Visualizing and Discovering Nontrivial Patterns In Large Time Series Databases

Jessica Lin, Eamonn Keogh, Stefano Loardi

Presented By

Nicholas Chen
Samah Ramadan
Time Series

• What?
  – sequences of values or events changing with time

• Why?
  – Applications
    • Medicine: ECG, EEG
    • Finance: stock market, credit cards
    • Aerospace: launch telemetry, satellite sensor
    • Entertainment: music, movies
Data mining Time series

• Why?
  – Trend Analysis
  – Similarity Search

• What tasks?
  1. **Sequence matching**: whole / subsequence / chunking
  2. **Anomaly detection**: deviation from normal
  3. **Motif discovery**: overrepresentation
VizTree

- **What?**
  - Visualization tool for time series data
  - Based on *subsequence* trees

- **How?**
  - Time series $\xrightarrow{\text{VizTree}}$ Symbolic representation
  - Symbolic representation $\xrightarrow{\text{VizTree}}$ Tree representation
Previous Approaches

• Cluster and calendar based visualization
  – Time series $\rightarrow$ Sequence of day patterns
  – By bottom-up clustering algorithm
  – Limitations: calendar pattern data, prior knowledge of patterns

Figure 4. Calendar view of the number of employees
Previous Approaches (cont..)

• Spiral
  – Periodic section of time → one ring
  – Data values → color and line thickness
  – Limitations: Data should be periodic (known period)

*Figure 3*: The Spiral visualization approach of Weber et al. applied to the power usage dataset.
Previous Approaches (cont..)

• TimeSearcher
  – Query-by-content
  – Flexible
  – User must specify query regions (must know what to look for)
  – Scalability issues
VizTree Example

• An interesting problem
  – Two sets of binary sequences of length 200 were generated
  – One sequence generated by a pseudo-random-number generator by the computer
  – The other was generated by hand by a group of volunteers
VizTree Example

- Can you tell who generated which?
- VizTree can!
  - Subsequence tree representations for all sets of 3 digits in each sequence.
Discretizing Time Series

Problem: Most time series are not discrete
• Must convert real-valued data to symbols
• Symbolic Aggregate approxXimation, SAX
  – Lower bound symbolic space
  – Feasible approximation for large databases
• Normalization before discretization (usually)
A subsequence $C$ is extracted by a sliding window of length $n$

- Each window is divided into $w$ equal-sized regions
- Average the data points in each region
- The average will fall into one of $\alpha$ levels (alphabet size)
- A symbol corresponds to each level (a, b, c, d … )

Example: Above window corresponds to $a\ c\ d\ c\ b\ d\ b\ a$
A Sample Tree

α possible levels

α × w leaves (independent of time series length)

w regions
VizTree in Action

• Subsequence matching
  – Would like to find patterns that have certain characteristics
  – Do this by specifying a path of nodes through the big tree

• Demo
  – Find certain patterns in heartbeat pattern
  – Notice interactive detail view
VizTree In Action II

• Finding Motifs
  – VizTree is very good at showing commonly occurring motifs
  – Simply look at thick branches

• Demo
  – Look at what most weeks look like in power consumption
  – Can step through, or go directly to a motif
VizTree In Action III

• Finding simple anomalies
  – One way is to do the opposite of finding motifs
  – Go through all the thin lines

• Demo
  – Locate weeks where power consumption is unusual
  – Also, locate where heart beat is irregular
VizTree In Action IV

• Diff-Tree
  – For more complex anomalies
  – Compare a times series against a reference time series
  – Three concepts
    • Difference in frequencies (Blue or Green)
    • Confidence (Luminosity)
    • Difference x Confidence = Surprisingness (Red)

• Demo
  – Two similar data sets, find areas where they differ
  – VizTree can rank surprisingness
Numerosity Reduction

- Fancy term for reducing the noise by removing trivial patterns
- Consecutive windows are often similar or identical.
- Results in overcounting, can obscure differences

All 3 windows will be “medium – low - high”
VizTree Criticisms

• While exploring VizTree to prepare demos, noticed a couple of issues:
  – Atrocious time series UI
  – Parameter values somewhat of a black art
  – Phase issues
  – Hierarchical tree structure is somewhat misleading
VizTree

• Advantages:
  – Scalable to large data sets
  – Good for tasks it is designed for (finding motifs, anomaly detection, high level sequence search)

• Disadvantages:
  – Not so good for other data mining tasks
  – Not completely intuitive – need to think in terms of the program
  – Settings are arbitrary and dataset dependant