# **Run-Time Adaptation of Complex Event Forecasting**

Manolis Pitsikalis<sup>1</sup>, Elias Alevizos<sup>1,2</sup>, Nikos Giatrakos<sup>3</sup> and Alexander Artikis<sup>1,4</sup>

<sup>1</sup>NCSR Demokritos, Athens, Greece

<sup>2</sup>The American College of Greece, Athens

<sup>3</sup>Technical University of Crete, Greece

<sup>4</sup>University of Piraeus, Greece

{manospits,alevizos.elias,a.artikis}@iit.demokritos.gr ngiatrakos@tuc.gr





#### Introduction

Complex Event Forecasting (CEF) is a process whereby complex events of interest are forecast over a stream of simple events. Figure 1 illustrates the typical architecture of CEF systems.

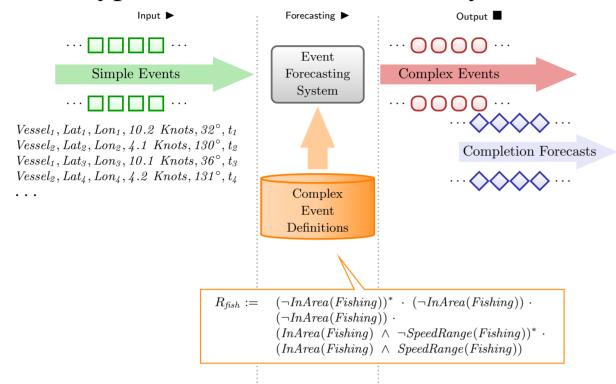


Figure 1: Complex Event Forecasting. We use Wayeb, a CEF engine where:

- CE patterns are defined as regular expressions,
- then they are compiled into automata,
- which are then used for Complex Event Recognition.

Wayeb forecasts CEs by employing prediction suffix trees (PST).

• Automata runs + PSTs  $\rightarrow$  CEF

**Challenge:** Wayeb's forecasting quality depends heavily on PST hyper-parameters. However, static (offline) optimisation fails when data evolves over time (see Figure 2).

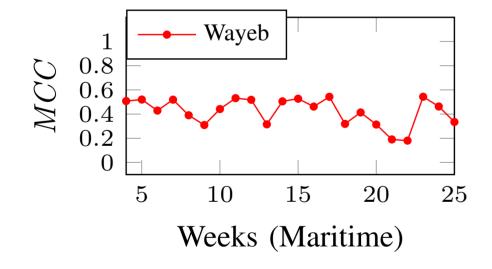


Figure 2: CEF for Maritime Situational Awareness.

To address this, we propose RTCEF: a run-time CEF adaptation framework offering:

- Distributed architecture,
- Re-train vs. re-optimise policies,
- Minimal forecasting disruption.

## **Stationarity Assumption**

Assuming a stationary environment (no data evolution), we introduce offCEF, an offline CEF optimisation framework. The architecture of offCEF is illustrated in Figure 3.

- A *Model Factory* trains and evaluates Wayeb models using candidate hyper-parameters.
- A *Controller* uses **Bayesian optimisation** to search the hyper-parameter space and prescribe train-test runs to the Factory.

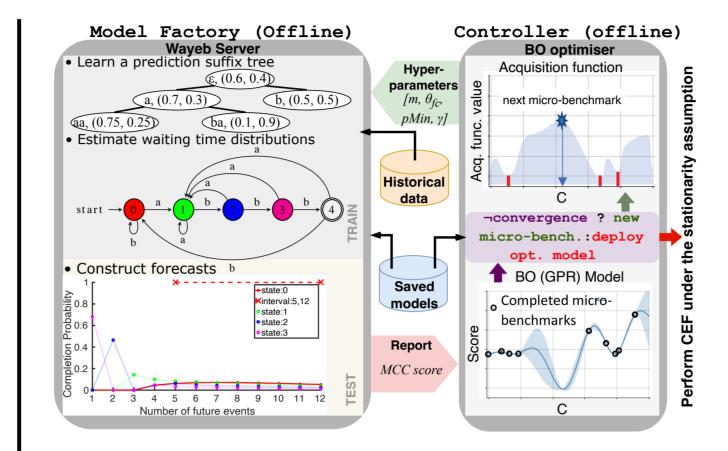


Figure 3: Architecture of offCEF.

# **Run-Time CEF Adaptation**

Continuous data evolution necessitates *run-time adaptation*. Frequent hyper-parameter tuning is computationally expensive, while critical systems demand high availability. RTCEF addresses these challenges through a lightweight and distributed framework (Figure 4).

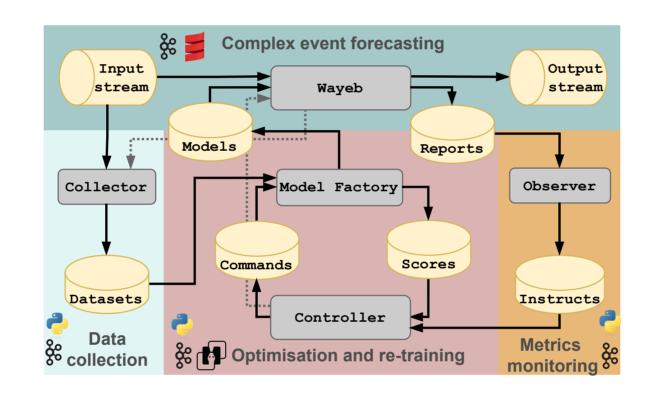
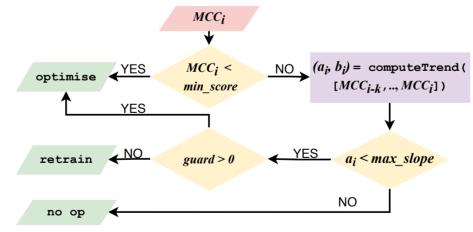


Figure 4: Architecture of RTCEF.

**Metrics monitoring**. The *Observer* service monitors Wayeb's performance, and accordingly issues re-train or optimisation instructions to the *Controller*. The re-train vs re-optimise policy is illustrated in the flow chart of Figure 5, while an example run is illustrated in Figure 6.



**Figure 5:** Re-train vs. Re-optimise decision flow in RTCEF.

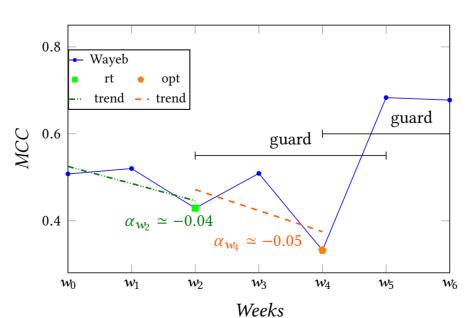


Figure 6: Example run of the Observer.

**Optimisation and re-training** Hyper-parameter optimisation happens in a similar manner to offCEF. However, instead of running each optimisation run from scratch, in RTCEF the controller uses samples from previous runs, thus

enabling faster Bayesian optimisation convergence. Re-training, happens using the last hyperparameters produced.

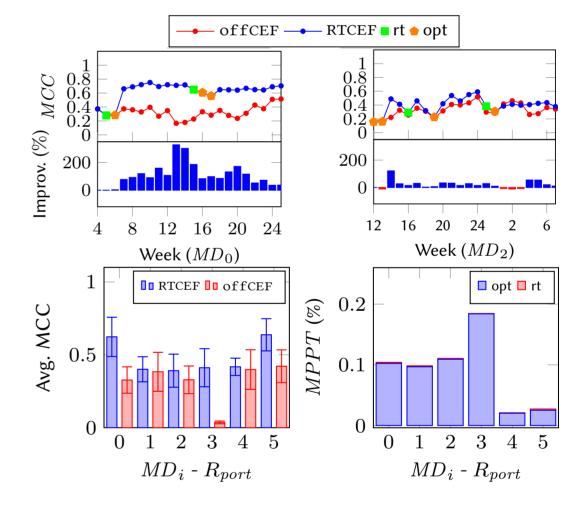
Complex Event Forecasting. Wayeb consumes the input stream and produces an output stream of CE forecasts. Additionally, Wayeb produces a stream of performance reports. Wayeb stops only when a new model produced by hyper-parameter optimisation or re-training is available. RTCEF ensures that model replacement is *lossless* i.e., no automaton run is interrupted and no forecast is lost.

**Data collection**. To ensure that the latest dataset is available for re-training or hyper-parameter optimisation tasks, the *Collector* stores subsets of the input stream in a sliding window manner.

# **Experimental Evaluation**

We evaluate RTCEF in Maritime Situational Awareness and in Credit Card Fraud Detection. We use a 6 month real-world maritime dataset<sup>1</sup> containing 18M AIS vessel position signals and a 21 month synthetic card transaction dataset<sup>2</sup> containing 1M transactions.

#### **Maritime Situational Awareness**



**Figure 7:** Maritime Experiments. MPPT stands for average percentage of time spent for re-training and hyperparameter optimisation every four weeks.

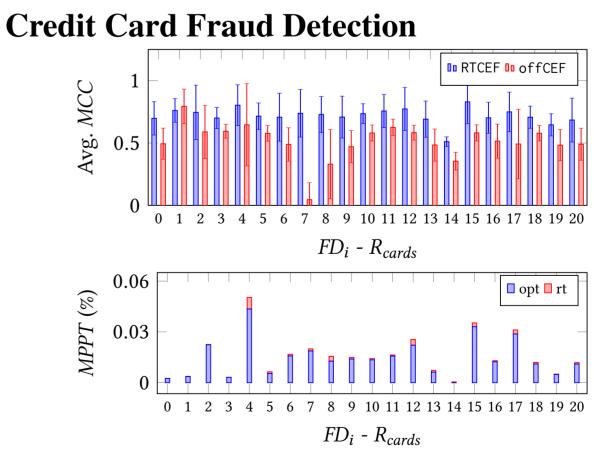


Figure 8: Credit Card Fraud Detection Experiments.

### Summary

We presented RTCEF:

- a novel framework for **run-time adaptation** of CEF,
- involving services running synergistically for **undisrupted** CEF,
- with **clear benefits** over the offline approach.

<sup>1</sup>https://zenodo.org/records/1167595

<sup>2</sup>https://feedzai.com/