| OUALI YASMINE   M2 MCI EUROPE |
|---|

| | |
|---|---|
| ***My level 1*** | SokSabaye |
| ***Source:*** | google scholar / IEEE<br>[https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9203823](https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9203823) |
| ***Paper title*** | A Graph-Based Approach to Interpreting Recurrent Neural Networks in Process Mining |
| ***Keywords specific to the paper*** | Directly-follows graph, explainable AI, long short term memory, process mining, recurrent<br>neural network |

# Summary

### *PROCESS MINING*

Process mining focuses on autonomously deriving process models from event logs in order to examine implementation of processes in reality. Process mining is primarily concerned with extracting knowledge and insights from transaction records. This allows for the examination of business processes from different viewpoints including control-flow, organization, and performance. The data obtained from process mining can be utilized to monitor and analyze an organization's performance, detect impediments, anticipate run durations, find deviations in control flow, uncover roles and relationships, and optimize resource use.

Event logs are the cornerstone for all process mining approaches. They contain cases, which are either traces or sequences of events. Moreover, event logs may contain additional columns like transaction type and cost. The most common process mining tasks include three processes mining task:

1) *process discovery* : it entails automatically generating a process model from an event log in order to accurately depict the actual business process.
2) *conformance checking* : it ensures that business process participants adhere to the process by comparing a discovered model to actual logs, and vice versa.

3) *process model enhancement*: focuses on improving or expanding the previously established process model using data from the real process recorded in an event log.

## DEEP LEARNING AND RECURRENT NEURAL NETWORKS

Deep learning has enabled significant advancements in a wide range of tasks such as image recognition, speech recognition, anomaly detection, disease diagnosis, and natural language processing. Neural networks, which are the foundation of deep learning, consist of multiple layers of interconnected neurons that carry out nonlinear transformations of data to allow the network to learn patterns in the data.

Recurrent neural networks (RNNs) are widely recognized as one of the most popular deep learning (DL) architectures due to their capability to learn and generalize over sequences of inputs rather than individual patterns. RNNs can assess patterns over time, recognize short and long term dependencies, and distinguish temporal differences.With recurrent connections and hidden states distributed over time, RNNs effectively retain past information. In RNN, the current time step's output is used as the input for the next step.

An RNN model considers both current input and prior points to learn temporal dependence and contextual information in input data.The network's ability to retain information allows it to understand long-term dependencies within a sequence, preventing the loss of prior knowledge. However, the capacity of a standard RNN model to capture contextual information is restricted, potentially impacting the model's effectiveness. This is related to the problem of vanishing or exploding gradients encountered during RNN model training via back propagation.

LSTM (Long Short-Term Memory), a variant of the RNN architecture, was proposed to address the issue of vanishing or exploding gradients. While the control-flow of LSTM remains the same as that of RNN, the key difference lies in the operations within LSTM cells.

Bidirectional Long Short-Term Memory (BLSTM), divides the conventional LSTM's state neurons into two, the forward state (responsible for positive time direction) and the backward state (responsible for negative time direction), Both states are connected to a shared output layer. This unique architecture enables the training of a model using both past and future information within a specific time frame.

## PROCESS DISCOVERY VERSUS PROCESS PREDICTION METHODS

Process discovery is the most commonly studied sort of process mining. Process discovery is based on data acquired from an information system over time, or real-time data output by running processes, which is commonly referred to as event logs. The main objective of process discovery is to enhance the understanding of control-flow relationships between tasks as observed in event logs. By utilizing process discovery methods, we can gain insights into how a process should be executed or is currently being executed. Moreover, these discovered process models serve as a basis for further analysis. Transition diagrams, such as Petri nets, BPMN diagrams, causal nets, state machines, and directed acyclic networks, are frequently employed to depict the results of process discovery methods.

DL-based sequence modeling techniques like LSTM are often seen as an alternative to process discovery methods. These techniques have become increasingly popular in the field of process mining. Sequence prediction involves forecasting future events based on past events and can be applied to both ongoing and completed cases in an event log. Studies have shown that DL models can generate more accurate next event predictions compared to process models derived from event logs. However, the decision-making process behind these predictions remains unclear when using DL models, unlike process discovery methods that provide visually explainable process model graphs illustrating the sequence of events and event instances running in parallel.