Automating the Placement of Time Series Models for IoT Healthcare Applications

Lauren Roberts†, Peter Michalák*, Sarah Heaps†, Michael Trenell‡, Darren Wilkinson†, Paul Watson*
*School of Computing, †School of Mathematics, Statistics and Physics, ‡Institute of Cellular Medicine
Newcastle University
Newcastle upon Tyne, UK
(L.K.Roberts, P.Michalak1, Sarah.Heaps, Michael.Trenell, Darren.Wilkinson, Paul.Watson)@ncl.ac.uk

***Abstract*—There has been a dramatic growth in the number and range of Internet of Things (IoT) sensors that generate healthcare data. These sensors stream high-dimensional time series data that must be analysed in order to provide the insights into medical conditions that can improve patient healthcare. This raises both statistical and computational challenges, including where to deploy the streaming data analytics, given that a typical healthcare IoT system will combine a highly diverse set of components with very varied computational characteristics, e.g. sensors, mobile phones and clouds. Different partitionings of the analytics across these components can dramatically affect key factors such as the battery life of the sensors, and the overall performance. In this work we describe a method for automatically partitioning stream processing across a set of components in order to optimise for a range of factors including sensor battery life and communications bandwidth. We illustrate this using our implementation of a statistical model predicting the glucose levels of type II diabetes patients in order to reduce the risk of hyperglycaemia.**

I. Introduction

Due to recent advances in the Internet of Things, a growing number of sensors are generating large volumes of time series data that can assist in real time decision-making. In healthcare analytics, these data can be analysed to give actionable insights into patients health and well-being. Our work focuses on real-time Bayesian monitoring and forecasting of a bivariate time-series of blood glucose levels and physical activity in patients with type II diabetes.

Continuous Glucose Monitors (CGMs) are small, minimally invasive sensors that give accurate glucose readings. Wearable activity monitors provide high frequency accelerometer readings from which the overall activity of the user can be derived. Time series models can be fitted to the data from both these sensors, and forecasts made based on past patient activity levels, past glucose levels and the relationship between them. Forecasting some time points ahead gives warning of pending hyperglycaemic episodes, and allows users to be issued with behavioural prompts, for example a message on their watch encouraging them to increase their activity level to reduce the glucose levels. Achieving this has required us to address both computational and statistical challenges.

This work was supported by the Engineering and Physical Sciences Research Council, Centre for Doctoral Training in Cloud Computing for Big Data [grant number EP/L015358/1] and Digital Catapult Teaching Grant Award (KH161118).

II. Time Series Modelling and Stream Analytics

The accelerometer data is pre-processed using the Euclidean Norm minus 1 algorithm [1], [2] to give an overall activity level per epoch. The epoch length is chosen to match the frequency of the glucose data (5 minutes) to overcome the challenge of having a mismatch in frequency of the two data sets. Measurements are then classed as low, medium or high activity through a hidden Markov model (HMM) using a skew Normal distribution to describe the density of each activity state. Using this distribution accounts for the long tails in the activity data. Conditional on the activity level, patient blood glucose levels are then modelled using a dynamic linear model (DLM) [3] with Fourier components to capture the seasonality and a time-invariant regression on previous activity levels.

Inference is carried out in a Bayesian framework, performing approximate computation using Markov chain Monte Carlo methods to find the posterior distribution of the HMM and DLM parameters. These computationally intensive operations are executed in the cloud and generate personalised models for individual patients that are used for online forecasting. The placement of the forecasting algorithm raises computational challenges. In particular the activity watch and CGM communicate with a mobile phone that is connected to the cloud. If all the data are sent to the cloud for forecasting then the watch battery is drained by the energy needed to transmit so many messages. On the other hand, the activity watch can only run basic analytics algorithms, and these can also drain the watch battery more quickly than is desirable.

To address these challenges we automatically determine the optimal way to holistically partition the computation over the available components. To do this, the data processing tasks are expressed as queries in a high-level, declarative, event processing language. These queries are parsed and decomposed into a Directed Acyclic Graph that is subject to logical and physical optimisation [4], [5] to find the deployment option that best satisfies key non-functional requirements, including Energy, Bandwidth and Cost. To determine the best plan, cost models, such as those for the energy consumption of IoT devices, are integrated within the system so that alternatives can be quantitatively compared [6]. Following this, the software is then automatically deployed.

978-1-5386-9156-4/18/$31.00 ©2018 IEEE
DOI 10.1109/eScience.2018.00056

REFERENCES

[1] V. T. van Hees, L. Gorzelniak, E. C. D. Leon, M. Eder, M. Pias, S. Taherian, U. Ekelund, F. Renström, P. W. Franks, A. Horsch *et al.*, "Separating movement and gravity components in an acceleration signal and implications for the assessment of human daily physical activity," *PloS one*, vol. 8, no. 4, p. e61691, 2013.

[2] V. T. van Hees, Z. Fang, J. Langford, F. Assah, A. Mohammad, I. C. da Silva, M. I. Trenell, T. White, N. J. Wareham, and S. Brage, "Autocalibration of accelerometer data for free-living physical activity assessment using local gravity and temperature: an evaluation on four continents," *Journal of Applied Physiology*, vol. 117, no. 7, pp. 738–744, 2014.

[3] G. Petris, S. Petrone, and P. Campagnoli, "Dynamic linear models," in *Dynamic Linear Models with R*. Springer, 2009, pp. 31–84.

[4] H. Garcia-Molina, *Database systems: the complete book*. Pearson Education India, 2008.

[5] M. Hirzel, R. Soulé, S. Schneider, B. Gedik, and R. Grimm, "A catalog of stream processing optimizations," *ACM Computing Surveys (CSUR)*, vol. 46, no. 4, p. 46, 2014.

[6] P. Michalák and P. Watson, "PATH2iot: A Holistic, Distributed Stream Processing System," in *Cloud Computing Technology and Science (CloudCom), 2017 IEEE International Conference on*. IEEE, 2017, pp. 25–32.