



Deep Learning Based on TensorFlow and Keras for Predictive Monitoring of Business Process Execution Delays

Walid Ben Fradj^(✉), Mohamed Turki, and Faiez Gargouri

MIRACL Laboratory, ISIMS, University of Sfax, P.O. Box 242, 3021 Sfax, Tunisia
walid.benfradj@isimsf.u-sfax.tn, mohamed.turki@isims.usf.tn,
faiez.gargouri@usf.tn

Abstract. In order to enhance their performance and responsiveness, organizations must identify, manage, and monitor all business processes that involve crucial knowledge. This can be achieved through a multidisciplinary approach that combines Knowledge Management, Business Process Management, and Process Mining. To achieve these objectives, the implementation of an automated computer system for managing business processes has become paramount. In this context, we adopt the CRISP-DM approach to propose a new method called BPEDPM (Business Process Event Data Predictive Monitoring) for predictive process monitoring. This method leverages process mining techniques to exploit the execution data from a Business Process Management System (BPMS) workflow engine. Particularly, in the modeling phase, we apply deep learning based on the TensorFlow and Keras tools. To demonstrate the applicability of the BPEDPM method, we have developed an intelligent system named iBPMS4PED to predict the execution times of business processes. The research focuses on the incoming mail management process within a group health insurance context.

Keywords: Business Process Management · Process Mining · Knowledge Management · Machine Learning · Deep Learning · TensorFlow · Keras · CRISP-DM

1 Introduction

Nowadays, businesses are increasingly realizing the vital importance of optimizing their knowledge exploitation while implementing a process-oriented quality management approach. This can be achieved by adopting an interdisciplinary approach that combines Knowledge Management (KM) with Business Process Management (BPM). To improve their performance and enhance adaptability, companies must identify and master all essential processes that may involve implicit or explicit knowledge, which represents a valuable source to be leveraged. To achieve these objectives, the implementation of an automated computer system for business process automation is essential. According to [1], various information systems are linked to processes, including Enterprise Resource Planning (ERP), Workflow Management (WFM), Customer Relationship Management

(CRM), Supply Chain Management (SCM), Product Data Management (PDM), and Business Process Management (BPM). Most of these systems utilize execution engines, also known as workflow engines. These engines serve a dual role: firstly, they facilitate the deployment of processes, and secondly, they record business data in the form of databases, along with technical information related to process execution in the form of event logs. These data can be exploited through process mining techniques, thereby extracting new knowledge that proves valuable for process optimization and decision-making within the organization.

According to Van der Aalst [2], a pioneer in the field of process mining, several types of process mining can be distinguished, with the most commonly used ones being process discovery, conformance checking, performance analysis, comparative process mining, predictive process mining, and action-oriented process mining. Process discovery is particularly crucial in all process mining endeavors. As stated in [3], process discovery algorithms undergo a step of event log filtering, eliminating infrequent activities to retain only dominant behaviors. The resulting output is a graphical model, such as Directly Follows Graphs (DFG), Petri Nets (PN), Process Trees (PT), or Business Process Model and Notation (BPMN). This raises questions: Is it possible to extract various transitions, regardless of their frequency, and enrich a database with all paths followed by process instances? If so, can we predict the execution times of instances based on these transitions? Furthermore, despite a literature review, there is no standard scientific approach to successfully conduct a process mining project. Hence, the idea of following the well-established CRISP-DM (Cross Industry Standard Process for Data Mining) procedure used in data science is being considered.

This article focuses on predictive process monitoring and introduces an innovative method that combines process discovery and predictive process mining. Unlike extracting a graphical model, which requires removing less frequent activities, this method allows extracting all paths followed by process instances in a database. By using these paths as a basis, it predicts whether a process instance will complete its execution on time or with a delay. The adopted approach follows the CRISP-DM model and leverages event logs that record the execution data of a workflow engine, combining process mining techniques to create an intelligent system based on machine learning.

The second and third sections of this article respectively delve into the key concepts related to Knowledge Management (KM) and Business Process Management (BPM), highlighting the current trends that integrate artificial intelligence into these domains. In the fourth section, an overview of predictive process monitoring techniques for business processes is presented. The fifth section provides a detailed account of the method used to predict the execution times of business processes, following the CRISP-DM approach. The sixth section describes the implementation and evaluation of the developed IT solution, applied to a specific business process: the management of incoming mail in the health insurance management activity of the company I-WAY in Tunisia. Finally, the article concludes with a summary and outlines prospective research directions for the future.

2 KM: Knowledge Management

According to Grundstein M. (2000) in [4], capitalizing on enterprise knowledge involves identifying crucial knowledge, preserving it, and making it sustainable, while ensuring widespread sharing and usage to enhance the company's wealth. Explicit knowledge is quantifiable, understandable, directly captured, and expressed by individuals within the organization. On the other hand, tacit knowledge, also known as know-how, is specific to each individual and encompasses their informal technical expertise as well as personal beliefs and aspirations. The knowledge management process, as per the same reference, is based on five interacting facets around crucial knowledge: identification, preservation, valorization, updating, and management. Each of these facets includes sub-processes aimed at addressing associated problems. In [5], the authors address the issue of identifying sensitive business processes, focusing on the facet of crucial knowledge identification. Sensitive business processes require enhanced security management to reduce the risks of compromise and ensure business continuity. To address this, they propose a new methodology called SOPIM (Sensitive Organization Process Identification Methodology), based on a multi-criteria decision-making approach to construct a coherent family of evaluation criteria for identifying these processes. This approach mainly consists of two phases: (1) constructing a decision-maker preference model and (2) using the preference model (decision rules) to rank "potentially sensitive organizational processes." Once identified, these sensitive organizational processes will be modeled, executed, and explored. The research article discussed here specifically focuses on the exploration of these processes.

With the advancements in information technology, a new approach to knowledge management has emerged, known as Intelligent Knowledge Management (IKM) [6]. This innovative approach stands out for leveraging artificial intelligence (AI) tools to harness expert knowledge. According to [7], AI plays a potentially significant role in supporting knowledge management activities, offering various benefits such as:

- Improving predictive analysis through machine learning capabilities to anticipate future events.
- Identifying previously unknown patterns in data.
- Exploring organizational data to discover hidden relationships and correlations.
- Developing new declarative knowledge to enrich the organization's understanding.
- Efficiently collecting, classifying, organizing, storing, and searching explicit knowledge.
- Analyzing and filtering numerous content and communication channels to access relevant information.
- Facilitating knowledge reuse by teams and individuals to optimize process efficiency.
- Connecting people working on similar problems to foster weak ties and expertise sharing.
- Promoting collaborative intelligence and organizational memory sharing.
- Creating a holistic perspective on knowledge sources and potential bottlenecks.
- Establishing more connected coordination systems between different parts of the organization to encourage collaboration.
- Improving the application of localized knowledge by identifying and preparing specific knowledge sources tailored to needs.

- Ensuring equitable access to knowledge without fearing prohibitive social costs.
- And many more advantages.

3 BPM: Business Process Management

Every company, regardless of its size, operates through processes that encompass all its activities and services (e.g., Human Resources, Sales, Quality, Purchasing, etc.). According to [8], Business Process Management (BPM) adopts a process-centric approach, enabling effective monitoring of activities within the organization to improve overall performance and consequently, results. Using a BPM tool provides real-time traceability of exchanges between stakeholders involved in a process, allowing for greater responsiveness through the generation of indicators in the form of alerts or automatic notifications. This facilitates decision-making, accelerates the identification of bottlenecks, ensures adherence to deadlines, and controls production costs of products and services [9]. The BPM lifecycle consists of five phases: Design, Modeling, Execution, Monitoring, and Optimization. Firstly, the Design phase involves identifying existing processes and designing future processes. Next, the Modeling phase graphically represents the model in a manner faithful to reality. Once these preliminary steps are completed, the Execution phase takes place, where Business Process Management is implemented. This step entails interpreting business procedures by an execution engine, which coordinates all interactions between users, system tasks, and IT resources. The Monitoring phase, on the other hand, focuses on regulating individual processes, providing easy access to information about their status and delivering statistics on the performance of one or more processes. Finally, the Optimization phase allows for adjusting processes to minimize costs and optimize efficiency. Within the scope of this research article, particular attention is given to the Monitoring phase, where we position ourselves to closely examine this crucial step in the Business Process Management lifecycle.

The field of Business Process Management has witnessed significant technological advancements with the integration of artificial intelligence. In this context, the American consulting and research company Granter introduced the concept of Intelligent Business Process Management Suite (iBPMS) in 2012. iBPMS represents an evolution of BPM by combining predictive analytics, process intelligence, and emerging technologies with traditional BPM practices. This innovative approach aims to make business process management smarter, more efficient, and better suited to the current challenges faced by businesses.

4 Predictive Monitoring of Business Processes

According to reference [10], the objective of predictive monitoring of business processes is to identify and anticipate potential issues in advance. It enables the implementation of preventive measures to avoid problems and facilitates proactive decision-making. Real-time monitoring techniques allow data to be analyzed in real-time, contributing to decision-making and optimization of ongoing processes. According to [11], predictive process monitoring occurs in real-time during the execution of instances. It is important to note that during the learning phase, the input data for the predictive monitoring

method includes event logs along with supplementary information. These data undergo a coding step to be interpreted by the prediction algorithm. This algorithm creates a prediction model that is applied to the ongoing process instances to determine the predicted output value for each process instance. Most predictive monitoring techniques include an offline component (which involves expensive computations) for generating the prediction model, as well as an online component (faster) that performs predictions based on the generated model.

Event logs play a crucial role as the primary data source for Process Mining techniques, particularly in the domain of predictive process monitoring. Each line in these files contains execution information related to an activity, primarily consisting of the process instance identifier and timestamp. Referring to sources [12] and [13], additional information can also be found in the log, such as the activity's cost or the name of the resource responsible for its execution. Technical characteristics describing event logs include the number of instances, the number of activities, the number of events, and the number of process variants [14].

Predictive process monitoring with a temporal dimension was first explored by [15]. Their prediction approach extracts a transition system from the event log, including additional temporal information. This approach served as a fundamental reference for the work of [12, 16-18]. According to [19], the transition system uses decision trees to predict the execution time and the next activity of a process instance. The research of [16] and [17] extended the approach of [15] by clustering event log traces based on contextual features. According to [18], an approach is used by integrating naive Bayesian regression models and support vector regressions to annotate the transition system. The addition of additional attributes has had a positive impact on the quality of predictions. However, these methods have a major drawback: they assume that the event log used for learning contains all possible process behaviors, which is generally not the case in reality. The approaches proposed by [20] and [19] share similarities. They provide a general framework to enrich the event log with derived information and discover correlations between process characteristics using decision trees. The distinction between these two approaches lies in the fact that the one proposed by [19] excludes infrequent behaviors. This latter approach is similar to the one proposed by [18], except that the process model is a reduced version of the transition system. The issue related to correlation is that numerical values must be discretized, which reduces precision. Two probabilistic methods based on Hidden Markov Model (HMM) are presented in the works of [21] and [22]. These probabilistic approaches allow predicting the probability of future activities, providing insights into the process evolution. In parallel, the approach proposed by [23] uses a generic model based on decision trees, thereby providing decision criteria tailored to the actual objectives of the process. Other approaches from various domains have been proposed for predicting delays. According to [24], the Process Mining approach relies on queuing systems. It involves building an annotated transition system and using non-linear regression algorithms to predict delays. Similarly, according to [25], the proposed approach predicts the remaining time using expressive probabilistic models and is based solely on information concerning the workflow. [26] presents a predictive model based on decision trees. This model assesses the probability of satisfying a user-defined constraint for ongoing instances. A similar approach is also explored in the work of [27].

In this approach, traces are treated as complex symbolic sequences, encoded using two methods: indexed encoding and HMM encoding. [28] employs the same encoding for their analyses. To perform clustering, the dataset is partitioned, and a predictor based on random forests is trained for each group. According to reference [29], natural language processing (NLP) is combined with various classifiers to obtain representative features for each document. Among the predictors, random forests have proven to be the most effective. Subsequently, deep neural network-based approaches emerged. Both works [30] and [22] utilize a recurrent neural network (RNN) with two hidden RNN layers, employing basic LSTM cells to predict the next event. The approach of [22] also integrates an LSTM neural network to predict both the next activity and its execution time. These two approaches do not consider additional attributes, are sensitive to hyperparameter selection, and require lengthy training periods. Furthermore, the approach of [10] utilizes artificial neural networks (ANN) to predict if a process instance exceeds the expected time. Lastly, [31] presents a comparison between two machine learning models (random forest and SVM) and two deep learning models (LSTM and DNN). The results indicate that the LSTM model performs the best.

After reviewing the existing literature, we did not find a standard scientific approach to follow for conducting a predictive process monitoring project. In our work, we opted for the well-established CRISP-DM procedure in data science. Additionally, we observed that most of the event logs used contain business data. For this study, we chose to process execution data from the workflow engine of a BPMS. Furthermore, we noticed that models are typically represented as Petri nets, with a tendency to eliminate less frequent transitions. However, in our approach, we decided to include all paths taken by process instances in the database as additional attributes. Finally, it is essential to highlight that most approaches rely on machine learning rather than deep learning. Long Short-Term Memory (LSTM) neural networks are commonly used in deep learning cases. In our study, we chose to create a deep learning model using the state-of-the-art TensorFlow technology to predict the execution times of business processes.

5 PMBPED (Prediction Method of Business Process Execution Delays)

During its execution, a business process can go through several stages, representing its tasks or activities. The sequence of these tasks forms a case or an instance, and all the information related to this instance is recorded in the form of an event log. For the same process, there can be multiple instances with distinct paths. The execution time of a process depends on the order of tasks and their respective durations. When a process or task exceeds a given time threshold, it may indicate the presence of bottlenecks, enabling decision-makers to optimize and improve the performance of different business processes.

The CRISP-DM (CRoss Industry Standard Process for Data Mining) is a well-established, field-proven procedure that guides data exploration activities, playing a key role in the success of data science. According to [32], CRISP-DM is a widely adopted process both in practice and research. It is an organizational process model that is not limited to a specific technology. It includes descriptions of typical project

phases and tasks within each of these phases, as well as an explanation of the relationships between these tasks. It provides an overview of the data exploration lifecycle. According to [33], CRISP-DM breaks down the exploration process into six main steps: business understanding, data understanding, data preparation, modeling, evaluation, and result released (see Fig.1). In this context, using the CRISP-DM procedure as a foundation, the PBMPED method (see Fig.2) is employed to extract various paths followed by process instances from the event log. Subsequently, the execution time of each path is calculated.

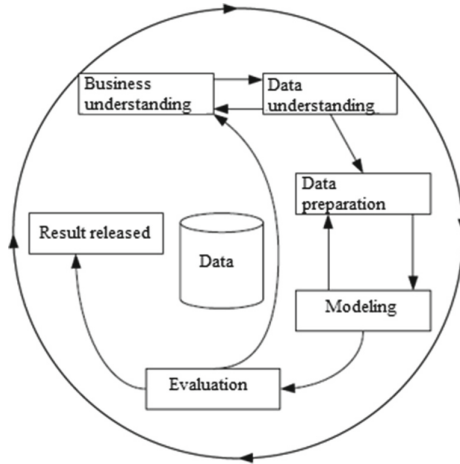


Fig. 1. CRISP-DM procedure [33] p. 216

The objective of PBMPED is to predict the execution times of business processes. This prediction can be valuable for enabling decision-makers, in the case of a semi-automatic process, or the system, in the case of an automated process, to choose the most optimal path.

In the first phase (Business understanding), our method aims to optimize overall performance and results by focusing on the business processes of the company and the flow of activities. Every company relies on a set of processes that encompass its activities. With technological advancements, companies increasingly depend on intelligent software solutions to address issues swiftly and enhance their competitiveness. Business Process Management Systems (BPMS) represent a potential solution for managing and controlling these processes. By implementing workflow engines, which are an integral part of BPMS, it becomes possible to separate business data from process execution data.

In the second phase (Data understanding), data related to process execution is recorded by the workflow engine in event logs. Retrieving this data can be done in different ways from a BPMS workflow engine: either from an integrated database within the BPMS, where tables are typically temporary and created in memory but lost upon server restart; or by configuring a REST API to connect to the integrated database and retrieve execution traces, then saving them to disk; or by configuring a separate database (e.g.,

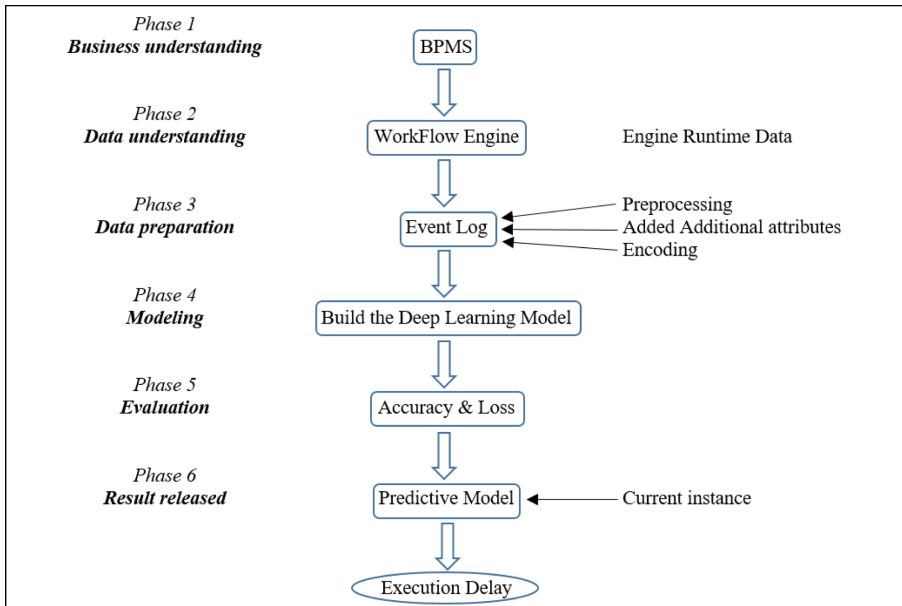


Fig. 2. PBMPED

MySQL, MongoDB, Oracle, etc.) connected to the BPMS through connectors. In our work, we use two event logs in CSV format. The content of these two files is extracted from a MySQL database configured to store execution data of a BPMS workflow engine.

In the third phase (Data preparation), the mere possession of an event log containing execution traces from a workflow engine does not guarantee that the data is ready for modeling. Indeed, the file may have redundancies, missing values, unnecessary data, and categorical variables that require encoding, among other issues. Moreover, the data needs enrichment by adding additional attributes. In the context of our PBMPED method, we propose the following approach for data preparation:

1. Import the event log (csv file) provided by the BPMS.
2. Transform the file into a Data Frame.
3. Remove empty columns.
4. Ignore indexing columns.
5. Eliminate redundant columns.
6. Replace null values with appropriate values.
7. Sort the data based on the “date” column.
8. Convert the timestamp to seconds.
9. Determine the duration of each activity.
10. Determine the execution duration of each instance.
11. For each activity, calculate the time remaining at the end of the instance.
12. For each instance, determine the path followed by the task (human or automatic).
13. Add a column “path per instance.”
14. Separate each “path per instance” and distribute it into columns corresponding to the number of tasks per instance.

15. Calculate the interquartile range, first quartile, second quartile, and third quartile of the column containing instance durations.
16. Add a column named “description” containing the value “late” if the instance duration is greater than the third quartile, and “in time” otherwise.
17. Add all the previously defined columns as additional attributes to the Data Frame.
18. Convert the Data Frame into a CSV file.
19. Visualize the correlation table for all variables.
20. Decompose the dataset into independent variables X: process ID, instance ID, actor who initiated the instance, actor who completed the instance, and the path followed by the instance and a dependent variable y representing the “description” variable.
21. Encode categorical variables using LabelEncoder to transform them into numerical values.
22. Encode the variables using OneHotEncoder.
23. Split the dataset (X; y) into a training set (X_train; y_train) and a test set (X_test; y_test).
24. Standardize the values of the X_train and X_test datasets to reduce the difference between the values of different variables.

In the fourth phase (Modeling), the first step is to identify the type of problem. In our case, our objective is to predict whether a process instance is completed on time or delayed. Therefore, the prediction is a qualitative variable. Thus, we can conclude that our problem falls under the classification category in the framework of supervised learning. In our paper [34], we addressed the same topic by applying the following machine learning algorithms: KNN, Decision Tree, Random Forest, SVM, and Logistic Regression. SVM with an RBF kernel outperformed the other algorithms in terms of accuracy (84%). In this work, we create a deep learning model based on the new TensorFlow tool. We start by importing the Keras library from TensorFlow. Then, we use the sequential model to create a neural network composed of four layers: The first layer consists of 13 input neurons representing the different columns of X, the second and third layers are two hidden layers, each composed of 7 neurons, and using the relu activation function, and an output layer with a single neuron, using the sigmoid activation function for our binary classification problem. For model compilation, we use cross-entropy for error calculation, batch gradient descent with a batch size of 128 to adjust neuron weights, and accuracy for evaluation. Finally, we configure the compiler to run 300 epochs (1 epoch = propagation + Backpropagation).

In the fifth phase (Evaluation), we employ accuracy, which denotes the rate of correct classification. It quantifies how effectively instances are classified compared to the total instances.

At the end of our work, in the sixth phase (Result released), we deploy the PBMPED method for predicting the execution delays of business processes to end-users.

6 iBPMS4PED (intelligent Business Process Management System for Prediction Execution Delays)

In this section, we present iBPMS4PED (Intelligent Business Process Management System For Prediction Execution Delays), an intelligent business process management system developed using our PBMPED method. The objective of this system is to predict the execution delays of a specific business process. For our study, we chose to apply iBPMS4PED to the intelligenceWay group (<https://iway-tn.com/>), which uses an IT solution called I-Santé (<https://i-sante.tn/>). This solution offers comprehensive and tailored management for all healthcare professions, including health funds, mutuals, and insurance companies. I-Santé is built on a 100% digital platform and uses highly secure health cards. Leveraging advanced technologies such as the BPM workflow engine, ECM (Electronic Content Management), and rule engine, I-Santé meets the needs of policyholders, insurers, and healthcare professionals. In this study, we specifically focus on the “Incoming Mail” process, which is modeled on the Bonitasoft BPMS platform (see Fig. 3). Our approach aims to enhance the management of this process by predicting execution delays, which can have a significant impact on the overall efficiency of the business process management system.

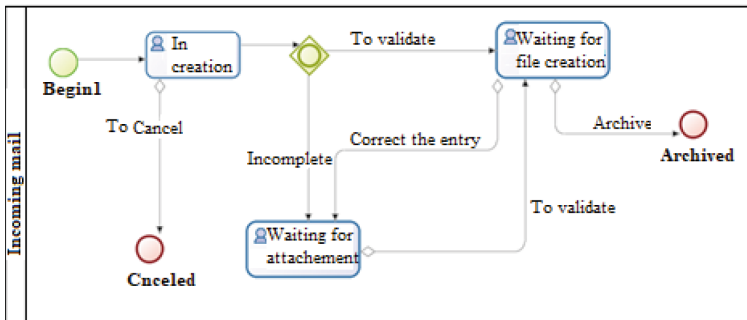


Fig. 3. Incoming mail process

In this study, we used two CSV format event log files extracted from a MySQL database configured on the I-WAY platform. These files record the execution data of a workflow engine from the BonitaSoft BPMS. The first file, “BN_PROC_INST.csv,” records the execution traces for each process instance, while the second file, “BN_ACT_INST.csv,” records the execution traces for each task. Using the Python programming language, we processed these two event logs following the steps described in the PBMPED method. For data preparation, we utilized the pandas, numpy, pylab, scipy-optimize, matplotlib, and seaborn libraries. The result of this processing is a single file named “Result.csv,” containing 4817 rows and 24 columns. The columns ready for modeling include: [PROCESS_UUID_, INST_UUID_, START_BY, END_BY, user_act0, user_act1, user_act2, ..., useractnb, duration_INST, description]. Here’s a brief explanation of these columns:

- PROCESS_UUID_: Represents the process identifier.

- INST_UUID_: Represents the instance identifier of the process.
- START_BY: Indicates the user who started the process instance.
- END_BY: Indicates the user who completed the process instance.
- user_acti: Designates the executor (person or machine) of activity number i.
- duration_INST: Gives the execution time of the process instance.
- Description: Contains a description of the observed delay in the process instance.

These data are essential for our approach to better understand and predict the execution delays of the analyzed business processes. In terms of modeling, we start by importing the Keras library from TensorFlow and selecting the sequential model. Then, we build a neural network with 13 input neurons, two hidden layers with 7 neurons each, and one output neuron. During the evaluation step, we utilize the metrics: Loss to calculate the error (see Fig. 4) and accuracy to compute the rate of correct classification (see Fig. 5). The achieved results are: accuracy = 85% and Loss = 34%.

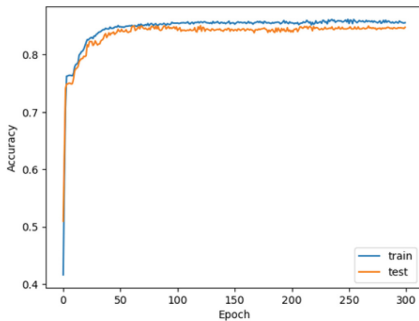


Fig. 4. Model Accuracy

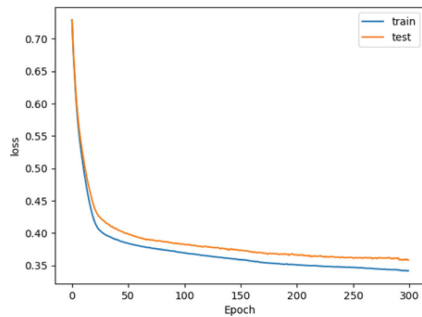


Fig. 5. Model Loss

7 Conclusion and Perspectives

In this research article, we addressed the problem of predictive monitoring of business processes, which is a current topic in the Business Process Management (BPM) domain and plays a crucial role in process-oriented organizations. To do so, we developed a method called PBMPEd, which allows predicting the execution delays of a business process. We followed the CRISP-DM method, a well-known procedure in data science, to develop a Process Mining approach. We applied Process Mining techniques to event logs that record the execution data of a BPMS workflow engine. During the data preparation phase, we performed data cleaning on the recorded event logs. Then, for each instance, we defined and added additional attributes, including the relative path, execution time, and a description of the execution time. If an instance's execution time was less than or equal to the third quartile of the execution time column, we considered it as completed on time (description = "in time"); otherwise, it was considered as completed late (description = "late"). For encoding categorical variables, we used the LabelEncoder method. Next, to reduce discrepancies between their values, we standardized all the independent variables. In the modeling phase, we used TensorFlow and Keras to create a neural network for predicting the execution delays. The goal of this prediction

was to determine whether a new process instance would be completed on time or late. In terms of accuracy, the evaluation demonstrated that this Deep Learning model outperformed the six Machine Learning models studied in our work [34], namely decision trees, random forest, SVM with rbf kernel, SVM with linear kernel, KNN, and logistic regression. For implementation, we applied the PBMPED method to create an intelligent Business Process Management system called iBPMS4PED, which enables predicting execution delays. This system was applied to an incoming mail management process in the health mutual insurance domain.

In addition to the contributions made in this article, some points deserve further investigation. In the short term, it would be feasible to explore other types of event logs. In the medium term, we plan to leverage business data stored in relational or NoSQL databases, in conjunction with the workflow engine's execution data, to detect bottlenecks. Furthermore, we intend to develop a monitoring system that allows visualizing performance indicators related to business processes.

References

1. van der Aalst, W.M.P., Schonenberg, M.H., Song, M.: Time prediction based on process mining. *Inf. Syst.* **36**(2), 450–475 (2011)
2. van der Aalst, W.M.P.: Process mining: a 360 degree overview. In: van der Aalst, W.M.P., Carmona, J. (eds.) *Process Mining Handbook. Lecture Notes in Business Information Processing*, vol. 448, pp. 3–34. Springer: Cham (2022). https://doi.org/10.1007/978-3-031-08848-3_1
3. van der Aalst, W.M.P.: Foundations of process discovery. In: van der Aalst, W.M.P., Carmona, J. (eds.) *Process Mining Handbook. Lecture Notes in Business Information Processing*, vol. 448, pp. 37–75. Springer, Cham 37-75. (2022). https://doi.org/10.1007/978-3-031-08848-3_2
4. Morey, D., Maybury, M.T., Thuraisingham, B.M.: *Knowledge Management: Classic and Contemporary Works*. MIT Press, Cambridge (2002)
5. Turki, M., Saad, I., Gargouri, F., Kassel, G.: A business process evaluation methodology for knowledge management based on multicriteria decision-making approach. In: *Information Systems for Knowledge Management*, pp. 249–277. Wiley, Hoboken (2014)
6. Sanzogni, L., Guzman, G., Busch, P.: Artificial intelligence and knowledge management: questioning the tacit dimension. *Prometheus* **35**(1), 37–56 (2017)
7. Jarrahi, M.H., Askay, D., Eshraghi, A., Smith, P.: Artificial intelligence and knowledge management: a partnership between human and AI. *Bus. Horiz.* **66**(1), 87–99 (2023)
8. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: *Fundamentals of business process management*. In: *Fundamentals of Business Process Management*, pp. 371–412. Springer, Heidelberg (2018). https://doi.org/10.1007/978-3-662-56509-4_10
9. Ko, R.K., Lee, S.S., Wah Lee, E.: Business process management (BPM) standards: a survey. *Bus. Process Manage. J.* **15**(5), 744–791 (2009)
10. 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), 1–57 (2019)
11. Marquez-Chamorro, A.E., Resinas, M., Ruiz-Cortes, A.: Predictive monitoring of business processes: a survey. *IEEE Trans. Serv. Comput.* **11**(6), 962–977 (2018)
12. Aalst, W.M.P.: Data scientist: the engineer of the future. In: Mertins, K., Bénaben, F., Poler, R., Bourrières, J.-P. (eds.) *Enterprise Interoperability VI. PIC*, vol. 7, pp. 13–26. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-04948-9_2

13. vom Brocke, J., Zelt, S., Schmiedel, T.: On the role of context in business process management. *Int. J. Inf. Manage.* **36**(3), 486–495 (2016)
14. Augusto, A., et al.: Automated discovery of process models from event logs: review and benchmark. *IEEE Trans. Knowl. Data Eng.* **31**(4), 686–705 (2019)
15. van der Aalst, W., et al.: Process mining manifesto. In: Daniel, F., Barkaoui, K., Dustdar, S. (eds.) *BPM 2011*. LNBP, vol. 99, pp. 169–194. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28108-2_19
16. Folino, F., Greco, G., Guzzo, A., Pontieri, L.: Mining usage scenarios in business processes: outlier-aware discovery and run-time prediction. *Data Knowl. Eng.* **70**(12), 1005–1029 (2011)
17. Folino, F., Guarascio, M., Pontieri, L.: Discovering context-aware models for predicting business process performances. In: Meersman, R., et al. (eds.) *On the Move to Meaningful Internet Systems: OTM 2012*, vol. 7565, pp. 287–304. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33606-5_18
18. Polato, M., Sperduti, A., Burattin, A., de Leoni, M.: Data-aware remaining time prediction of business process instances. In: 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, pp. 816–823 (2014)
19. Ceci, M., Lanotte, P.F., Fumarola, F., Cavallo, D.P., Malerba, D.: Completion time and next activity prediction of processes using sequential pattern mining. In: Džeroski, S., Panov, P., Kocev, D., Todorovski, L. (eds.) *DS 2014*. LNCS (LNAI), vol. 8777, pp. 49–61. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11812-3_5
20. de Leoni, M., van der Aalst, W.M.P., Dees, M.: A General framework for correlating business process characteristics. In: Sadiq, S., Soffer, P., Völzer, H. (eds.) *BPM 2014*. LNCS, vol. 8659, pp. 250–266. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10172-9_16
21. Lakshmanan, G.T., Shamsi, D., Doganata, Y.N., Unuvar, M., Khalaf, R.: A Markov prediction model for data-driven semi-structured business processes. *Knowl. Inf. Syst.* **42**(1), 97–126 (2015)
22. Tax, N., Verenich, I., La Rosa, M., Dumas, M.: Predictive business process monitoring with LSTM neural networks. In: Dubois, E., Pohl, K. (eds.) *CAiSE 2017*. LNCS, vol. 10253, pp. 477–492. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-59536-8_30
23. Ghattas, J., Soffer, P., Peleg, M.: Improving business process decision making based on past experience. *Decis. Support Syst.* **59**, 93–107 (2014)
24. Senderovich, A., Weidlich, M., Gal, A., Mandelbaum, A.: Queue mining for delay prediction in multi-class service processes. *Inf. Syst.* **53**, 278–295 (2015)
25. Rogge-Solti, A., Weske, M.: Prediction of business process durations using non-Markovian stochastic Petri nets. *Inf. Syst.* **54**, 1–14 (2015)
26. Maggi, F.M., Di Francescomarino, C., Dumas, M., Ghidini, C.: Predictive monitoring of business processes (2013)
27. Leontjeva, A., Conforti, R., Di Francescomarino, C., Dumas, M., Maggi, F.M.: Complex symbolic sequence encodings for predictive monitoring of business processes. In: Motahari-Nezhad, H.R., Recker, J., Weidlich, M. (eds.) *BPM 2015*. LNCS, vol. 9253, pp. 297–313. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23063-4_21
28. Di Francescomarino, C., Ghidini, C., Maggi, F.M., Milani, F.: Predictive process monitoring methods: which one suits me best? In: Weske, M., Montali, M., Weber, I., vom Brocke, J. (eds.) *BPM 2018*. LNCS, vol. 11080, pp. 462–479. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98648-7_27
29. Teinemaa, I., Dumas, M., Maggi, F.M., Di Francescomarino, C.: Predictive business process monitoring with structured and unstructured data. In: La Rosa, M., Loos, P., Pastor, O. (eds.) *BPM 2016*. LNCS, vol. 9850, pp. 401–417. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45348-4_23

30. Evermann, J., Rehse, J.-R., Fettke, P.: A deep learning approach for predicting process behaviour at runtime. In: Dumas, M., Fantinato, M. (eds.) BPM 2016. LNBP, vol. 281, pp. 327–338. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58457-7_24
31. Kratsch, W., Manderscheid, J., Röglinger, M., Seyfried, J.: Machine learning in business process monitoring: a comparison of deep learning and classical approaches used for outcome prediction. *Bus. Inf. Syst. Eng.* **63**(3), 261–276 (2021)
32. Schröer, C., Kruse, F., Gómez, J.M.: A systematic literature review on applying CRISP-DM process model. *Procedia Comput. Sci.* **181**, 526–534 (2021)
33. Zhang, Y.: Sales forecasting of promotion activities based on the cross-industry standard process for data mining of E-commerce promotional information and support vector regression. *32*(1), 212–225 (2021)
34. Ben Fradj, W., Turki, M.: Prediction of business process execution time. In: Abraham, A., Pillana, S., Casalino, G., Ma, K., Bajaj, A. (eds.) ISDA 2022. LNCS, vol. 715, pp. 105–114. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-35507-3_11