Combining Event Processing and Support Vector Machines for Automated Flight Diversion Predictions

Cristina Cabanillas, Andreas Curik, Claudio Di Ciccio,*
Manuel Gutjahr, Jan Mendling, Johannes Prescher and Jan Simecka

Institute for Information Business
Vienna University of Economics and Business, Austria
claudio.di.ciccio@wu.ac.at

Extended abstract

Multi-modal logistics chains are those transportation processes in which different modes of transportation are involved for the delivery of goods. These modes are adopted in consecutive *legs*, which have to be synchronized. An example scenario is the delivery of goods from a production center in New York, USA, to a plant in Utrecht, the Netherlands. The transportation chain consists of two legs: *(i)* an aicraft carries the goods from John F. Kennedy International Airport (New York) to Schiphol (Amsterdam) and *(ii)* a truck sent by a *Logistics Service Provider* (LSP) transports the cargo from Dutch airport to Utrecht. The growth in international and inter-continental trade has led to a significant increase of multi-modal transports worldwide. Therefore, guaranteeing its efficiency is of crucial relevance for the successful completion of such transportation processes. In the example scenario, the goal of the LSP is to deliver the goods in time, in conformance with the *Service Level Objectives* (SLOs).

The example scenario presented in this work stems from the experience that the authors gained during the development of the GET (Green European Transport) Service[1] project. GET Service is an ongoing European research project aiming at the enhancement of logistics processes, from the viewpoint of their ecological impact and efficiency.

In this work, the research is focused on transportation processes involving aircrafts. In particular, the objective is to design and realise a service-oriented software architecture allowing for the run-time automated detection of aircrafts' diversions. A diversion consists in the landing of the aircraft in an airport that differs from the planned one. Though rare, diversions can seriously prejudice the successful

*Corresponding author
[1] http://www.getservice-project.eu/

A. Baumgraß, N. Herzberg, G. Kappel, J. Mendling, A. Meyer, S. Rinderle-Ma (eds.): Proceedings on Inter-Organizational Process Modeling and Event Processing in Business Process Management, Vienna, Austria, 12-06-2014, published at http://ceur-ws.org

completion of the transportation process. In the example, adverse weather conditions in the area of Schiphol impose the pilot to make the aircraft land in Brussels. Therefore, the LSP must reroute the truck from Schiphol to the Belgian airport to let goods be delivered to the final destination. In order for these corrective actions to be effective, it is crucial that the LSP is aware of the aircraft diversion as soon as possible. Unfortunately, experience reveals that the communication between LSPs and cargo airlines are not as prompt as required. Specifically, LSPs do not have access to real-time information and are only notified of the diversion once the aircraft has landed at another airport. This delayed notification threatens the ability of LSPs to meet their objectives. For this reason, the approach presented here sets out to reduce the impact of diversions by detecting them in a timely manner, i.e., as soon as an anomalous behaviour is recognised, while the aicraft is still flying. This approach utilises data that are publicly available, i.e., event streams reporting subsequent flight positions, altitude and speed. Thus, it is independent of the communication with airlines.

The architecture of the proposed solution is mainly based on two core modules: *(i)* a module wrapping a complex event processing engine [EN10] and *(ii)* a module based on an automated discriminative classifier [Mit97], namely a Support Vector Machine [CV95]. The latter is meant to read events reporting the flight data, in order to select, filter, aggregate over a time-span and finally transform them in a way that is readable for the former. The automated discriminative classifier is in charge of discriminating which processed flight data represent a normal behaviour from those that show the characteristics of a diverted flight. Support Vector Machines (SVMs) are *supervised learning models*, in the sense that they classify new data on the basis of a previous training phase, performed on already classified historic information. Namely, they define their decision function on the basis of input data that were previously labelled with the category they belong to. The training sessions are known to require a higher computational effort, whereas the classification on new data is a by far lighter operation.

The work of the authors in GET Service has led to promising results in the usage of SVM-based systems for the automated detection of diverted flights, in terms of accuracy. However, the tests have been conducted on log files containing a dump of collected events, thus disregarding the computational issues that may arise in a on-line elaboration of data. Therefore, given the number of flights which can be potentially monitored (order of hundreds of thousands) and the update rate of their position (order of per-second), the core modules are decoupled and deployed as services on two separate architectural components, in a distributed environment. In the presented approach, the classifier module acts as a service for the event processing engine. The event processing engine module is a service as well, invoked by client applications in order to monitor given flights.

A further advantage brought by the logical and physical decoupling of the service-oriented architecture resides in the opportunity of retraining the classifier, without empairing the time performance of the classification task. Indeed, the historic information can be enriched by newly categorised flight data. Thus, the training set can be extended and used for a new training phase, possibly increasing the accuracy of the classifier in a continuous improving fashion.

The core modules in the proposed architecture have been respectively realised adopting Esper,[2] a well-known Java-based framework for complex event processing, and Scikit-learn,[3], a Python-based framework providing implemented Machine Learning algorithms. The prototype has been deployed in two separate web-based containers on Amazon EC2 platform. The approach has been validated connecting the event processing module to the Flightstats API[4], providing updated information on current flights.

Keywords: Complex Event Processing, Support-Vector Machines, Machine Learning, Prediction Model, Aircraft

Acknowledgements

The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement 318275 (GET Service).

References

[CV95] Corinna Cortes and Vladimir Vapnik. Support-Vector Networks. *Machine Learning*, 20(3):273–297, 1995.

[EN10] Opher Etzion and Peter Niblett. *Event Processing in Action*. Manning Publications Company, 2010.

[Mit97] Thomas M. Mitchell. *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition, 1997.

[2] http://esper.codehaus.org/
[3] http://scikit-learn.org/stable/
[4] https://developer.flightstats.com/