

Supporting Rule Generation and Validation on Environmental Data in EnStreamM

Alexandra Moraru^{1,3}, Klemen Kenda¹, Blaž Fortuna¹, Luka Bradeško¹, Maja Škrjanc¹,
Dunja Mladenčić^{1,3}, Carolina Fortuna²

¹Artificial Intelligence Laboratory, Jožef Stefan Institute, Ljubljana, Slovenia

²Department of Communication Systems, Jožef Stefan Institute, Ljubljana, Slovenia

³Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

{firstname.lastname}@ijs.si

Abstract. Detection rules represent one of the components of the rule models in event processing systems. These rules can be discovered from data using data mining techniques or domain experts' knowledge. We demonstrate a system that provides its users the means for creating and validating such rules. The system is applied on real-life environmental scenarios, where the main source of data comes from sensors. Based on historical data about events of interest, the scope is to formulate rules that could have caused these events. Using a scalable infrastructure the rules can be tested on massive amount of data in order to observe how past events would fit to these rules. In addition, we create semantic annotations of the dataset and use them in the system outputs in order to support interoperability with other systems.

Keywords: Visual analytics, sensor data, rule model, semantic annotations.

1 Introduction

The avalanche of data which information systems have to face nowadays influences their evolution and characteristics. One such family of systems, called information flow processing (IFP) systems [1], refers to data stream management systems and complex event processing systems. Such systems are able to handle multiple data sources, often streams, by applying a set of processing rules in order to derive new knowledge. These rules can be discovered using data mining and machine learning techniques from a vast research area [2,3] or they can be defined by domain experts based on their knowledge. For the second case, an example can be related to landslides phenomena, for which an expert already knows the causes producing landslides. Many of these situations follow specific patterns which can be expressed through rules. The next step to represent these rules in a format which can be used by information systems is to provide to the experts an environment where they can create and validate the rules.

We demonstrate a system which can be used by domain experts to explore large datasets in order to define processing rules for environmental data. The rules can be

created and validated on real datasets through a graphical user interface (GUI). Similar work can be found on visual pattern discovery [4], where the focus is set on time series visualization for detection of unknown events. In contrast, we consider the situation when the events of interest are already known and the possible causes of these events can be explored.

Our system uses EnStreaM infrastructure which is based on tightly integrated and scalable custom software modules. In addition, the indexing of the data is application-oriented, specific and therefore extremely efficient, allowing development of various applications, such as real-time mashups [5]. For interoperability with other systems, rules created in our system are exported in RuleML¹ format, using concepts from OpenCyc² for relations' names. Furthermore, the datasets to which the rules apply are exported in RDF format and annotated with OpenCyc concepts.

The demonstration of the system consists of live running EnStreaM platform with which the visitors can interact through the GUI. For illustrating the functionalities of the system a landslide use case is prepared as presented in Section 2.2. Furthermore, the standardized exports of the systems can be presented for those interested.

2 EnStreaM

EnStreaM is a scalable system which implements efficient storage and retrieval methods for handling large amounts of data, both static and dynamic [6]. It is used in the ENVISION³ project for data stream mining tasks, for several environmental scenarios related to landslides, oil spills and river floods. A high level architecture illustrating the inputs and outputs of EnStreaM used in our demonstration is presented in Fig. 1 and discussed in the flowing subsection.

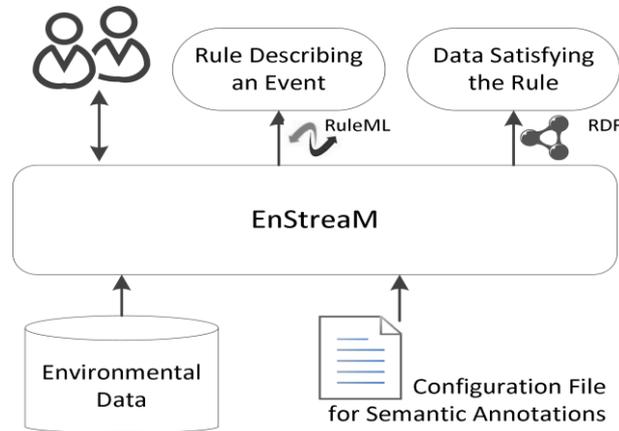


Fig. 1. EnStreaM – overall architecture

¹ <http://ruleml.org/>

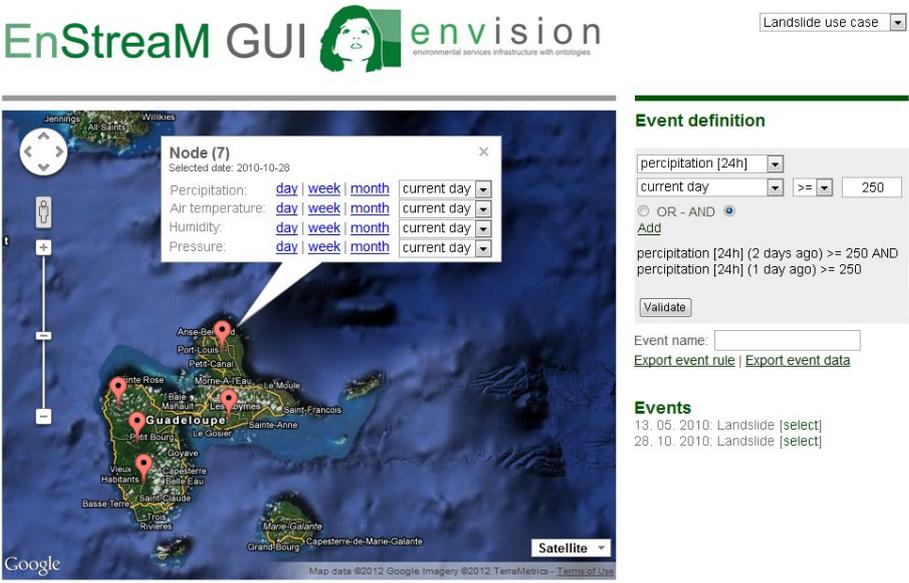
² <http://www.opencyc.org>

³ <http://www.envision-project.eu/>

2.1 System Overview

The input for EnStreamM consists of environmental datasets and configuration files used for annotating the datasets with ontology concepts. The environmental data is composed mostly of sensor measurements of the relevant properties from the area of interest (e.g. volume of rainfall for a given geographical location) and events that have occurred in the past. The metadata attached to the sensor measurements is providing the context needed for understanding these measurements. The sensor measurements have a dynamic nature, while the metadata associated is static. For efficient data management, these two types of data are stored internally in EnStreamM using specialized indexing methods. In order to provide domain experts the possibility to explore archived data in an ad-hoc manner, we index data based on different aspects (e.g. location, date of measurement) and also provide numerous aggregates (sum, min, max, mean, standard variation, etc.). A unified view over different sources of sensor data is created through semantic annotations, based on a configuration file which maps the internal structure of EnStreamM stores to concepts from an ontology.

The abstraction layer provided by the semantic annotations and aggregation of data enables the domain experts to analyze historical data in order to find various patterns. These patterns are represented by rules which can be created and tested through the GUI of the system (see Fig. 2). The process of rule creation can be done in repetitive steps in which the user can refine or add new parameters. For validating the rule, the user can test it on the historical data. Finally, the rule can be exported in RuleML format and the dataset complying with the rules can be exported in RDF format.



The screenshot displays the EnStreamM GUI. At the top left, the logo reads "EnStreamM GUI" with a woman's face icon and the tagline "envision environmental services infrastructure with ontologies". A dropdown menu on the top right is set to "Landslide use case". The main area features a satellite map of Guadeloupe with a "Node (7)" popup window for the date 2010-10-28. This window lists environmental parameters: Precipitation, Air temperature, Humidity, and Pressure, each with options for "day", "week", "month", and "current day". To the right, the "Event definition" panel shows a rule: "precipitation [24h] current day >= 250 AND precipitation [24h] (2 days ago) >= 250 AND precipitation [24h] (1 day ago) >= 250". Below this, there are fields for "Event name", "Validate" button, and "Export event rule" / "Export event data" links. The "Events" section lists two results: "13. 05. 2010: Landslide [select]" and "28. 10. 2010: Landslide [select]".

Fig. 2. EnStreamM User Interface

2.2 Use Case Scenario

To continue with our example from the introduction, let us consider that a landslide domain expert knows that some amount of raindrop can be an alarm for an eminent landslide. For illustration purposes we can consider that a pattern for this is represented by the following rule: *if the amount of rainfall exceeds 250 mm per day in 3 consecutive days then a landslide can occur*. Based on historical data gathered from rain gauge sensors, together with events when landslides have occurred in the past, the validity of such a rule can be verified.

Creation and Validation of Rules

The user can start by analyzing the events which have occurred in the past, listed in the bottom right corner of the interface. Next, the sensors related to the event selected are displayed on the map based on their geographical location. The sensor measurements can be visualized for different time periods as illustrated in Fig. 2. Next, the fields on the right-hand side of the GUI are used to specify the relations and operators to appear in the rules. For our example we have three relations in conjunction (the logic operators supported are “AND” and “OR”) which constitute the conditions of the rule. The result of such conditions being fulfilled represents a type of event, whose name is given by the user in the “Event name” field. The validation step is done by running the query with all the conditions specified over the historical data and comparing the events returned by the query with the list of entire events available for the specified location. The user should decide the importance and quality of the rule defined.

The rules created through the interface are exported in RuleML Datalog format, which provides a simple and clean syntax for expressing “if-then” rules. Each condition is represented by one or more atomic formulas (“Atom”). For example the condition that raindrop exceeds 250 mm per day is represented in our scenario as illustrated in Fig. 3. The export in the RuleML format is depended on the vocabulary used for the relation constants (“Rel”). Specialized domain ontologies can simplify the RuleML representation as they can have more specific relations and concepts.

```
<And> <Atom> <op> <Rel iri="openCyc:sensorObservation"/> </op>
  <Var> sensor </Var>
  <Ind iri="openCyc:Raindrop"/> </Atom>
<Atom>n <op> <Rel iri="openCyc:doneBy"/> </op>
  <Var> sensor </Var>
  <Var> measurement </Var> </Atom>
<Atom> <op> <Rel iri="openCyc:measurementResult"/> </op>
  <Var> measurement </Var>
  <Var> vall </Var> </Atom>
<Atom> <op> <Rel iri="openCyc:duration"/> </op>
  <Var> measurement </Var>
  <Ind type="xs:time">24:00:00</Ind> </Atom>
<Atom> <op> <Rel iri="openCyc:greaterThanOrEqualTo"/> </op>
  <Var> vall </Var>
  <Ind ttype="xs:float">250</Ind> </Atom> </And>
```

Fig. 3. RuleML sample from a rule

Semantic Annotations

The RDF export of datasets corresponding to the rules created is using as model the OpenCyc ontology. We choose to use OpenCyc ontology as it is very large and contains concepts for many specific domains, however, any ontology can be used for annotation as the EnStream infrastructure is not tied to a specific ontology. Since our scenario is closely related to the domain of sensor networks, an alternative for OpenCyc could be the Semantic Sensor Network⁴ ontology to which extension must be added for representing the landslides domain. For the semantic annotation of the datasets corresponding to a rule, the input configuration file is used.

3 Conclusions and Future Work

In this paper we have presented a system for supporting rule generation on environmental data based on EnStream infrastructure. The efficient implementation of data storing and indexing allows the user to interact with the system in timely fashion and makes the system appropriate for demonstration. The use case based on which we demonstrated our system is an environmental scenario using real live data related to landslides phenomena. We plan to extend EnStream for real-time monitoring of streaming data in order to detect the events described in the rules generated. Moreover, other future work includes integrating the rules discovered into knowledge bases used by specific reasoning engines. This will help in semi-automatic extension of knowledge bases, supporting advanced reasoning for problems such as complex events processing, anomaly detection or automatic monitoring.

Acknowledgements. This work was partially supported by the Slovenian Research Agency through the programme P2-0016 and project J2-4197, the competence center KC OPCOMM, and the ICT Programme of the EC under PASCAL2 (ICT-NoE-216886), ENVISION (ICT-2009-249120) and PlanetData (ICT-NoE-257641)

References

1. Cugola, G., Margara, A.: Processing Flows of Information : From Data Stream to Complex Event Processing. ACM Computing Surveys (To Appear)
2. Bishop, C. M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)
3. Mitchell, H. B.: Multi-Sensor Data Fusion: an Introduction. Springer-Verlag, Berlin (2007)
4. Schaefer, M., Wanner, F., Mansmann, F., Scheible, C., Stennett, V., Hasselrot A.T., Keim, D.A.: Visual pattern discovery in timed event data. Proc. SPIE 7868, 78680K (2011)
5. Kenda, K., Fortuna, C., Fortuna, B., Grobelnik, M. Videk: A Mash-up for Environmental Intelligence. AI Mashup Challenge, ESWC (2011)
6. Škrjanc, M., Mladenić, D. Stream mining on environmental data. In Proceedings of Information Society conference IS-2010, volume A, pp. 184-187, Ljubljana, Slovenia, (2010)

⁴ <http://purl.oclc.org/NET/ssnx/ssn>