Extended Abstract: Simulation of unit journeys using process crowding in Generative Adversarial Networks

Yoann Valero[1,2], Frédéric Bertrand[1], and Myriam Maumy[1]

[1] University of Technology of Troyes, 12 rue Marie Curie CS 42060, 10004 Troyes CEDEX, France
[2] QAD, 100 Innovation Pl, Santa Barbara, CA, USA
{firstname.lastname}@utt.fr

Abstract. Neural networks, particularly Generative Adversarial Networks (GAN), have proven to be efficient for predicting the sequence of events of ongoing units in processes. A next step would be to let users choose the initial state of a process, and let the model generate the typical journey that would unfold as a result. We propose a way of doing so in the context of deep learning by taking what we call *process crowding* into account. By training a GAN using crowdings calculated in an event log (or simply log), we get a model which can receive any date and crowdings as inputs, and output a journey according to these initial conditions. This method is a simple form of simulation at the scale of units. We propose an efficient formula for process crowding and its variable selection, as well as a means to evaluate such simulations through the sampling of observed crowdings with the cube sampling method.

Keywords: Process · Simulation · Generative Adversarial Networks · Crowding · Deep Learning.

1 Introduction

Deep learning models such as LSTMs used in [2] and [5] and Transformers in [1] currently aim at predicting the next event for ongoing units, others such as [6] perform well for the remaining sequence of events. However, these models predict each unit without taking their surroundings into account. They also cannot generate an output unless they are fed an ongoing unit. This is a core difference compared to classical approaches such as Petri nets, which can generate events from an initial marking. We propose to bridge this gap in the field of deep learning.

2 Methodology

Crowding [7] counts the number of units in each activity of a process at a given time. We train a GAN model based on [6] with a Wasserstein loss [3] by duplicating the first event of complete units, replacing their first activity with a *Start of*

State, and concatenating to it crowdings calculated at the corresponding times. We train the model to output a full journey by feeding it this modified first event. It is then able to generate a journey that would result from user-chosen crowdings. We evaluate these outputs by sampling observed crowdings and measuring both diversity and accuracy of outputs in *Traffic fines* and *Helpdesk* logs.

3 Results

Table 1 shows that our method allows diverse journeys to be output depending on starting crowdings, with good accuracy on activity sequences [4] and completion times (MAE).

Table 1: Simulation diversity and accuracy

Event log	Number of unique simulated journeys / Number of unique real journeys	DL similarity (activities)	MAE (cycle times)
Traffic fines	24/44	0.65	243.84 days
Helpdesk	37/226	0.88	8.51 days

4 Conclusion

Our method takes a process' population at a given time, all given by a user, and outputs the journey that would most likely result from these given conditions with good accuracy and some diversity in its outputs, thus allowing for a simple type of process simulation. Unbalanced data needs to be addressed in order to allow more unlikely scenarios to be output, while conforming to the underlying process.

References

1. Bukhsh, Z.A., Saeed, A., Dijkman, R.M.: Processtransformer: Predictive business process monitoring with transformer network (2021)
2. Evermann, J., Rehse, J.R., Fettke, P.: Predicting process behaviour using deep learning. Decision Support Systems **100**, 129–140 (8 2017). https://doi.org/10.1016/j.dss.2017.04.003
3. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of wasserstein gans (2017)
4. Oommen, B., Loke, R.: Pattern recognition of strings with substitutions, insertions, deletions and generalized transpositions. Pattern Recognition **30**(5), 789–800 (1997). https://doi.org/10.1016/S0031-3203(96)00101-X
5. Tax, N., Verenich, I., La Rosa, M., Dumas, M.: Predictive business process monitoring with lstm neural networks. Lecture Notes in Computer Science p. 477–492 (2017), `http://dx.doi.org/10.1007/978-3-319-59536-8\_30`
6. Taymouri, F., Rosa, M.L.: Encoder-decoder generative adversarial nets for suffix generation and remaining time prediction of business process models (2020)
7. Valero, Y., Bertrand, F., Maumy, M.: Use of process crowding in conditional wgan for remaining process events prediction. In: Joint Statistical Meetings 2022 Proceedings (2022)