Predicting Activities in Business Processes with LSTM Recurrent Neural Networks

Edgar Tello-Leal[1]; Jorge Roa[2]; Mariano Rubiolo[2,3]; Ulises M. Ramirez-Alcocer[1]

[1]Faculty of Engineering and Science, Autonomous University of Tamaulipas, Victoria, Tamaulipas, Mexico, 87000
[2]CIDISI, Santa Fe Regional Faculty, National Technological University, Lavaisse 610, (S3004EWB) Santa Fe, Argentina.
[3]Research Institute for Signals, Systems and Computational Intelligence, sinc(i), FICH/UNL-CONICET, Ciudad Universitaria, (3000) Santa Fe, Argentina.

ABSTRACT

The Long Short-Term Memory (LSTM) Recurrent Neural Networks provide a high precision in the prediction of sequences in several application domains. In the domain of business processes it is currently possible to exploit event logs to make predictions about the execution of cases. This article shows that LSTM networks can also be used for the prediction of execution of cases in the context of an event log that originates from the IoT and Industry 4.0 domain. This is a key aspect to provide valuable input for planning and resource allocation (either physical or virtual), since each trace associated with a case indicates the sequential execution of activities in business processes. A methodology for the implementation of an LSTM neural network is also proposed. An event log of the industry domain is used to train and test the proposed LSTM neural network. Our preliminary results indicate that the prediction of the next activity is acceptable according to the literature of the domain.

Keywords - LSTM, event log, process mining, business process

1. INTRODUCTION

In a knowledge-based economy, public and private organizations require proper knowledge asset management to maintain a competitive advantage in global markets or in government services. With the advent of robotics, machine learning, and 5G networks there will be a wealth of opportunities for cooperation between robots and humans improving productivity and speeding up the delivery of services for citizens. In this context, the Business Process Management (BPM) is considered a key component to managing the life-cycle of business processes that orchestrate the activities performed in organizations as well as the resources (humans, robots, or information systems) that execute such activities.

A *business process* consists of a set of activities that are performed in a coordinated way in an organization in a technical environment, and have at least one correlated business goal [1]. The standard language for modeling business processes is BPMN (Business Process Modeling Notation) [2]. Information technologies in general and information systems in particular play an important role in the management of business processes, because a large number of activities that organizations perform are supported by information systems. Several types of activities contained in the business processes can be executed automatically by the information systems, without the participation of a human.

The convergence of solutions and products towards the BPM and the Service Oriented Architecture (SOA) paradigm adopted for industrial systems contributes to the improvement in the reactivity and performance of industrial processes such as manufacturing or logistics among others [3]. This is leading to a situation where information is registered in event logs, making it available in near-real time based on asynchronous events, and to business-level applications that are able to use high-level information for various purposes, such as diagnosis, performance indicators, or traceability. [3].

In this context, predicting the behavior of a business process, i.e. exploiting event logs to make predictions about the execution of activities [4], is a key aspect in order to provide valuable input for planning and resource allocation [4]. There are two main factors for the growing interest in predicting business process behavior. On the one hand, with the advent of 5G and the Internet of Things (IoT) in the context of Industry 4.0 [3], more and more events are recorded due to the great number of devices connected to the Internet, providing detailed information about the history of business processes. On the other hand, there is a need to improve and support business processes in competitive and rapidly changing environments.

Process mining techniques [5, 6] are capable of extracting knowledge from event logs, commonly available in information systems. These techniques provide new means to discover, monitor and improve business processes in a variety of application domains. However, standard process mining techniques cannot deal with predicting process behavior.

The recurrent neural network (RNN) architecture has become a model of the neural network implemented in different domains, due to its natural ability to process sequential entries and to know their long-term dependencies [7]. Unlike the feed-forward neural network, the RNN neurons are connected to each other in the same hidden layer and a training function is applied to the hidden states repeatedly [7]. The Long Short-Term Memory (LSTM) neural network is an extension of the RNN, which has achieved excellent performance in various tasks, especially for sequential problems [8], [9], [10]. The implementation of LSTM neural networks for the discovery of events or activities of a business process through predictive analysis can be considered an important strategy as a technique of process mining and has been used with success in this domain [4, 11].

In this work, we propose an approach for the discovery of events and activities of a business process through predictive analysis from traces contained in event logs taken from the IoT and Industry 4.0 domain. The predictive model is based on an LSTM recurrent neural network that is trained with event logs, enabling the prediction of the following activity in a trace of execution that follows another activity or a set of activities given as input. In order to validate the approach and show the applicability to the proposed domain we present preliminary results based on a dataset with 255 traces. The test carried out on the trained LSTM network shows that it has the capacity to predict the next activity of a business process model.

This work is structured as follows. Section 2 presents the background. Section 3 presents an introductory example. Section 4 introduces an approach to predict business process behavior. Section 5 shows the results. Section 6 presents related work. Finally, Section 7 concludes this work and proposes future work.

2. BACKGROUND

2.1 Process Mining

Process mining is an area of research that is located, on the one hand, between computational intelligence and data mining, and on the other hand, between business process modeling and analysis. There are several areas that are included in process mining, such as process discovery, compliance verification, process improvement, organizational mining, process model extension, automatic repair of process models, case prediction, automatic construction of models based on simulation, and recommendations based on the history of execution of processes.

In process mining, it is assumed that it is possible to record events sequentially since each event has a reference to an activity and is related to a particular case (an instance of the process) [6]. Then, the input data in the process mining is an event log. An *event log* is a hierarchically structured file with data on the executions of business processes [12]. This file contains data on several executions of the same business process. An *event* is the atomic part of the execution of a specific process and may contain a large number of attributes. Event data, generated by information systems, is usually found as updates to a state (for example, the status of "sent invoice" changes to the status "paid invoice"), or also as activity records (for example, " email sent to the client "). A *trace* is a set of events that belong to the same execution of a business process. Therefore, event logs can contain additional information about events, such as the user who runs the activity or device that initiates the activity, the time the event started, the duration of the event, among others.

The main tasks of process mining are discovery, compliance and process improvement [6]. *Process discovery* consists of using an event log as input and producing a business process model without using a-priori information [6]. The model discovered is typically a business process model represented using a graphical notation such as the BPMN language [2], Petri nets [13, 14, 15], Event-driven Process Chains (EPC) [16], or UML activity diagrams [17]. *Process conformance* consists of comparing a business process model with the event record generated by the execution of the same process model [6]. Conformance verification can be used to evaluate whether the information stored in the event log is equivalent to the model and vice versa. *Process improvement* consists of extending or improving an existing process model using the stored information of the current process in the event log.

2.2 LSTM Neural Networks

Recurrent neural networks (RNN) with Long Short-Term Memory (LSTM) emerged as an effective and scalable model for learning problems related to sequential data [18]. RNNs have two types of input, the present, and the recent past. RNN use both types of input to determine how they behave with respect to new data. This means that the output of a RNN at time step *t*-1 affects its output at time step *t*. LSTMs are general and effective at capturing longterm temporal dependencies [18].

The information contained in LSTMs are outside the normal flow of the recurrent network in a gated cell. Information can be stored, written or read from a cell, similar to data in a computer's memory. The cell makes decisions about what should be stored and when it should be allowed to read, write and delete, through gates that open and close. These gates are implemented with the multiplication of elements by sigmoids, which are all in the range of 0-1.

3. INTRODUCTORY EXAMPLE

Industries work to increase the overall effectiveness of their plants and equipments, in order to get better system integration, availability, maintainability, performance, quality, or functionality. [3]. The example analyzed in this paper is based on [19] and focuses on the control of a plant to increase its overall performance, including predictive maintenance. The plant produces parts made of metal such as spurs, fastener, ball nuts, discs, tubes, wheel shafts, or clamps. To build these parts there are 28 machines for lapping, milling, turning, sinking, wire cutting, turning and milling, laser marking, and round and flat grinding.

Table 1 shows an excerpt of the event log of the process that controls the logic of the plant. The complete log can be found in [19]. The dataset contains process data from a production process, including data on cases, activities, resources, timestamps, and more data fields.

It is known that, in general, with higher quality and information coming from sensors in the process and the critical equipment for the control of the plant, it is possible to improve the plant operation and production planning [3]. In this scenario, exploiting event logs, that provide detailed information about the history of business processes and register sensor information, for predicting the next activity to be executed in a business process is important to provide valuable input for planning and resource allocation, such as

preparing a machine or a resource to be ready and on time for production.
Besides physical resources, the planning and allocation of resources could also refer to the cloud. With the advent of the Industry 4.0 and the automation of cyber-physical systems, many information systems are being executed in the cloud. In this context, on-demand elasticity is a key aspect. In cloud computing, elasticity is defined as "the degree to which a system is able to adapt to workload changes by provisioning and de-provisioning resources in an autonomic manner, such that at each point in time the available resources match the current demand as closely as possible" [20]. Knowing from advance which one is the next activity of a business process that is going to be executed is key to pro-actively release or reserve resources to support elasticity on the cloud.

4. PREDICTING BUSINESS PROCESS ACTIVITIES

This Section introduces a methodology to predict activities in business processes from information registered in event logs derived from the execution of business processes.
The proposed methodology is based on the LSTM neural network and consists of three phases: 1) pre-processing of the event log, 2) categorization, and 3) prediction model based on LSTM, as shown in Figure 1.

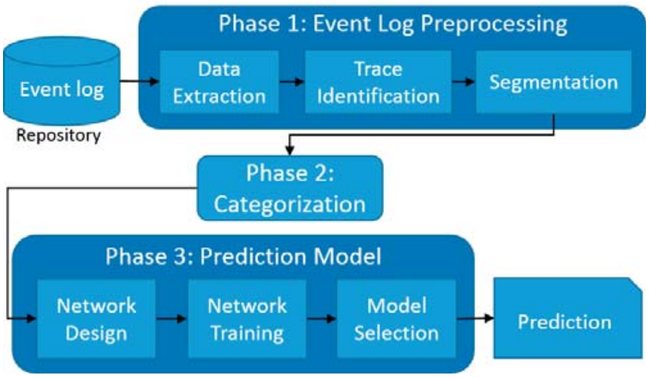

Figure 1 – The methodology for predicting activities of a business process using the proposed approach.

4.1 Phase 1: Event Log Preprocessing

The pre-processing phase of the event log consists of the following stages:

Data Extraction. A detailed analysis of the event logs is performed (.XES file format), which allows the identification of the different attributes contained in the event log, allowing to select the attributes required for a prediction, in this case, the attribute "*activity*".

Trace Identification. It consists of identifying and obtaining the traces, with their respective events. Then, the traces are added in a text file maintaining their order of appearance.

Segmentation. A segmentation task is applied to the text file generated in the previous stage, which consists of creating a list of all the events of a trace, using a criterion of separation between each event. Then, each event is represented as a unique integer, allowing the traces to be converted into a sequence of integers, generating two sequence lists of "integers", the first list consisting of input activities (X), and the second list of output activities (Y). Finally, the sequence list of input activities is transformed into a two-dimensional matrix (number of sequences, the maximum length of sequences).

4.2 Phase 2: Categorization

The intermediate categorization phase consists of a process to categorize the sequence of integers corresponding to the output activities (Y), in a one hot encoding representation type, specifying that the number of classes will be equal to the size of the vocabulary.

4.3 Phase 3: Prediction Model

The prediction model phase based on LSTM network is composed of the following stages:

Network Design. It consists of generating a design of the LSTM network by layers. First, an input layer is generated (embedding) to the network, then the hidden layer (LSTM units) is created so that finally an output layer is built. In each of these layers, some necessary parameters are defined.

Network Training. The training of the LSTM network is carried out using as training data the sequence list of integers represented by the activities contained in the matrix (X) and in the representation one hot type (Y).

Model Selection. The results of the training will allow choosing a model of the LSTM network as the final model to be implemented. A network with training with a high degree of accuracy should be selected as the model to make the predictions. Otherwise, it is recommended to modify the design of the network, adjusting the required parameters and execute network training again.

Prediction. It is the output generated by the LSTM neural network, which through a training stage allows predicting the next activity in a business process model, from an input activity or a sequence of input activities, which is explained in the following sections of the document.

4.4 Implementation

The proposed approach is based on the definition of a recurrent neural network LSTM, considered as a network of a special structure consisting of memory blocks and memory cells, together with the gate units that contain them [21], i.e, an LSTM unit consists of a cell and three gates (input, forget, and output). Through this special structure, an LSTM network can select which information is forgotten or remembered.

Table 1 – Excerpt of event log based on [19]

Case ID	Activity	Resource	Time-stamp
Case 1	Turning & Milling - Machine 4	Machine 4 - Turning & Milling	29/1/2012 23:24
Case 1	Turning & Milling - Machine 4	Machine 4 - Turning & Milling	30/1/2012 05:44
Case 1	Turning & Milling - Machine 4	Machine 4 - Turning & Milling	30/1/2012 06:59
Case 1	Turning & Milling - Machine 4	Machine 4 - Turning & Milling	30/1/2012 07:21
Case 1	Turning & Milling Q.C.	Quality Check 1	31/1/2012 13:20
Case 1	Laser Marking - Machine 7	Machine 7- Laser Marking	1/2/2012 08:18
Case 1	Lapping - Machine 1	Machine 1 - Lapping	14/2/2012 00:00
Case 1	Lapping - Machine 1	Machine 1 - Lapping	14/2/2012 00:00
Case 1	Lapping - Machine 1	Machine 1 - Lapping	14/2/2012 09:05
Case 1	Lapping - Machine 1	Machine 1 - Lapping	14/2/2012 09:05
Case 1	Round Grinding - Machine 3	Machine 3 - Round Grinding	14/2/2012 09:13
Case 1	Round Grinding - Machine 3	Machine 3 - Round Grinding	14/2/2012 13:37
Case 1	Final Inspection Q.C.	Quality Check 1	16/2/2012 06:59
Case 1	Final Inspection Q.C.	Quality Check 1	16/2/2012 12:11
Case 1	Final Inspection Q.C.	Quality Check 1	16/2/2012 12:43
Case 1	Packing	Packing	17/2/2012 00:00
...	...	...	...
Case 253	Flat Grinding - Machine 11	Machine 11 - Grinding	10/1/2012 11:59
Case 253	Lapping - Machine 1	Machine 1 - Lapping	11/1/2012 00:00
Case 253	Laser Marking - Machine 7	Machine 7- Laser Marking	11/1/2012 14:23
Case 253	Final Inspection Q.C.	Quality Check 1	15/1/2012 06:50
Case 253	Packing	Packing	16/1/2012 00:00
Case 253	Packing	Packing	16/1/2012 00:00
Case 254	Laser Marking - Machine 7	Machine 7- Laser Marking	2/1/2012 10:15
Case 254	Flat Grinding - Machine 11	Machine 11 - Grinding	2/1/2012 14:00
Case 254	Flat Grinding - Machine 11	Machine 11 - Grinding	3/1/2012 17:04
Case 254	Flat Grinding - Machine 11	Machine 11 - Grinding	4/1/2012 10:28
Case 254	Final Inspection Q.C.	Quality Check 1	4/1/2012 15:26
Case 254	Packing	Packing	6/1/2012 00:00
Case 254	Final Inspection Q.C.	Quality Check 1	6/1/2012 10:24
Case 255	Turning - Machine 8	Machine 15 - Turning	2/1/2012 07:00
Case 255	Turning Q.C.	Quality Check 1	5/1/2012 13:58
Case 255	Laser Marking - Machine 7	Machine 7- Laser Marking	10/1/2012 09:22
Case 255	Final Inspection Q.C.	Quality Check 1	11/1/2012 10:22
Case 255	Packing	Packing	16/1/2012 00:00

The multiplicative input gate units are used to avoid the negative effects that unrelated inputs can create. The input gate controls the input flow to the memory cell, and the output gate controls the output sequence of the memory cell to other LSTM blocks.
The forget gate in the structure of the memory block is controlled by a single-layer of the neural network. At a time t, the components of the LSTM unit are updated by means of equation *(1)* [22, 9].

$$f_t = \sigma(W[x_t, h_{t-1}, C_{t-1}] + b_f) \tag{1}$$

where x_t is the input sequence, h_{t-1} is the previous block output, C_{t-1} is the previous LSTM block memory, and b_f is the polarization vector. W represents separate weight vectors for each input and σ is the logistic sigmoid function. The sigmoid activation function, which is the output of the forgetting gate, is applied to the previous memory block by multiplication by elements. Therefore, the degree to which the previous memory block will be effective in the current LSTM is determined. If the activation output vector contains values close to zero, the previous memory will be forgotten.
The input gate is a section where the new memory is created by a simple neural network with the activation function $tanh$ and the previous memory block. These operations are calculated using equations *(2)* and *(3)*.

$$i_t = \sigma(W[x_t, h_{t-1}, C_{t-1}] + b_i) \tag{2}$$

$$C_t = f_t.C_{t-1} + i_t.tanh(W[x_t, h_{t-1}, C_{t-1}] + b_c) \tag{3}$$

Finally, the output gate is the section where the probabilities of the current LSTM block are generated [9]. The output is calculated by means of equations *(4)* and *(5)*.

$$o_t = \sigma(W[x_t, h_{t-1}, C_{t-1}] + b_o) \tag{4}$$

$$h_t = tanh(C_t).o_t \quad (5)$$

5. RESULTS

Keras [23] was used for the implementation, which is a Python library that allows building models of deep learning networks. The implementation parameters of the LSTM network are presented in Table 2.

Table 2 – Configuration parameters of the LSTM neural network

Parameter	Value
epochs	500
batch size	20
optimizer	Adam
loss	categorical_crossentropy
LSTM units	50

The LSTM neural network was trained with an event log described in Section 3. This event log includes 255 traces of the business process model. There are 56 different activities contained in the log. The number of sequences identified during the network training was 4541. The LSTM network accepts as input data an activity, in order to predict the next activity of the sequence. The neural network was configured to predict three outputs per instance, ordered by a higher to lower probability. The objective is to know the prediction capacity of the neural network of the next activity. The algorithms and datasets can be accessed at `\http://dx.doi.org/10.17632/trskzyg3j9.1`.

Table 3 summarizes the activities in the event log and their acronyms. The name of the activities in the table are acronyms from the real name included in the event log. For instance: "Turning & MillingQ.C." (TMQC), "LaserMarking-Machine7" (LMM7).

Table 4 presents an extract of the results obtained in the prediction of the neural network using the Event Log presented before. In the column "Input Activity" it is mentioned the activity used as a new input for the LSTM network in the prediction process. The "Target Activity" is the expected activity (or activities) for the corresponding input activity, that is, the activities with the highest probability of prediction by the neural network, based on the weights of each activity. Each row in the table shows a case of prediction of the next activity from the input one. The "Output Activity" column presents the activities that the LSTM neural network predicted from the input activity.

The test carried out on the trained LSTM network shows that it has the capacity to predict the next activity of a business process model. For the cases number 3, 5, 7 and 8, the network was able to predict the exact next activity. For instance, in the third case, receiving the GRM27 as input, the LSTM network was able to predict the expected FIQC (the output activity is included in the target activity list, with the highest probability). In other cases, as the number 1, 2, 4 and 6, the most of the target activities were identified, missing one that was not predicted. For instance, in the first case, were predicted the LMM7 and LPM1 activities, but not the TMM4. However, in these cases, the next activity that is predicted is the one with the highest probability. Furthermore, in the case number 9 in which the prediction obtain the desired activity but one of them was not expected in the target. In this instance, using the FM15 as input, it was expected that the LSTM throw as output only the PACK, but the TMQC was also included as a response. At last, in the case number 10, the target activity is FIQC, but the LSTM network predicts two activities that do not match with activity what was expected.

Table 3 – List of activities and acronyms

Activity name	Acronym
Turning&Milling-Machine	TMM
Turning&MillingQ.C.	TMQC
LaserMarking-Machine	LMM
RoundGrinding-Machine	RGM
RoundQ.C.	RQC
FinalInspectionQ.C.	FIQC
Packing	PACK
TurningQ.C.	TQC
GrindingRework-Machine	GRM
GrindingRework	GR
WireCut-Machine	WCM
Fix-Machine	FM
NitrationQ.C.	NQC

Table 4 – An extract of the prediction from LSTM

No.	Input Activity	Target Activity	Output Activity 1	Output Activity 2
1	TMQC	LMM7 | LPM1 | TMM4	LMM7	LPM1
2	PACK	FIQC | FM15	FIQC	
3	GRM27	FIQC	FIQC	
4	GR	LPM1 | TMQC	LPM1	
5	WCM18	TQC	TQC	
6	RGM19	RGM12 | FIQC	RGM12	
7	NQC	TMM5 | TMQC	TMM5	TMQC
8	RQC	PACK | FIQC	PACK	FIQC
9	FM15	PACK	PACK	TMQC
10	FGM26	FIQC	PACK	MM14

6. RELATED WORK

The development of technological solutions for event log analysis for business process discovery using the principles of data mining has been previously studied in [6, 12]. The most relevant proposals that are related to the approach proposed in this research work are discussed in this section. However, existing techniques are not able to predict at runtime the next activities that are going to be executed in a business process. We expect that techniques based on LSTM neural networks, like the proposed in this work, can also be of help in the discovery of business process models.

There are a few approaches using patterns and statistical models to predict activities in business processes. The approach described in [24], aims at identifying partial business process models to be used for training predictive models. It infers two types of predictive models. The first model is used to identify frequent partial processes in form of frequent activity sequences, the sequences are extracted using a frequent pattern mining algorithm and are represented in form of sequence trees. The second model is used for the estimation of the completion time, associating each node of the tree a specific prediction model that takes into account attributes such as the performer of each activity, the cost associated to the event or the place where the event is performed. In [25], authors propose Markov chains to estimate the instance-specific probabilistic process model (PPM) that can take as input a running process instance, and compute the probability of execution of a particular task in that instance. An instance-specific PPM serves as a representation to predict the likelihood of different outcomes. Similarly, in [26] authors propose methods which use sequential k-nearest neighbor and higher order Markov models for predicting the next tasks in a business process instance. The sequential k-nearest neighbor technique is applied when the default prediction is required (when the given sequence cannot be found in the transition matrix). The matching procedure is applied in order to extract the given sequence's most similar sequences (patterns) from the traces.

The use of neural networks to predict activities is a recent field. In [11], the author presents an approach to predict the next process event using deep learning based on an LSTM recurrent neural network. The proposed approach is based on the prediction of the next word in a sentence (natural language processing), i.e., by interpreting process event logs as text, process traces as sentences, and process events as words. The implementation of the LSTM network is through an architecture of two hidden layers. The approach is evaluated on two real datasets commonly used in the state-of-the-art, presenting better results in prediction precision. Similarly, in [4], authors propose a method based on LSTM that allows predicting the next activity and its time-stamp per case contained in an event record. The authors mention that the results of the neural network are outstanding using real-life data sets compared to traditional methods of automatic learning. Additionally, the authors conclude that predicting the next activity and its time-stamp via a single model yields a higher accuracy than predicting them using separate models.

In our proposal, the sequences of the traces identified in the event record are conserved, which allows us to improve the accuracy of the model. In addition, the use of a one-hot encoding is proposed, allowing to increase the percentage of the prediction of the next activity. Additionally, the proposed LSTM neural network has the ability to predict the next activity from one or more input activities, as well as to predict one to three output activities, ordered from highest to lowest probability of occurrence.

7. CONCLUSIONS AND FUTURE WORK

With the advent of 5G, IoT, and Industry 4.0 there is a growing interest in predicting business process behavior. More and more events are recorded due to the great number of devices connected to the Internet, providing detailed information about the history of business processes.

In this work, a methodology for the prediction of business process activities has been proposed. The methodology is based on an LSTM recurrent neural network and exploit event logs to make predictions about the execution of cases. This is key to provide valuable input for planning and resource allocation (either physical or virtual), specially in competitive and rapidly changing environments.

The proposed methodology considers event log preprocessing, categorization, and prediction model, and allows identifying the phases required to predict the next activity or event, through the implementation of the LSTM neural network. The novelty of this approach resides in the use of event logs that originates from an IoT domain within the context Industry 4.0.

In order to validate the approach and show the applicability to the proposed domain we present preliminary results based on a dataset with 255 traces. The test carried out on the trained LSTM network shows that it has the capacity to predict the next activity of a business process model. However, in order to fully validate the approach, more tests are needed.

It is also necessary to take into account that, since the LSTM cells maintain the state, the order in which the traces are used to train the network has a direct effect on the result. In addition, the selection of the trace set of an event log (training data) also influences results, and hence, results obtained with a particular sample may not be generalized to other cases (or event logs).

The predictive analysis implemented in this work allows us to obtain useful information to determine the next activity to be executed based on event logs. Our next steps deal with expanding our study to other event logs with a greater number of traces, as well as considering including two or more classes to predict the next activity or event.

REFERENCES

[1] M. Weske, "Business process management architectures," in *Business Process Management*. Springer, 2012, pp. 333–371.

[2] BPMN 2.0, "Business Process Modeling Notation 2.0," http://www.omg.org/spec/BPMN/2.0/, 2011.

[3] A. W. Colombo, T. Bangemann, S. Karnouskos, J. Delsing, P. Stluka, R. Harrison, F. Jammes, J. L. Lastra *et al.*, *Industrial cloud-based cyber-physical systems*. Springer, 2014.

[4] N. Tax, I. Verenich, M. La Rosa, and M. Dumas, "Predictive business process monitoring with LSTM neural networks," in *Advanced Information Systems Engineering*, E. Dubois and K. Pohl, Eds. Cham: Springer International Publishing, 2017, pp. 477–492.

[5] W. Van Der Aalst, A. Adriansyah, A. K. A. De Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. van den Brand, R. Brandtjen, J. Buijs *et al.*, "Process mining manifesto," in *International Conference on Business Process Management*. Springer, 2011, pp. 169–194.

[6] W. M. Van der Aalst, *Process mining: data science in action*. Springer, 2016.

[7] W. Xia, W. Zhu, B. Liao, M. Chen, L. Cai, and L. Huang, "Novel architecture for long short-term memory used in question classification," *Neurocomputing*, vol. 299, pp. 20–31, 2018.

[8] Y. Li and H. Cao, "Prediction for tourism flow based on LSTM neural network," *Procedia Computer Science*, vol. 129, pp. 277–283, 2018.

[9] B. Cortez, B. Carrera, Y.-J. Kim, and J.-Y. Jung, "An architecture for emergency event prediction using LSTM recurrent neural networks," *Expert Systems with Applications*, 2017.

[10] F. Liu, Z. Chen, and J. Wang, "Video image target monitoring based on RNN-LSTM," *Multimedia Tools and Applications*, pp. 1–18, 2018.

[11] J. Evermann, J.-R. Rehse, and P. Fettke, "Predicting process behaviour using deep learning," *Decision Support Systems*, vol. 100, pp. 129 – 140, 2017, smart Business Process Management.

[12] T. Baier, J. Mendling, and M. Weske, "Bridging abstraction layers in process mining," *Information Systems*, vol. 46, pp. 123–139, 2014.

[13] K. Jensen and L. M. Kristensen, *Coloured Petri Nets: Modelling and Validation of Concurrent Systems*, 1st ed. Springer Publishing Company, Incorporated, 2009.

[14] J. Roa, O. Chiotti, and P. Villarreal, "Behavior alignment and control flow verification of process and service choreographies," *Journal of Universal Computer Science*, vol. 18, no. 17, pp. 2383–2406, sep 2012.

[15] E. Tello-Leal, O. Chiotti, and P. D. Villarreal, "Software agent architecture for managing inter-organizational collaborations," *Journal of applied research and technology*, vol. 12, no. 3, pp. 514–526, 2014.

[16] F. Gottschalk, W. M. van der Aalst, and M. H. Jansen-Vullers, "Merging event-driven process chains," in *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*. Springer, 2008, pp. 418–426.

[17] UML 2.0, "Unified Modeling Language 2.0," http://www.omg.org/spec/UML/2.0/, 2005.

[18] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space odyssey," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 10, pp. 2222–2232, 2017.

[19] D. Levy, "Production Analysis with Process Mining Technology. Dataset," 2014. [Online]. Available: https://data.4tu.nl/repository/collection:event_logs

[20] H. N. Roman, S. Kounev, and R. Reussner, "Elasticity in cloud computing: What it is, and what it is not," in *Proceedings of the 10th International Conference on Autonomic Computing (ICAC 2013)*, 2013.

[21] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *15th Annual Conference of the International Speech Communication Association*, September 2014, pp. 338–342.

[22] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] Chollet, F. Keras, "Keras: The Python Deep Learning library," https://github.com/fchollet/keras, 2018.

[24] M. Ceci, P. F. Lanotte, F. Fumarola, D. P. Cavallo, and D. Malerba, "Completion time and next activity prediction of processes using sequential pattern mining," in *Discovery Science*, S. Džeroski, P. Panov, D. Kocev, and L. Todorovski, Eds. Cham: Springer International Publishing, 2014, pp. 49–61.

[25] G. T. Lakshmanan, D. Shamsi, Y. N. Doganata, M. Unuvar, and R. Khalaf, "A markov prediction model for data-driven semi-structured business processes," *Knowledge and Information Systems*, vol. 42, no. 1, pp. 97–126, Jan 2015. [Online]. Available: https://doi.org/10.1007/s10115-013-0697-8

[26] M. Le, B. Gabrys, and D. Nauck, "A hybrid model for business process event prediction," in *Research and Development in Intelligent Systems XXIX*, M. Bramer and M. Petridis, Eds. London: Springer London, 2012, pp. 179–192.