Incorporating spatial context into remaining-time predictive process monitoring

Ogunbiyi, Niyi, Basukoski, Artie and Chaussalet, Thierry

An author version of a paper presented at the 36th Annual ACM Symposium on Applied Computing. Virtual Event, Republic of Korea, 22 - 26 Mar 2021, ACM.

https://doi.org/10.1145/3412841.3441933

Incorporating Spatial Context into Remaining-Time Predictive Process Monitoring

ABSTRACT

Predictive process monitoring aims to accurately predict a variable of interest (e.g. remaining time) or the future state of the process instance (e.g. outcome or next step). The quest for models with higher predictive power has led to the development of a variety of novel approaches. However, though the location of events is a crucial explanatory variable in many processes, as yet there have been no studies which have incorporated spatial context into the predictive process monitoring framework. This paper seeks to address this problem by introducing the concept of a spatial event log which captures trace and event location details.

The predictive utility of spatial contextual features is evaluated vis-à-vis other contextual features. An approach is proposed to predict the remaining time of an in-flight process instance by calculating the buffer distances between the location of events in a spatial event log to capture spatial proximity and connectedness. These distances are subsequently used to construct a regression model which is used to predict the remaining time for events in the test data. The proposed approach is benchmarked against existing approaches using five real-life event logs and demonstrates that spatial features improve the predictive power of process monitoring models.

CCS CONCEPTS

• **Information systems → Information systems application→** Spatial-temporal systems → Geographic information systems; Robotics • **Applied Computing** → Operations research → Forecasting

KEYWORDS

Operational business process management, Process monitoring, Remaining time predictive modelling, Spatial context, Distributed processes.

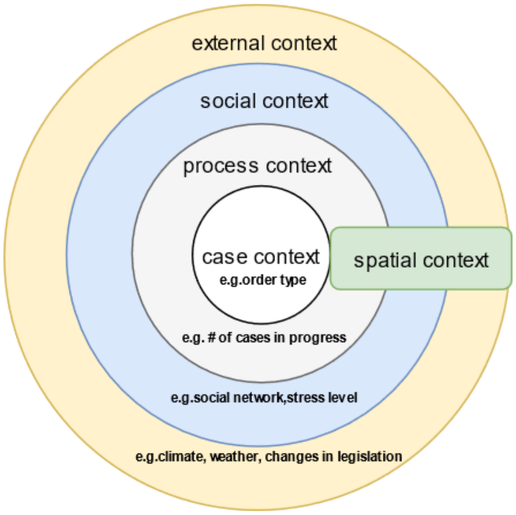
1 INTRODUCTION

Effectively predicting process outcomes in operational business management is essential for Customer Relationship Management (e.g. 'will this customer's order be completed on time?'), Enterprise Resource Planning (e.g. 'what level of resourcing will be required to manage running cases/process instances?') and Operational Process Improvement (e.g. 'what are the common attributes of cases that consistently complete late?'), among others. Predicting the remaining time of a process instance is also very useful. It is essential for effective scheduling of sequentially dependent processes and is a crucial determinant of consumer choice (e.g. where two or more services are identical in price and quality).

Reference [0] highlights the importance of contextual factors in predictive process monitoring and identifies four pertinent contextual types:

- **Case context** - the properties or attributes of a case.
- **Process context** – similar cases that may be competing for the same resources.
- **Social context** - the way human resources collaborate in an organisation to work on the process of interest.
- **External context** – factors in the broader ecosystem that impacts the process. e.g. weather, legislation, location, etc.

That study makes the point that "although … external context can have a dramatic impact on the process being analysed; it is difficult to select the relevant variables." This study aims to address the problem of incorporating spatial context into the process mining workflow by introducing the idea of a spatial event log which includes the locations of process traces and events



external context
social context
process context
case context
e.g.order type
spatial context
e.g. # of cases in progress
e.g.social network,stress level
e.g.climate, weather, changes in legislation

Figure 1: Contextual Factors and Relationship – (Adapted from [0])

Even though every event occurs at a location, event logs do not typically capture spatial data. As shown by figure 1, this contextual type overlaps with the other types. For example, relevant process legislation (external context) and the manner process performers interact (social context) are both a function of location. Incorporating the spatial context enables process analysts to determine whether processes outcome exhibit spatial patterns. This is a question of interest particularly with distributed processes and one that has increased in salience with the COVID-19 pandemic which has necessitated the distribution of process execution, for example, due to the requirement for process performers to work from home. If it can be established that process outcomes display spatial pattern(s), location becomes a key explanatory variable. The first law of geography (Tobler's Law) states that "all objects are related, but nearer objects are more related than further objects"[0]. The concept of spatial autocorrelation, which attempts to "measure…simultaneously…the similarities in the location of spatial objects and their attributes", explains this relationship [0]. Besides, incorporating the spatial dimension into event logs facilitates the discovery of the trajectory of process artefacts which could help detect motion waste.

Furthermore, it would be possible to construct a de jure process model for different locations (e.g. because of legislative requirements) and check whether discovered processes (stratified by location) conforms. However, for this paper, the focus will be on utilising the spatial context to improve the prediction of the remaining time. In addition to a contribution to the knowledge base by proposing a novel way to incorporate the spatial context into the predictive process mining workflow, we demonstrate by empirical evaluation, the importance of these contextual features. We show that our proposed approach performs comparably with start-of-the-art predictive process monitoring techniques.

The remainder of the paper is structured as follows: Section 2 details preceding studies which have provided the motivation and methodological basis for this study. Section 3 defines vital terms built on throughout the paper. Section 4 describes the proposed approach, while Section 5 details the evaluation results of the proposed approach. The penultimate section describes the threats to the validity of the study while the final section summarises the findings and proposes further research areas for extending these.

2 RELATED WORKS

A review of the literature reveals three primary predictive process monitoring approaches: Model-based approaches **Error! Reference source not found.Error! Reference source not found.**, sequence-to-feature encoding (STEP) approaches **Error! Reference source not found. Error! Reference source not found.**[0] and simulation-based approaches **Error! Reference source not found.Error! Reference source not found.**.

STEP approaches encode event log into feature-outcome pairs using a variety of approaches such as last state, aggregation, index-based or tensor encoding**Error! Reference source not found.**[**Error! Reference source not found.**]. However, it is worth mentioning a subset of STEP approaches that have become popular in recent years .i.e. neural-network-based approaches 0000. These state-of-the-art models make it relatively easy to include additional features into the prediction model. While the majority of these approaches focus on the next activity as the prediction target, the approach proposed by 0 utilises an LSTM (a particular type of a Recurrent Neural Network) to iteratively predict the remaining activities till case completion and associated timestamps. This enables estimation of the remaining time of the process instance.

With regards to spatial analysis, as mentioned earlier [0] proposed the law which laid the foundation for spatial dependence and autocorrelation. Numerous studies have built on this foundation, and it is commonly accepted as a "reasonable regularity that generally holds true". Reference [0] argues that rather than merely being a confounding variable, spatial autocorrelation "is information-bearing since it reveals the spatial association among geographic entities”.

Reference [0] proposes an approach for spatial prediction that utilises buffer distances from observation points as features to build a spatial machine learning model. Their approach offers advantages over traditional geostatistical approaches (e.g. kriging) because it makes "no rigid statistical assumptions about the distribution and stationarity of the target variable, it is more flexible towards incorporating, combining and extending covariates of different types, and it possibly yields more informative maps characterising the prediction error."

In this paper, we utilise the STEP approach combined with the approach proposed by [0] to build a spatial predictive process monitoring framework.

3 BACKGROUND

3.1 Definitions

1. Event, Traces and Event Logs

Several key terms to be built on throughout this review are formally defined. We adopt the standard attribute notation defined in [0].

Definition 3.1 (Event). Let ε represent the event universe and T the time domain, A represent the set of activities and P represent the set of performers (i.e. individuals and teams).

An event e is a tuple (#case_identifier(e), #activity(e), #start_time(e),#completion_time(e),#attribute$_1$(e)..#attribute$_n$(e)). The elements of the tuple represent the attributes associated with the event. Though an event is minimally defined by the triplet ((#case_identifier(e), #activity(e), # completion_time(e)), it is common and desirable to have additional attributes such as indicating the performer associated with the event and #trans(e) indicating the transaction type associated with the event, amongst others. For each of these attributes, there is a function which assigns the attribute to the event .e.g. attr$_{start\_time}$ $\in \varepsilon \rightarrow T$ assigning a start time to the event, attr$_{completion\_time}$ $\in \varepsilon \rightarrow T$ assigning a completion time to the event, attr$_{activity}$ $\in \varepsilon \rightarrow A$ assigning an activity label to the event and attr$_{performer}$ $\in \varepsilon \nrightarrow P$, a partial function assigning a performer (or resource) to events. Note that attr$_{performer}$ is a partial function as some events may not be associated with any performers.

An event is often identified by the activity label (#activity(e)) which describes the work performed on a process instance (or case) that transforms input(s) to output(s).

Definition 3.2 (Terminal activities) Let $Z \subseteq A$ represent the set of valid terminal activity labels.
e_n is a valid terminal event if #activity_label(e_n) $\in Z$. This event indicates a 'clean' completion of the process instance. Otherwise, the process instance is still in-flight or abandoned.

Definition 3.3 (Event log) An event log is set of traces (full and partial) $L \subseteq C$ for a particular process such that each event appears at least once in the log .i.e for any $\sigma_1, \sigma_2 \in L$: $\forall e_1 \in \sigma_1 \forall e_2 \in \sigma_2\ e_{1 \neq} e_2$ *or* $\sigma_1 = \sigma_2$

Definition 3.4 (Remaining time) Let σ^f represent a full trace, $\tau.e_n$ represent the completion time associated with the terminal event, #completion_time(e_n), and t represents the prediction point. For $t < \tau.e_n$,the remaining time τ_{rem} = t - $\tau.e_n$. It indicates the remaining time to completion of case/process instance. Note that predicting at or after the completion time (i.e. $t \geq \tau.e_n$) is pointless.

Definition 3.5 (Elapsed time) Let σ^f represent a full trace, $\tau.e_1$ represent the start time associated with the start event, #start_time(e_1), and t represents the prediction point. For $t > \tau.e_1$, the elapsed time τ_{ela} = t - $\tau.e_1$. It indicates the elapsed time from the start of case/process instance to the prediction time.

Definition 3.6 (Cycle time) Let σ^f represent a full trace, $\tau.e_1$ represent the start time associated with the start event, #start_time(e_1) and $\tau.e_n$ represent the completion time associated with the terminal event, #completion_time(e_n), The trace cycle time τ_{cyc} $= \tau.e_n - \tau.e_1$. It indicates the time taken to complete the process instance from start to finish

2. **Spatial Objects and Event Logs.**

Definition 3.7 (Point) Let R^2 represent a two-dimensional Euclidean space. A point is a zero-dimensional geographical object used to indicate a spatial occurrence in R^2.
A point's coordinates can be specified as longitude, and latitude or Northing N and Easting E offsets relative to a specified origin, depending on the defined Coordinate Reference System (CRS - see Def 3.10-iii)

Definition 3.8 (Spatial Point Process) Let $X \subseteq \mathbb{R}^2$ for some distance d. A spatial point process is a stochastic model for a random scattering of points on X for d which describe the occurrence over time of points {#location$_{(x,y)}$(e_1), #location$_{(x,y)}$(e_2)….#location$_{(x,y)}$(e_n)}over time {#completion_time(e_1), #completion_time(e_2)… #completion_time(e_n)}

Definition 3.9 (Buffer Distances) Let #location$_{(x,y)}$(e_z) represent the location attribute for event Z. D_z =(d(#location$_{(x,y)}$(e_1), d(#location$_{(x,y)}$(e_2)…. d(#location$_{(x,y)}$(e_n)) represents the buffer distance between #location$_{(x,y)}$(e_z) and the other events. It captures the spatial relationship between the location of events in the log.

Definition 3.10 (Spatial event log) An event log where all events are associated with a location attribute (#location$_{(x,y)}$(e)). For example, we could define a function attr$_{location(x,y)}$ $\in \varepsilon \rightarrow P$, to assign a location to each performer (or resource) who execute events. However, it could represent some other location that is meaningful to the process; e.g., for a process to report and track the resolution of a defect, the location could represent the location of the reported defect. We recommend providing the following attributes at the event log metadata level:

i. Location scope attribute (#location_scope(L)) to indicate whether the scope of the location attribute is trace- or event-wide.
ii. Location function (#location_function(L)) to describe the nature of the location attribute in the log.
iii. Coordinate Reference System (#CRS(L)) to indicate the Coordinate Reference System for the event location attribute

To illustrate the terms above, consider a process for reporting and remediating defects to public goods, e.g. potholes, street light outages. An event in this process would be any from the valid set: {'Create Service Request', 'Initial Review', 'Assign Service Request', 'Assign Crew', 'Contact Citizen', 'Put Service Request On Hold', 'Close Service Request'}. Each event will be associated with a start and end time and the resource who performed the activity, amongst others. An example of a full trace for a process instance would be {'Create Service Request', 'Review', 'Assign Service Request', 'Assign Crew', 'Contact Citizen', 'Close Service Request'}. Note that 'Create Service Request' and 'Close Service Request' are the start and terminal events, respectively. An example of a partial trace for a process instance would be {'Create Service Request', 'Initial View', 'Assign Service Request'}. Note the absence of a valid terminal event indicating that the process is in-flight. This event log could be transformed into a spatial event log by, for example, associating the location of the appropriate performer with each event (see Table 1).

4 APPROACH

4.1 Overview

Figure 2 provides an overview of the proposed approach used in the evaluation of our proposed approach (see section 5). The initial step is the creation of a spatial event log which associates the events in the log with spatial context. Subsequently, we create measures of spatial proximity by calculating buffer distances for each point in the training data set to all the other points. These distances are used to build a spatial regression model. We improve runtime performance by performing these steps offline.

After that, in the online phase, the remaining time for test data are predicted using the regression models based on the location of the event/trace

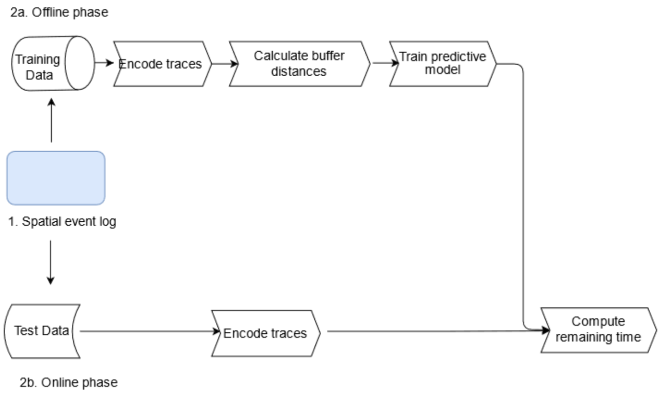


Figure 2: Overview of the proposed approach

4.2 Spatial Event Log

An event log can be transformed into a spatial event log by associating the coordinates of a meaningful location to each event in the log. A location is considered meaningful if it facilitates the discovery of spatial patterns in the event log. A typical choice is the location for the performer associated with each event.

Another example is the location of reported defects (coded as longitude and latitude) in a service management application. This approach is considered meaningful for these processes as the location of defects is expected to demonstrate evidence of a spatial point process. For example, for road defects, the spatial process will likely depend on the weather, maintenance schedule, organisational process, regulation, among others.

Table 2: Event Log Overview

Service Request ID	Service Category	Longitude	Latitude	Activity	Start Time	End Time
XY4567	Roads	51.3161	0.06047	Create Service Request	22/10/2017 18:34	22/10/2017 18:38
XY4567	Roads	51.2425	0.06132	Accept Ownership	25/10/2017 10:16	25/10/2017 10:17
XY4567	Roads	51.2557	0.06156	Assign Crew	25/10/2017 16:01	25/10/2017 16:22
XY4567	Roads	51.2557	0.06132	Contact Citizen	27/10/2017 11:04	27/10/2017 11:09
XY4567	Roads	51.2557	0.06114	Close Service Request	27/10/2017 11:45	27/10/2017 11:55

4.3 Predictive Modelling

The approach consists of two phases: offline (training) and online (testing). In the training phase, the traces in the event log are encoded. For the event-level logs, we used indexed based encoding to encode the traces while we utilised a combination of aggregation and last state encoding for the trace level logs.

We subsequently utilised the approach proposed by [0] to build a Random Forest spatial predictive monitoring model as described below.

We transformed the event location from the longitude-latitude CRS to the Universal Transverse Mercator (UTM) CRS. We subsequently converted the event log into the spatial data frame to efficiently handle the spatial data. We then calculated the Euclidean buffer distances for each event location in the spatial training set to capture the spatial relationship between the locations. We then spatially overlaid the dependent variable over the spatial window (.i.e. the area where the process was executed). The output of the overlay function and buffer distances are used as the input to build the spatial predictive monitoring model.

In the testing phase, the in-flight traces are encoded utilising the same approach as in the training phase. The spatial model built in the training phase was used to estimate remaining time directly for the event level logs. However, for trace-level logs, the total cycle time for the trace was estimated and the remaining time for the trace is computed by subtracting the elapsed time from the estimated cycle time

Figure 3 details the spatial predictive modelling algorithm

```
Input: An event log L over some trace universe σ with a location scope attribute #location_scope(L), an associated target measure remaining time $\tau_{rem}$, time $\tau_{ela}$, cycle time $\tau_{cyc}$, a spatial window B,a spatial overlay method O and a spatial regression method (REGR) method
Output: A spatial predictive model (S-PM) model for L
Method: Perform the following steps:
    i.    Associate a point spatial object #location$_{(x,y)}$(e) with each trace σ ϵ L (see definition 3.7)
    ii.   Encode each trace using a suitable encoding function
    iii.  For each #location$_{(x,y)}$($e_i$), calculate $D_i$ =(d(#location$_{(x,y)}$($e_1$), d(#location$_{(x,y)}$($e_2$)…. d(#location$_{(x,y)}$($e_n$))
  If attribute #location_scope(L) = 'event'
    iv.   Overlay $\tau_{rem}$ over B using method O to return b
    v.    Induce a regression model s-pm out of L using method REGR using {#location$_{(x,y)}$($e_i$),{ $D_i$ … $D_n$}, b} as input value and $\tau_{rem}$(σ) as target value
    vi.   Estimate the remaining time for each trace $\tau_{i \cdot rem\_pred}$ : s-pm($\sigma_i$)
  If attribute #location_scope(L) = 'trace',
    vii.  overlay $\tau_{cyc}$ over B using method O to return b
    viii. Induce a regression model pst-pm out of L using method using {#location$_{(x,y)}$($e_i$),{ $D_i$ … $D_n$}, b} as input value and $\tau_{cyc}$ (σ) as target value
    ix.   Estimate the cycle time for each trace $\tau_{i \cdot cyc\_pred}$ : s-pm($\sigma_i$)
    x.    For each $\sigma_i$ do
    xi.   Estimate the remaining time for each trace $\tau_{i \cdot rem\_pred}$ :
```

Figure 3: *S-PM* algorithm

5 EVALUATION

In this section, we detail our approach to evaluate the importance of spatial features in the predictive process mining workflow. We evaluated the proposed spatial predictive monitoring techniques against similar predictive monitoring techniques which are based on other features. Specifically, we sought to address the following research questions:

RQ1. Do spatial features contribute to the predictive power of remaining-time predictive approaches vis-à-vis other features?

RQ2. How does spatial-based remaining-time predictive process monitoring approaches compare with existing approaches?

In the following section, we provide further details about the experimental setup and how we answer the research questions.

5.1 Datasets

We used five real-life events for our experiments (see Table 2). For four logs we enriched the event log with synthetic spatial data as follows: Traffic Fines [0], BPI Challenge 2017 [0], BPI Challenge 2019 [0], BPI Challenge 2020[0]. We simulated the synthetic data to reflect as faithfully as possible the spatial patterns we expect to be present in the process. For example, all the event locations were simulated within the territory of the country where the event log was generated. Besides for each event, we approximated the expected distribution. To illustrate, for the traffic fine event log, the expectation is that traffic fines are predominantly issued in urban areas; hence we simulated spatially clustered locations for these events. For these logs, the location for each event is the simulated location of the performer executing each event. We subsequently refer to these logs as the event-level logs.

The fifth event log included real-life spatial data. This log is from a cloud-based request management platform currently used by public service providers (i.e. municipalities and regions) in Canada and the US. Citizens or service provider staff can raise service requests (i.e. requests for information or work to be carried out, application for permits, etc.) via an app on hand-held devices or through a web interface. Functionality exists for the public service provider (typically a municipal agency) to manage these requests through to completion as well as a suite of supporting functionality, e.g. analytics, work management, etc. The scope of the locations in this log are at a trace level .i.e. every event has the same location and the coordinates indicate the location of the reported defects; hence we hereafter refer to this as the trace-level log. We filter the log to extract defects related to road-related defects. However, we are unable to make the data available as doing so will create privacy concerns due to the location coordinates representing observed locations of real people. We considered robust anonymisation of the data; however, we concluded that doing so without loss of accuracy was not achievable

We added additional features such as elapsed time, remaining time, the number of requests raised on the same day as the service request (a measure of workload) and a couple of temporal features to each log.

Table 2: Event Log Overview

	Traffic Fines	BPIC 17	BPIC 19	BPIC 20	Road Defects
# of events	149354	55358	140056	56437	9392
# of cases	26633	3084	306	10500	1324
# of traces	215	1126	305	99	413
# of distinct activities	11	25	34	17	29
Mean trace length	5.61	17.95	457.7	5.37	7.09
Mean throughput time (days)	528.96	21.87	156.78	11.53	82.3
Throughput time SD (days)	346.62	12.94	529.98	17.02	244.78
Location Scope	Event	Event	Event	Event	Trace

5.2 Experimental Setup

For the evaluation, we implemented an approach named *spatial* in R for the spatial approach described in section 4.3, respectively. This approach enables assessment of the importance of the spatial features by building a predictive model from these features and evaluating them vis-à-vis predictive models based on non-spatial features. We evaluated the spatial approaches against a couple of approaches which used a zero prefix-bucketing combined with a gradient boosting machine (*gbm*) and multilayer perceptron (*mlp*) neural network regressors respectively to predict the remaining time for each trace **Error! Reference source not found.**. Both of these models were built using non-spatial features in the event log. We blend each of these approaches with the spatial model using the arithmetic mean of the predictions to create a couple of ensemble models for evaluation purpose. To ensure completeness, we also create a blended ensemble of the non-spatial models. The code and data for the experiments are located in the following Github repository(see https://github.com/etioro/SpatialProcessMonitoring)

For the event-level logs, we used indexed based encoding to encode the traces as it is "lossless and has been shown to achieve relatively high accuracy and reliability" [0]. However, for the trace-level log, we utilise a combination of the aggregation and last state encoding technique **Error! Reference source not found.** where the aggregation function computes the trace length, throughput time and set of activity labels for each trace

We split each event log into test and training sets. We further subdivided the training set, using only the spatial features for 200 data points to build the spatial model and the non-spatial features for the remaining data points to construct the non-spatial models. We subsequently used the test set for making remaining-time predictions which are then evaluated.

As with the methodology used in [**Error! Reference source not found.**], the training & test set were not temporally disjoint.

We chose to utilise the Mean Absolute Error (MAE) to evaluate the accuracy as other measures such as the Root Mean Square Error (RSME) are susceptible to outliers and Mean Percentage Error (MAPE) would be skewed towards the end of a case where remaining time tends towards zero [0].

To achieve the best performance from both the spatial and non-spatial models, we tuned the relevant model hyperparameters. For the spatial-based model, we utilise the approach proposed in [0], while for the non-spatial methods, we use the tuning capabilities inbuilt into the caret package.

5.3 Results

Table 3 details the global MAE and Standard Deviation (SD) for each dataset/algorithm pair. The performance of the algorithms visualised in figure 4, which displays the average ranking of each algorithm over the datasets with associated error bars. Over the five datasets, the ensemble model *gbm+spat* performed best. In general, blending the spatial model with a non-spatial model improved the performance of the non-spatial model. This is explained by the fact that the spatial features explained as much as 30% of the dependent variable (i.e. remaining time) in the spatial models. It is also worth mentioning that the *spatial* model outperformed the ensemble non-spatial models (.i.e. *gbm+mlp*). This confirms the valuable contributions of the spatial features

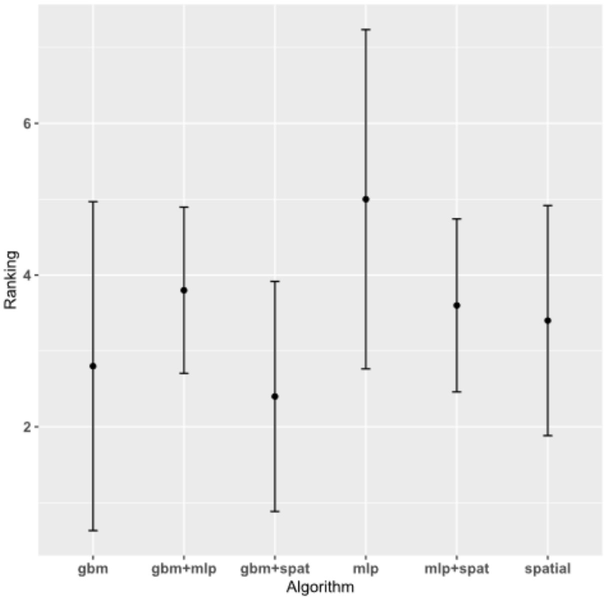


Figure 4: Average Algorithm Ranking with associated error bars.

Figures 5 show the aggregated error values obtained by dividing the Global MAE and SD by the average throughput time for each event log. Normalising these values enables them to be directly comparable (see **Error! Reference source not found.**). *gbm+spat* has the lowest normalised median and mean MAE (0.43 and 0.62 respectively)

To determine which algorithms, differ from the others, we utilise the Quade post-hoc test to perform a pair-wise comparison between the various algorithms. Table 4 shows the results of the pair-wise comparisons (with the value(s) statistically significant at the 95% confidence level in bold font). For most of the pairs, there is insufficient evidence to reject the null hypothesis that they are significantly different. However, the results indicate that the *gbm+spat* method significantly outperforming the existing method(s) (see results in bold).

Table 3: Global MAE ± SD

	spatial	**mlp**	**gbm**	**gbm+mlp**	**gbm+spat**	**mlp+spat**
Traffic Fines	183.09 ± 180.22	276.86 ± 173.20	255.96 ± 206.70	259.97 ± 115.24	216.52 ± 156.45	224.38 ± 151.92
BPIC 17	11.68 ± 10.71	14.62 ± 8.93	8.79 ± 9.51	11.44 ± 8.05	9.86 ± 9.72	12.62 ± 9.07
BPIC 19	81.47 ± 62.45	156.13 ± 86.91	69.29 ± 57.87	100.39 ± 62.67	60.92 ± 40.08	98.64 ± 69.37
BPIC 20	6.12 ± 22.95	6.55 ± 22.06	4.61 ± 21.94	5.38 ± 21.94	4.98 ± 21.94	6.07 ± 22.04
Road Defects	114.64 ± 214.17	109.06 ± 224.99	126.25 ± 208.65	113.36 ± 208.77	115.81 ± 203.14	111.21 ± 219.17

Table 4: Quade post-hoc test of approach rankings across all datasets

	spatial	mlp	mlp+spat	gbm	gbm+spat
mlp	0.183				
mlp+spat	0.865	0.241			
gbm	0.61	0.072	0.498		
gbm+spat	0.399	**0.036**	0.313	0.734	
gbm+mlp	0.734	0.313	0.865	0.399	0.241

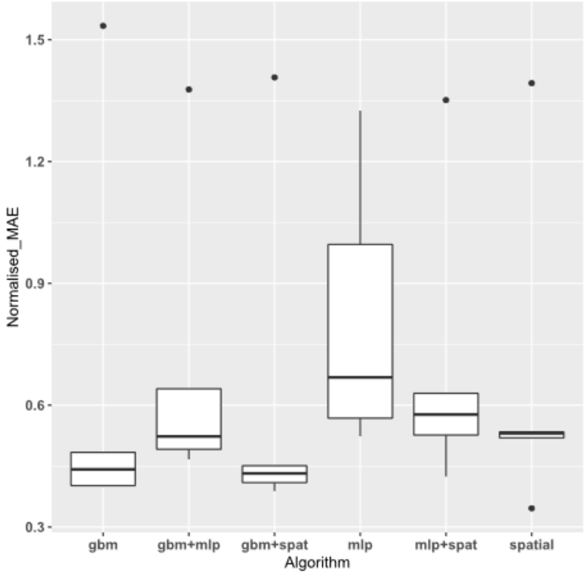


Figure 5(a): Average Normalised MAE

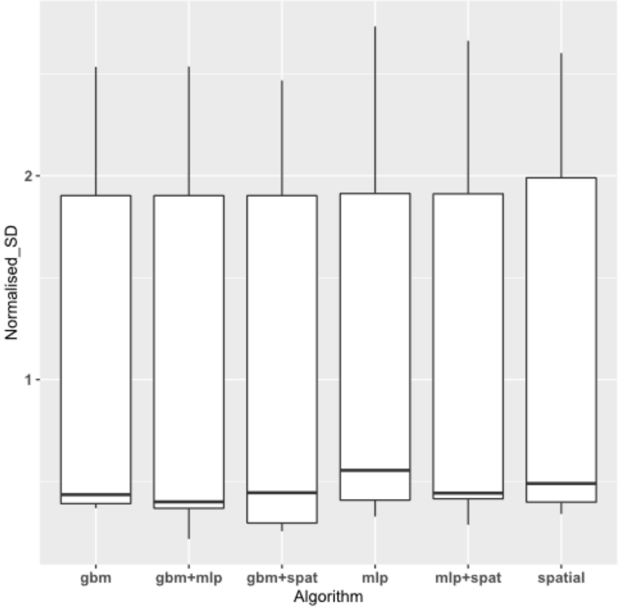


Figure 5(b): Average Normalised Standard Deviation

6 THREATS TO VALIDITY

The main threat to validity was the absence of real-life spatial data at the desired level of granularity. For the four event-level logs for which spatial data was simulated, even though care was taken to reflect the spatial distribution of the process in the simulated data, the spatial effect is likely under-estimated vis-à-vis real-life spatial data.

For the real-life spatial data, the available spatial data was at trace level. In other words, a single location (i.e. service request location) was associated with each completed trace. However, in reality, the location for events are typically dispersed, i.e. e_1 may occur at location A, e_2 at location B, etc. For example, a citizen may raise the service request at location A, reviewed by supervisor based in the field location (at location B) and assigned to a work crew based at location C. Lower granularity of location at event level is expected to produce better results as this captures more of the spatial variation present in the data

Another threat to validity is related to the real-life spatial data is geo-referencing uncertainty [0]. For that dataset, the request creator may introduce uncertainty by specifying the incorrect location for the service request or by the service request submission platform. Hence a point may be incorrectly positioned. We assume that this uncertainty is minimal as the relevant public service provider was able to locate and complete all the service requests we selected for our experiment.

Finally, we recognise that not all processes will possess a significant amount of spatial variation. For example, for centralised processes, the process performers may all be co-located. For these processes, spatial features are not likely to significantly contribute to the accurate prediction of the remaining time

7 CONCLUSION AND FUTURE WORK

This study has proposed an approach to incorporate spatial context into event logs and performed a comparative analysis of spatial features against other contextual features. It found that spatial features improve the predictive power of the model and that spatial ensemble approaches yielded the best result for processes that are likely to exhibit spatial point processes

As mentioned in section 1, incorporating the spatial context into the event log facilitates research opportunities which extend beyond predictive process monitoring. Referencing the refined process mining framework (see [**Error! Reference source not found.**]), it 'opens the door' to performing spatial process discovery (process models by location) and conformance testing. For 'Recommend', it would be possible to incorporate spatial context into the recommendation (i.e. The model recommends a user in location A performs activity X; however, suggests a user in location B performs activity Y).

Besides, a spatio-temporal extension to Tobler's law is proposed as follows: "everything is related to everything else but near and recent things are more related than distant things" [1]. As a result, we expect that a spatio-temporal model will make a more significant contribution to remaining-time predictive monitoring. In future work, we intend to attempt to tackle a number of these opportunities.

REFERENCES

[1] Bennett L., D'Acosta J. and Vale F. (2018) in 'Spatial Data Mining II: A Deep Dive Into Space-Time Analysis' [Online] Available at https://www.youtube.com/watch?v=0aV6HHwJuo4&t=364s [Accessed 30 May 2019]

[2] Breuker, D., Matzner, M., Delfmann, P. and Becker, J., 2016. Comprehensible Predictive Models for Business Processes. Mis Quarterly, 40(4), pp.1009-1034.

[3] Cesario E., Folino F., Guarascio M., Pontieri L. (2016) A Cloud-Based Prediction Framework for Analyzing Business Process Performances. In: Buccafurri F., Holzinger A., Kieseberg P., Tjoa A., Weippl E. (eds) Availability, Reliability, and Security in Information Systems. CD-ARES 2016. Lecture Notes in Computer Science, vol 9817. Springer, Cham Dua, S and Chowriappa, P (2012) Data Mining for Bioinformatics, Boca Raton: CRC Press

[4] de Leoni, M. (Massimiliano); Mannhardt, Felix (2015): Road Traffic Fine Management Process. 4TU.ResearchData. Dataset. https://doi.org/10.4121/uuid:270fd440-1057-4fb9-89a9-b699b47990f5

[5] Evermann, J., Rehse, J.R. and Fettke, P., 2017. Predicting process behaviour using deep learning. Decision Support Systems, 100, pp.129-140.

[6] Folino F., Guarascio M., Pontieri L. (2012) Discovering Context-Aware Models for Predicting Business Process Performances. In: Meersman R. et al. (eds) On the Move to Meaningful Internet Systems: OTM 2012. OTM 2012. Lecture Notes in Computer Science, vol 7565. Springer, Berlin, Heidelberg

[7] Hengl, T., Nussbaum, M., Wright, M.N., Heuvelink, G.B. and Gräler, B., 2018. Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables. *PeerJ*, *6*, p.e5518.

[8] Longley, P.A., Goodchild, M.F., Maguire, D.J. and Rhind, D.W., 2015. *Geographic information science and systems*. John Wiley & Sons.

[9] Miller, H.J., 2004. Tobler's first law and spatial analysis. *Annals of the Association of American Geographers*, *94*(2), pp.284-289.

[10] Pasquadibisceglie, V., Appice, A., Castellano, G. and Malerba, D., 2019, June. Using Convolutional Neural Networks for Predictive Process Analytics. In 2019 International Conference on Process Mining (ICPM) (pp. 129-136). IEEE.

[11] Rogge-Solti, A. and Weske, M., 2013, December. Prediction of remaining service execution time using stochastic petri nets with arbitrary firing delays. In International Conference on Service-Oriented Computing (pp. 389-403). Springer, Berlin, Heidelberg.

[12] Rozinat, A., Wynn, M.T., van der Aalst, W.M., ter Hofstede, A.H. and Fidge, C.J. (2009) 'Workflow simulation for operational decision support.' *Data & Knowledge Engineering*, 68(9), pp.834-850.

[13] Senderovich A., Di Francescomarino C., Ghidini C., Jorbina K., Maggi F.M. (2017) 'Intra and Inter-case Features in Predictive Process Monitoring: A Tale of Two Dimensions'. In: Carmona J., Engels G., Kumar A. (eds) Business Process Management. BPM 2017. Lecture Notes in Computer Science, vol 10445. Springer, Cham

[14] Tax, N., Verenich, I., La Rosa, M. and Dumas, M., 2017, June. Predictive business process monitoring with LSTM neural networks. In International Conference on Advanced Information Systems Engineering (pp. 477-492). Springer, Cham.

[15] Tobler, W.R., 1970. A computer movie simulating urban growth in the Detroit region. *Economic geography*, *46*(sup1), pp.234-240.

[16] Van der Aalst, W.M. (2016) *Process Mining: Data Science in Action*. 2nd edition. Springer Berlin Heidelberg.

[17] Veldhoen, J. (2011) *The Applicability of Short-term Simulation of Business Processes for the Support of Operational Decisions*, Masters Thesis, Technische Universiteit Eindhoven, Available at**:** http://alexandria.tue.nl/extra2/afstversl/tm/Veldhoen%202011.pdf

[18] van Dongen, B.F. (Boudewijn) (2017) BPI Challenge 2017. Eindhoven University of Technology. Dataset. https://doi.org/10.4121/uuid:5f3067df-f10b-45da-b98b-86ae4c7a310b

[19] van Dongen, Boudewijn (2019): BPI Challenge 2019. 4TU.ResearchData. Dataset. https://doi.org/10.4121/uuid:d06aff4b-79f0-45e6-8ec8-e19730c248f1

[20] van Dongen, B.F. (Boudewijn) (2020) BPI Challenge 2020. 4TU.Centre for Research Data. Dataset. https://doi.org/10.4121/uuid:52fb97d4-4588-43c9-9d04-3604d4613b51

[21] Verenich, I., Nguyen, H., La Rosa, M., & Dumas, M. (2017) White-box prediction of process performance indicators via flow analysis. In *Proceedings of the 2017 International Conference on Software and System Process Pages*, ACM, Paris, France, pp. 85-94.

[22] Verenich, I., Dumas, M., Rosa, M.L., Maggi, F.M. and Teinemaa, I., 2019. Survey and cross-benchmark comparison of remaining time prediction methods in business process monitoring. *ACM Transactions on Intelligent Systems and Technology (TIST)*, *10*(4), pp.1-34.