Process Model Forecasting Using Time Series

Johannes De Smedt[1], Artem Polyvyanyy[2], and Jochen De Weerdt[1]

[1] Department of Decision Sciences and Information Management
Faculty of Economics and Business, KU Leuven, Leuven, Belgium
{johannes.desmedt;jochen.deweerdt}@kuleuven.be
[2] School of Computing and Information Systems
The University of Melbourne, Australia
artem.polyvyanyy@unimelb.edu.au

Abstract. The surge in event-based data that is recorded during the execution of business processes is every-growing and has spurred an array of process analytics techniques to support and improve information systems. A major strand of process analytics encompasses the prediction of the process' future development, mostly focusing on next-step, remaining time, or goal-oriented prediction. The granularity of such approaches lies with the events in the process. This work approaches a process at the abstraction level of the process model, i.e., rather than fine-granular next-step prediction, the process' global behaviour is extrapolated into the future. To this purpose, event data is captured at various intervals and aggregated in the form of directly-follows graphs. Each activity pairs' relation within the graph is monitored over these intervals and their future values predicted using common time series techniques. Experiments show that these techniques are already well-capable of informing process analysts with the future status of the process.

Keywords: Process model forecasting, predictive process modelling, process mining, time series analysis

1 Introduction

2 Preliminaries

An event log L contains the recording of traces $\sigma \in L$ produced by an information system during its execution and contains a sequence of events. Events in these traces are part of the powerset over the alphabet of activities Σ which exist in the information system $\langle e_1, ..., e_{|\sigma|} \rangle \subseteq \Sigma^*$. Directly follows relations between activities in an event log can be expressed as a counting function over activity pairs $>_L$: $\Sigma \times \Sigma \rightarrow \mathbb{N}$ with $>_L (a_1, a_2) = |\{e_n = a_1, e_{n+1} = a_2, \forall e_i \in L\}|$. Directly follows (DF) relations can be calculated on traces and subtraces in a similar fashion. A Directly Follows Graph (DFG) of the process then exists as the weighted directed graph with the activities as nodes and their DF relations as weights $DFG = (\Sigma, >_L)$.

In order to obtain predictions regarding the evolution of the DFG we construct DFGs for subsets of the log. Many aggregations and bucketing techniques exist for next-step and goal-oriented outcome prediction [5, 6], e.g., predictions at a point in the process rely on prefixes of a certain length, or particular state aggregations [1]. In the proposed forecasting approach, however, not cross-sectional but time series data will be used. Hence, the evolution of the DFGs will be monitored over intervals of the log where multiple aggregations are possible:

- Equitemporal aggregation: each sublog contains a part of the event log of equal time duration. This can lead to sparesely populated sublogs when the events' occurrences are not uniformly spread over time, however, is easy to apply (on new traces).
- Equisized aggregation: each sublog cotnains a part of the event log of similar DF sum. This leads to well-populated sublogs, however, might be harder to apply when new data does not contain sufficient new DF occurrences.

Time series can be obtained for all $>_{Ls}$, $Ls \subseteq L$ by applying the aforementioned aggregations. Tables 1 and 2 provide an example of both.

Case ID	Activity	Timestamp
1	A1	11:30
1	A2	11:45
1	A1	12:10
1	A2	12:15
2	A1	11:40
2	A1	11:55
3	A1	12:20
3	A2	12:40
3	A2	12:45

Table 1: Example event log with 3 traces and 2 activities.

DF	Equitemporal	Equisized
$<_{Ls}$ $(A1, A1)$	(0,1,0)	(1,0,0)
$<_{Ls}$ $(A1, A2)$	(1,1,1)	(1,1,1)
$<_{Ls}$ $(A2, A1)$	(0,1,0)	(0,1,0)
$<_{Ls}$ $(A2, A2)$	(0,0,1)	(0,0,1)

Table 2: An example of using an interval of 3 used for equitemporal aggregation (75 minutes in 3 intervals of 25 minutes) and equisized intervals of size 2 (6 DFs over 3 intervals)).

3 Methodology

This section outlines the forecasting techniques that will be used, as well as their connection with time series extracted from event logs.

3.1 Time series techniques

To model the time series of DFGs, various algoritms are used. In time series modelling, the main objective is to obtain a forecast or prediction $\hat{y}_{T+h|T}$ for

a horizon $h \in \mathbb{N}$ based on previous T values in the series $(y_1, ..., y_T)$ [4]. A wide array of time series techniques exist, ranging from simple models such as naive forecasts over to more advanced approaches such as exponential smoothing and autoregressive models. Many also exist in a seasonal variant due to their application in contexts such as sales forecasting. Below, the most-widely used techniques are formalised.

The naive forecast simply uses the last value of the time series T as its prediction:

$$\hat{y}_{T+h|T} = y_T$$

A Simple Exponential Smoothing (SES) model uses a weighted average of past values where their important exponentially decays as they are further into the past:

$$\hat{y}_{T+h|T} = \alpha y_T + (1-\alpha)\hat{y}_{T|T-1}$$

with $\alpha \in [0, 1]$ a smoothing parameter where lower values of α allow for focusing on the more recent values more. Holt's models introduce a trend in the forecast, meaning the forecast is not flat:

$$\hat{y}_{T+h|T} = l_t + hb_t$$

with $l_t = \alpha y_t + (1-\alpha)(l_{t-1} + b_{t-1})$ modelling the overall level of the time series and $b_t = \beta(l_t - l_{t-1}) + (1-\beta)b_{t-1}$ modelling the trend over the series where α is as before and β the smoothing parameter for the trend. Exponential smoothing models often perform very well despite their simple setup.

AutoRegressive Integrating Moving Average (ARIMA) models are based on autocorrelations within time series. The combine autoregressions with a moving average over error terms.

An AutoRegressive (AR) model of order p uses the past p values in the time series and apply a regression over them. It can be written as:

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \epsilon_t$$

where all ϕ_i, $i \in [1, p]$ can assign different weights to different time lags.

A Moving Average (MA) model of order q can be written as:

$$y_t = c + \epsilon_t + \phi_1 \epsilon_{t-1} + \cdots + \phi_q \epsilon_{t-q}$$

with $\phi > 0$ a smoothing parameter, and ϵ_t again a white noise series, hence the model creates a moving average of the past forecast errors. Given the necessity of using a white noise series for AR and MA models, data is often differenced to obtain such series.

ARIMA models then combine both AR and MA models where the integration is taking place after modelling as these models are fitted over differenced time series. An ARIMA(p, d, q) model can be written as:

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 \epsilon_{t-1} + \cdots + \theta_q \epsilon_t$$

with y'_t a differenced series of order d. Forecasting is possible by introducing $T+h$ to the equation and iteratively obtaining results by starting off with $T+1$. ARIMA models are considered to be one of the strongest time series modelling techniques.

An extension to ARIMA which is widely used in econometrics exists in (Generalized) AutoRegressive Conditional Heteroskedasticity ((G)ARCH) models [2]. They resolve the assumption that the variance of the error term has to be equal over time, but rather model this variance as a function of the previous error term. For AR-models, this leads to the use of ARCH-models, while for ARMA models GARCH-models are used as follows.

ARCH model:

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \cdots + \alpha_q \epsilon_{t-q}^2$$

with $\alpha_0 > 0$ and $\alpha_i \geq 0$, $i > 0$ smoothing parameters, and $\epsilon_t = \sigma_t z_t$ where z_t is a white noise process with $\sigma_t = \sqrt{\alpha_0 + \alpha_1 y_{t-1}^2}$.

GARCH model:

$$\sigma_t^2 = \omega + \alpha_1 \epsilon_{t-1}^2 + \cdots + \alpha_q \epsilon_{t-q}^2 + \cdots + \beta_1 \sigma_{t-1}^2 + \cdots + \beta_p \sigma_{t-p}^2$$

with $\alpha_0 > 0$ and $\alpha_i \geq 0$, $i > 0$.

3.2 Connection with DF series

The aforementioned time series are all univariate in contrast with vector autoregression models, or machine learning-based methods such as neural networks or random forest regressors. To this purpose, a direct strategy [7] is used where

4 Experimental evaluation

4.1 Re-sampling and test setup

Sample size requirements [3]

Depending on the number of parameters that are required for an algorithm, depending on whether any seasonality is involved.

4.2 Results for equisize experiments

References

1. van der Aalst, W.M.P., Rubin, V.A., Verbeek, H.M.W., van Dongen, B.F., Kindler, E., Günther, C.W.: Process mining: a two-step approach to balance between underfitting and overfitting. Softw. Syst. Model. 9(1), 87–111 (2010)
2. Francq, C., Zakoian, J.M.: GARCH models: structure, statistical inference and financial applications. John Wiley & Sons (2019)
3. Hanke, J.E., Reitsch, A.G., Wichern, D.W.: Business forecasting, vol. 9. Prentice Hall New Jersey (2001)

4. Hyndman, R.J., Athanasopoulos, G.: Forecasting: principles and practice. OTexts (2018)
5. Tax, N., Verenich, I., Rosa, M.L., Dumas, M.: Predictive business process monitoring with LSTM neural networks. In: CAiSE. Lecture Notes in Computer Science, vol. 10253, pp. 477–492. Springer (2017)
6. Teinemaa, I., Dumas, M., Rosa, M.L., Maggi, F.M.: Outcome-oriented predictive process monitoring: Review and benchmark. ACM Trans. Knowl. Discov. Data 13(2), 17:1–17:57 (2019)
7. Weigend, A.S.: Time series prediction: forecasting the future and understanding the past. Routledge (2018)