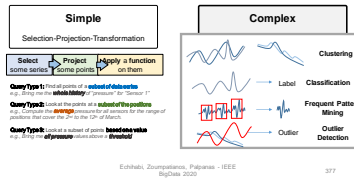
375

Query Types



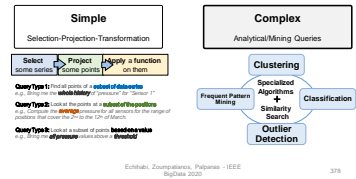
376

Query Types



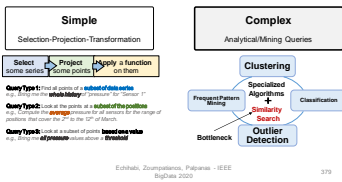
377

Query Types



378

Query Types



379

Time-Series Management Systems

a few more details on the popular systems:  
- InfluxDB  
- TimescaleDB

Edithaki, Zoumpatou, Paloukas - IEEE BigData 2020 380

380

InfluxDB

- Storage Engine:
  - Log Structured Merge Tree: LSM-Tree variant that expects data to arrive ordered by time and partitions them by distinct sequence. It then stores each series contiguously.
- Schema:
  - Tags and fields. Tags are used to describe meta-data and fields are used to store quantities that change over time.
- Queries
  - It supports group by (only on tags), join (on timestamps and fields), selections, projections, and aggregations.
  - It also supports continuous queries

Edithaki, Zoumpatou, Paloukas - IEEE BigData 2020 381

381

TimescaleDB

- Storage: Uses PostgreSQL as the backend.
  - It partitions time-series into multiple tables, forming a single virtual entity called a hypertable.
  - It allows for the compression of data, something that Postgres does not do by default.
- Schema: Tables are normal Postgres tables, where one has to specify a time column in order to create a hypertable.
- Queries: Full SQL support, with the addition of custom time-series functions.
  - Custom time-series operators: first, last, histogram, interpolation, time bucketing, gap filling, etc.
  - It also supports continuous queries

Edithaki, Zoumpatou, Paloukas - IEEE BigData 2020 382

382

Challenges and Open Problems

Challenges and Open Problems

- we are still far from having solved the problem
- several challenges remain in terms of
  - usability, ease of use
  - scalability, distribution
  - benchmarking
- these challenges derive from modern data series applications

Edithaki, Zoumpatou, Paloukas - IEEE BigData 2020

384

383

### Massive Data Series Collections

**NASA's Solar Observatory Telescope (2019)**  
1.5 TB per day  
Large Synoptic Survey  
~30 TB per night

**Human Genome Project**  
130 TB

**passenger aircrafts**  
20 TB per hour

**data center and services monitoring**  
2B data series  
4M points/sec

**facebook**

© IBM, Southampton, Palomar - IEEE Rights 2020

385

### Outline

- sequence management system
- benchmarking
- interactive analytics
- parallelization and distribution
- general high-dimensional vectors
- deep learning

© IBM, Southampton, Palomar - IEEE Rights 2020

386

### Management System

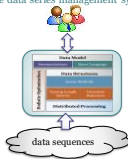
*"enable practitioners and non-expert users to easily and efficiently manage and analyze massive data series collections"*

© IBM, Southampton, Palomar - IEEE Rights 2020

387

### Management System

- Big Sequence Management System
  - general purpose data series management system

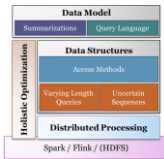


© IBM, Southampton, Palomar - IEEE Rights 2020

388

### Management System

- Big Sequence Management System

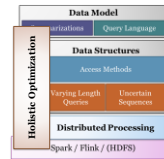


© IBM, Southampton, Palomar - IEEE Rights 2020

389

### Management System

- Big Sequence Management System



© IBM, Southampton, Palomar - IEEE Rights 2020

390

### Management System

- Big Sequence Management System

Dataset	ID	Scenario	Event	Event	Event	Event	Event
Simul	A	D	S	D	D	D	D
Large	A	S	S	M	M	M	M
Advers	A	U	V	V	V	V	V
DeepL	A	U	V	D	D	D	D
SATL	A	D	I	D	D	D	D
Simul	A	S	S	M	M	M	M
Simul	S	D	I	D	I	D	I
Large	S	S	S	M	M	M	M
Advers	S	V	V	V	V	V	V
DeepL	S	V	V	V	V	V	V
SATL	S	V	V	V	V	V	V
Simul	S	V	V	D	D	D	D

© IBM, Southampton, Palomar - IEEE Rights 2020

391

### Management System

- Big Sequence Management System

Dataset	ID	Scenario	Event	Event	Event	Event	Event
Simul	A	D	S	D	D	D	D
Large	A	S	S	M	M	M	M
Advers	A	U	V	V	V	V	V
DeepL	A	U	V	D	D	D	D
SATL	A	D	I	D	D	D	D
Simul	A	S	S	M	M	M	M
Simul	S	D	I	D	I	D	I
Large	S	S	S	M	M	M	M
Advers	S	V	V	V	V	V	V
DeepL	S	V	V	V	V	V	V
SATL	S	V	V	V	V	V	V
Simul	S	V	V	D	D	D	D

© IBM, Southampton, Palomar - IEEE Rights 2020

392

### Outline

- sequence management system
- benchmarking
- interactive analytics
- parallelization and distribution
- general high-dimensional vectors
- deep learning

© IBM, Southampton, Palomar - IEEE Rights 2020

393

### Previous Studies

evaluate performance of indexing methods using random queries  
 • chosen from the data (with/without noise)



Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

394

### Previous Studies

With or without noise



Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

395

### Problem with Random Queries



No control on their characteristics

We cannot properly evaluate summarizations and indexes

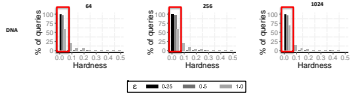
We need queries that cover the entire range from easy to hard

Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

396

### Previous Workloads

Most previous workloads are skewed to easy queries

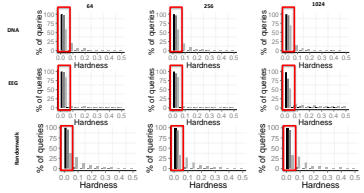


Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

397

### Previous Workloads

Most previous workloads are skewed to easy queries



Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

398

### Benchmark Workloads

If all queries are easy all indexes look good



If all queries are hard all indexes look bad



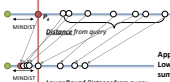
need methods for generating queries of varying hardness



Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

399

### Characterizing Queries

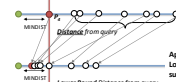


Approximating distances using Lower Bounding functions on summarizations.

Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

400

### Characterizing Queries



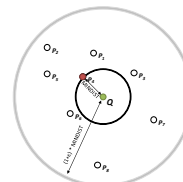
Approximating distances using Lower Bounding functions on summarizations.

Points with lower bounds below MINDIST cannot be pruned  
 Must be read from disk in order to dismiss false positives

Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

401

### Hardness



Ethiabi, Zoumpatianos, Pajares - IEEE BigData 2020

402



### Hardness

We define an  $\epsilon$ -area  
 $(1+\epsilon) * \text{MINDIST}$

**Hardness**  
 # of data-series in  $\epsilon$ -area  
 # all data series

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

403

### Hardness

**Significance**  
 Queries with larger hardness tend to have a larger minimum effort

data series close to the answer  
 higher chance that their lower bounding distance will be less than MINDIST

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

404

### Workload Generation

Random queries have random hardness

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

405

### Workload Generation

Can we generate queries of controlled hardness?

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

406

### 3 Step Process

**Sample**  
Random queries from a given dataset

**Filter**  
Subset of queries that have "independent"  $\epsilon$ -areas

**"Densify"**  
 $\epsilon$ -areas to reach given hardness

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

407

### Step 1: Sampling

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

408

### Step 2: Filtering-out "intersecting" queries

We need to **independently** control the  $\epsilon$ -areas

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

409

### Step 2: Filtering-out "intersecting" queries

The  $\epsilon$ -areas of  $(Q_i, Q_j)$  and  $(Q_k, Q_l)$  cannot be **independently controlled** because they **intersect**

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

410

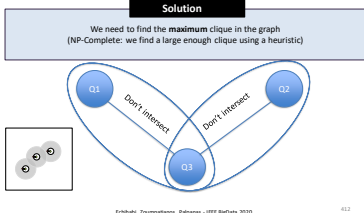
### Step 2: Filtering-out "intersecting" queries

Can be formulated as a **graph problem**  
**1 node** per query  
**1 edge** for each pair that doesn't intersect

EthHabi, Zoumpatianos, Palaganas - IEEE BigData 2020

411

### Step 2: Filtering-out "intersecting" queries



412

### Step 3: Densifying Number of data series to add

- Given a set of hardnesses as input
- We decide the number of data series to add for each query by solving a linear system of equations:

$$a_i = \frac{N_i + x_i}{N + \sum_{j=1}^n x_j}$$

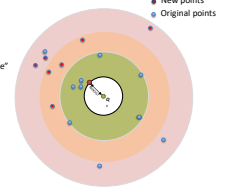
- $\alpha_i$ : hardness,
- $X_i$ : number of data series to add
- $N_i$ : number of data series already in e-area
- $N$ : Total number of data series

413

### Densification Method: Equi-densification

Distribute points such that:  
The worse a summarization the more data it checks

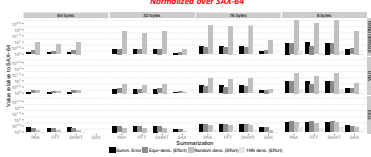
Equal number of points in every "zone"



414

### Experiments Densification Methods

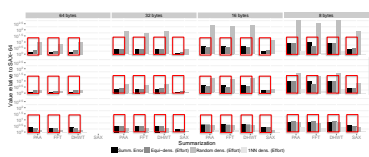
- Using all datasets with size 256 (100 queries for each dens. method), we measured the:
- **I-TLB: Summarization Error** (0: perfect bound, 1: worst possible bound)
  - **Minimum Effort** for a set of summarizations using  $\theta = 64$  bytes.



415

### Experiments Densification Methods

- For equi-densification **normalized Effort** is closer to the **normalized Summarization Error**
- The worse a summarization the bigger effort it does



416

### Summary

**Pros:**

**Theoretical background**  
Methodology for characterizing NN queries for data series indexes

**Nearest neighbor query workload generator**  
Designed to stress-test data series indexes at varying levels of difficulty

**Cons:**

**Time complexity**  
Need new approach to scale to very large datasets

417

### Outline

- sequence management system
- benchmarking
- **interactive analytics**
- parallelization and distribution
- general high-dimensional vectors
- deep learning

418

### Interactive Analytics?

- data series analytics is **computationally expensive**
  - very high inherent complexity
- may not always be possible to remove delays
  - but could try to hide them!

419

### Need for Interactive Analytics

- interaction with users offers **new opportunities**
  - **progressive** answers
    - produce intermediate results
    - iteratively converge to final, correct solution

420

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution

421

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution

422

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution

423

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

424

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

425

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

426

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

427

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

428

**Need for Interactive Analytics**

- interaction with users offers **new opportunities**
  - progressive answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

429

ciN 430

### Need for Interactive Analytics

- interaction with users offers **new opportunities**
  - progressive** answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way

Publications: Sig'09, 12, VLDB 12

430

ciN 431

### Need for Interactive Analytics

- interaction with users offers **new opportunities**
  - progressive** answers
    - produce intermediate results
    - iteratively converge to final, correct solution
    - provide bounds on the errors (of the intermediate results) along the way
- several exciting **research problems** in intersection of visualization and data management
  - frontend**: HCl/visualizations for querying/results display
  - backend**: efficiently supporting these operations

Publications: Sig'09, 12, VLDB 12

431

ciN 432

### Need for Parallelization/Distribution

- take advantage of all modern hardware opportunities!**
  - Single Instruction Multiple Data (SIMD)
    - natural for data series operations
  - multi-tier CPU caches
    - design data structures aligned to cache lines
  - multi-core and multi-socket architectures
    - use parallelism inside each computation server
  - Graphics Processing Units (GPUs)
    - propose massively parallel techniques for GPUs
  - new storage solutions: NVRAMs, FPGAs
    - develop algorithms that take these new characteristics/tradeoffs into account
  - compute clusters
    - distribute operation over many machines

Publications: HPS'11

432

ciN 433

### Outline

- sequence management system
- benchmarking
- interactive analytics
- parallelization and distribution**
- general high-dimensional vectors
- deep learning

Publications: HPS'11

433

ciN 434

### Need for Parallelization/Distribution

- further scale-up and scale-out possible!**
  - techniques inherently parallelizable
    - across cores, across machines

The diagram shows a multi-processor system with 'compute nodes' and 'compute node number'. It illustrates 'parallelized data series' and 'data series collection'. A note states: 'ends computation early, based on information from other nodes'. Another note indicates: 'subset of collection that contains the #SERIES'.

Publications: HPS'11

434

ciN 435

### Need for Parallelization/Distribution

- further scale-up and scale-out possible!**
  - techniques inherently parallelizable
    - across cores, across machines
- need to**
  - propose methods for concurrent query answering
  - combine multi-core and distributed methods
  - examine FPGA and NVM technologies
- more involved solutions required when optimizing for energy**
  - reducing execution time is relatively easy
  - minimizing total work (energy) is more challenging

Publications: HPS'11

435

ciN 436

### Outline

- sequence management system
- benchmarking
- interactive analytics
- parallelization and distribution
- general high-dimensional vectors**
- deep learning

Publications: HPS'11

436

ciN 437

### Data Series vs. high-d Vectors

- two sides of the same(?) coin**
  - data series as multidimensional points
  - for a specific ordering of the dimensions

Publications: HPS'11

437

ciN 438

### Data Series vs. high-d Vectors

- two sides of the same(?) coin**
  - data series as multidimensional points
  - for a specific ordering of the dimensions
- several techniques for similarity search in high-d vectors**
  - using LSH (SRFS), space quantization (IMI), k-NN graphs (HNSW)

Publications: HPS'11

438

ciNO 439

### Locality Sensitive Hashing (LSH) Indexing

**Locality Sensitive Hashing**  
(with distance  $d$ )

- Random hash function  $g$  on  $\mathbb{R}^d$  s.t. for any points  $p, q$ :
  - Close when  $\|p - q\| \leq r$
  - Far when  $\|p - q\| > cr$
- $P_1 = \Pr(g(p) = g(q))$  is "not-so-small"
- $P_2 = \Pr(g(p) = g(q))$  is "small"
- Use several hash tables:  $m$ , where  $\rho = \frac{\log 1/P_1}{\log 1/P_2}$

\*Alan Ailon, Locality Sensitive Hashing, Summer School on Hashing, 2014.

439

ciNO 440

### Locality Sensitive Hashing (LSH) Search

- A kNN query  $Q$  arrives
- The hash functions applied to the dataset are applied to  $Q$
- Points that fall at least once in the same bucket as  $Q$  are further processed in a linear scan
- The list of  $k$   $\delta$ -approximate nearest-neighbor is returned

440

ciNO 441

### Inverted Multi-Index (IMI)

441

ciNO 442

### k-Nearest Neighbor Graphs (kNNGs) Indexing

Each object is connected to its  $k$  most similar objects

\*Lin Z., Barabasi, M. Graph-based data clustering via multilink community detection, Appl Netw Sci 3 (2016)

442

ciNO 443

### k-Nearest Neighbor Graphs (kNNGs) Search

- A kNN query  $Q$  arrives
- A random vertex  $R$  is selected
- The closest neighbors of  $R$  to  $Q$  are put in a candidate list
- Each candidate node is visited and its neighbors are put in the candidate list until greedy termination condition is met
- The list of  $k$  ng-approximate neighbors are returned

443

ciNO 444

### Data Series vs. high-d Vectors

- two sides of the same(?) coin
  - data series as multidimensional points
  - for a specific ordering of the dimensions
- several techniques for similarity search in high-d vectors
  - using LSH (SRS), space quantization (IMI), k-NN graphs (HNSW)
- how do these high-d vector techniques compare to data series techniques?
  - currently conducting extensive experimental comparison

444

ciNO 445

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data

445

ciNO 446

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk

(s) Deep25GB(ng) (t) Deep25GB( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, SA-X2, SRS, VA-File

446

ciNO 447

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk

(s) Deep25GB(ng) (t) Deep25GB( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, SA-X2, SRS, VA-File

447

ciN 448

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk

(s) Deep25GB (ng) (t) Deep25GB ( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, ISAX2+, SRS, VA-file

448

ciN 449

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk
  - perform the best for long vectors, in-memory and on-disk

(g) Rand25GB 16384 (ng) (h) Rand25GB 16384 ( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, ISAX2+, SRS, VA-file

449

ciN 450

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk
  - perform the best for long vectors, in-memory and on-disk

(g) Rand25GB 16384 (ng) (h) Rand25GB 16384 ( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, ISAX2+, SRS, VA-file

450

ciN 451

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk
  - perform the best for long vectors, in-memory and on-disk
  - perform the best for disk-resident vectors

(m) Deep250GB (ng) (n) Deep250GB ( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, ISAX2+, SRS, VA-file

451

ciN 452

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
  - perform the best for approximate queries with probabilistic guarantees ( $\delta$ - $\epsilon$ -approximate search), in-memory and on-disk
  - perform the best for long vectors, in-memory and on-disk
  - perform the best for disk-resident vectors

(m) Deep250GB (ng) (n) Deep250GB ( $\delta\epsilon$ )

Legend: DS-Tree, HNSW, IMI, ISAX2+, SRS, VA-file

452

ciN 453

### Data Series vs. high-d Vectors

- data series techniques are the overall winners, even on general high-d vector data
- several new applications (and challenges) for data series similarity search techniques!

453

ciN 454

### Outline

- sequence management system
- benchmarking
- interactive analytics
- parallelization and distribution
- general high-dimensional vectors
- deep learning

454

ciN 455

### Connections to Deep Learning

- data series indexing for deep embeddings

455

ciN 456

### Connections to Deep Learning

- data series indexing for deep embeddings

sequences  
text  
images  
video  
graphs  
...

456

ciNO 457

### Connections to Deep Learning

- data series indexing for deep embeddings

sequences  
text  
images  
video  
graphs  
...

deep embeddings  
high-d vectors learned using a DNN

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

457

ciNO 458

### Connections to Deep Learning

- data series indexing for deep embeddings

sequences  
text  
images  
video  
graphs  
...

deep embeddings  
high-d vectors learned using a DNN

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

458

ciNO 459

### Connections to Deep Learning

- data series indexing for deep embeddings
  - deep embeddings are high-d vectors
  - data series techniques provide effective/scalable similarity search

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

459

ciNO 460

### Connections to Deep Learning

- data series indexing for deep embeddings
  - deep embeddings are high-d vectors
  - data series techniques provide effective/scalable similarity search
- deep learning for summarizing data series
  - eg, autoencoders can learn efficient data series summaries

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

460

ciNO 461

### Connections to Deep Learning

- data series indexing for deep embeddings
  - deep embeddings are high-d vectors
  - data series techniques provide effective/scalable similarity search
- deep learning for summarizing data series
  - eg, autoencoders can learn efficient data series summaries
- deep learning for designing index data structures
  - learn an index for similarity search

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

461

ciNO 462

### Connections to Deep Learning

- data series indexing for deep embeddings
  - deep embeddings are high-d vectors
  - data series techniques provide effective/scalable similarity search
- deep learning for summarizing data series
  - eg, autoencoders can learn efficient data series summaries
- deep learning for designing index data structures
  - learn an index for similarity search
- deep learning for query optimization
  - search space is vast
  - learn optimization function

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

462

ciNO 463

### Conclusions

- data series is a very **common** data type
  - across several different domains and applications

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

463

ciNO 464

### Conclusions

- data series is a very **common** data type
  - across several different domains and applications
- **complex data series analytics are challenging**
  - have very high complexity
  - efficiency comes from data series management/indexing techniques

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

464

ciNO 465

### Conclusions

- data series is a very **common** data type
  - across several different domains and applications
- **complex data series analytics are challenging**
  - have very high complexity
  - efficiency comes from data series management/indexing techniques
- **need for Sequence Management System**
  - optimize operations based on data/hardware characteristics
  - transparent to user

©Muhimbi Groupware, Inc. All Rights Reserved. 10/2018

465





## References

475

- Anna Cogolin, Theodoros Tzoufas, Karima Echihabi, Anastasia Roussiou, Themis Palpanas: Data Series Progressive Similarity Search with Probabilistic Quality Guarantees. SIGMOD Conference 2020: 4527-4537
- Themis Palpanas. Evolution of a Data Series Index - The ISAX Family of Data Series Indices. CCSR, 197 (2018)
- Djani Edina Yagoubi, Reza Akbarinia, Florent Masoaglia, Themis Palpanas: Massively Distributed Time Series Indexing and Querying. IEEE Trans. Knowl. Data Eng. 34(1): 108-120 (2020)
- Batao Peng, Panagiotá Fakoussou, Themis Palpanas: MESS: In-Memory Data Series Indexing. ICDM 2020: 2175-2188
- Kaidong Feng, Peng Wang, Jinye Wu, Wei Wang, Li-Math: A Lightweight and Effective Subsequence Matching Approach. IEEE Access 8: 71173-71 (2020)
- Batao Peng, Panagiotá Fakoussou, Themis Palpanas: Paria - Data series indexing on multi-core architectures. TKDE, 2020
- Michalis Liarohis, Themis Palpanas: Scalable Data Series Subsequence Matching with CLASSE. VLDBJ 2020
- John Paparinos, Chaoxi Liu, Aaron J. Elmore, Michael J. Franklin: Debunking Four Long-Standing Misconceptions of Time-Series Distance Measures. SIGMOD Conference 2020
- Karima Echihabi, Kostas Zoumpatianos, and Themis Palpanas: Scalable Machine Learning on High-Dimensional Vectors: From Data Series to Deep Network Embeddings. In WDM, 2020
- Batao Peng, Panagiotá Fakoussou, Themis Palpanas: SNG: Sequence Indexing Using GPUs. KDD, 2020

Echihabi, Zoumpatianos, Palpanas: 1930 Rights man

475

## References

476

- InfluxDB: <https://www.influxdata.com/>
- Timescale: <https://www.timescale.com>
- Beringel: <https://github.com/facebookarchive/beringel>
- Druid: <https://druid.apache.org>
- Prometheus: <https://prometheus.io>
- CrateDB: <https://crate.io>
- IoTDB: <https://iotdb.apache.org>
- OpenTSDB: <https://opentsdb.net/>
- QuasarDB: <https://www.quasardb.net/>
- Timestream: <https://aws.amazon.com/timestream/>

Echihabi, Zoumpatianos, Palpanas: 1930 Rights man

476