MULTI-LEVEL APPROACH TO SENSOR STREAMS ANALYSIS

Klemen Kenda, Luka Stopar, Marko Grobelnik
Artificial Intelligence Laboratory
Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
e-mail: klemen.kenda@ijs.si

1 INTRODUCTION

Monitoring of the systems, which are described with numerous time series, can be a complex task. Too much data is difficult to follow even by an expert human user. In presented work we focused on understanding dynamics of such complex systems and presenting the results in a humanly-comprehensible way.

Instead of following the dynamics of a system through numerous time series, the result of our methodology is a directed state graph (see Figure 1) equipped with corresponding transitional probabilities. To achieve such a result we extract different features (markers) from the time series, aggregate them in a sliding window and pack them into state vectors. We perform clustering on top of a set of such vectors and calculate transitional probabilities between the clusters.

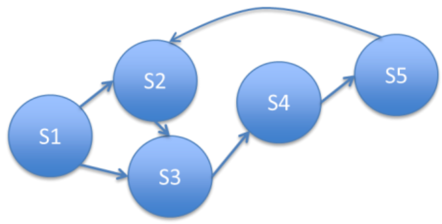

Figure 1: Internal states are clustered, transitional probabilities are calculated.

Typical states (centroids) are identified by domain experts. Such knowledge base can be later used for anomaly detection, root cause analysis and other tasks.

2 METHODOLOGY

Each system is observed by multiple sensors that output regular measurements forming multiple time series. In an energy forecasting setting, for example, such sensors could be sensors measuring weather conditions (temperature, pressure, wind direction, wind speed, cloud cover, humidity), sensors measuring energy demand and consumption and virtual sensors »measuring« day of week, time of day, working hours, holidays, sunrise, sunset and other relevant features

Each time series could have different marker extractors attached. These marker extractors give us certain aggregated information of the time series in a specified time window. Examples of typical markers are: local minimum and maximum, average value, maximum derivative, value threshold markers (is value in a certain range) etc.

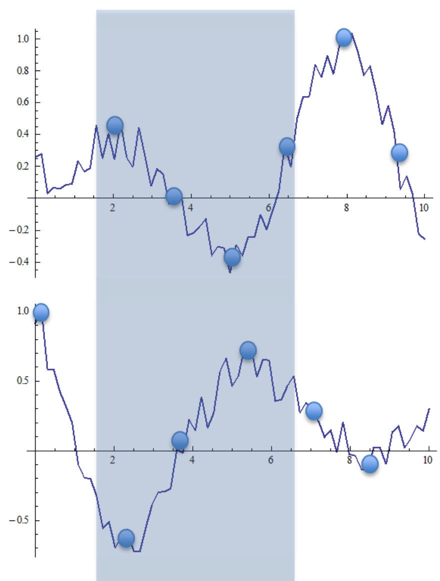

Figure 2: Time series are enriched with the markers, which are aggregated within the slidina window.

Next, we count appearance of the markers within a certain time window (which is implemented as a sliding window) as depicted in Figure 2. Values and/or counts of the markers are collected in a vector of features and those are collected for different times (see Figure 3). Clustering is performed on a minimal set of such feature vectors and initial transitional probabilities are calculated. Updating of the transitional probabilities and clusters is later done in an on-line manner.

Time	F_1	F_2	F_3	F_4	F_5	F_6
t_0	*	*	*	*		*
t_1	*	*		*		
…						
t_n	…	…	…	…	…	…

Figure 3: For each sliding window aggregated features represent internal state of the system.

Graph of the system offers human-managable view on the system and gives us a deeper understanding of the dynamics of the system. Many different upgrades are possible on top of such an approach.

3 POSSIBLE APPLICATIONS

The most obvious application could be **anomaly detection.** In case a new state of the system does not fit in any of the clusters, this might be a signal for an anomaly. Anomaly could represent another yet undiscovered typical state of the system and therefore the model would need updating or it could be a real anomaly.

Next logical step – as we do have a directed graph and history of transitions – would include **root cause analysis.** When an anomaly occurs we can trace back and identify the path which lead to the anomaly. Furthermore we could detect typical paths in the system that lead to such anomalies and perform **proactive anomaly detection.** This means that we could detect anomalies before they would happen, based on a subsequence path matching.

Another application might be in usage of the clustering for modelling, where we would use different models for different clusters. Thermal plant demand for example differs significantly between heating season and summer.

4 EARLY RESULTS, CONCLUSIONS AND FUTURE WORK

The idea was tested on a dataset of a thermal plant in Reggio nell'Emilia, Italy. Features were extracted from daily thermal production profiles. Clustering identified typical states of the system, annotated by domain experts, such as: early heating season, transitional season (spring, autumn) or high heating season with a variety of substates.

Early results are promising and can be viewed as a proof of concept. More complex systems need to be analyzed. On the other hand other clustering algorithms need to be tested and more markers (extractors) need to be introduced and tested.

Future work includes also development of interactive GUI, which would make it easy for a domain expert to explore and identify clusters at different levels. There is also a vast field of possible applications described in Section 3 that need to be addressed.

This work was supported by the ICT Programme of the EC under NRG4Cast (FP7-ICT-600074) and Proasense (FP7-ICT-612329).