On the Performance Analysis of the Adversarial System Variant Approximation Method to Quantify Process Model Generalization

Julian Theis, Ilia Mokhtarian, and Houshang Darabi(✉)

Department of Mechanical and Industrial Engineering, University of Illinois at Chicago, 842 West Taylor Street, Chicago, IL 60607, USA
{jtheis3,imokht2,hdarabi}@uic.edu

Abstract. Process mining algorithms discover a process model from an event log. The resulting process model is supposed to describe all possible event sequences of the underlying system. Generalization is a process model quality dimension of interest. A generalization metric should quantify the extent to which a process model represents the observed event sequences contained in the event log and the unobserved event sequences of the system. Most of the available metrics in the literature cannot properly quantify the generalization of a process model. A recently published method called Adversarial System Variant Approximation leverages Generative Adversarial Networks to approximate the underlying event sequence distribution of a system from an event log. While this method demonstrated performance gains over existing methods in measuring the generalization of process models, its experimental evaluations have been performed under ideal conditions. This paper experimentally investigates the performance of Adversarial System Variant Approximation under non-ideal conditions such as biased and limited event logs. Moreover, experiments are performed to investigate the originally proposed sampling parameter value of the method on its performance to measure the generalization. The results confirm the need to raise awareness about the working conditions of the Adversarial System Variant Approximation method and serve to initiate future research directions.

Keywords: Process Mining · Conformance Checking · Generalization · Generative Adversarial Networks

1 Introduction

Significant research effort has been spent on the automated discovery of process models from event logs and the quality assessment of such models, i.e., *conformance checking*. While the focus of conformance checking has been mainly on measuring how well a discovered process model reflects event sequences that are recorded in an event log, measuring the extent to which a process model

J. Munoz-Gama and X. Lu (Eds.): ICPM 2021 Workshops, LNBIP 433, pp. 281–293, 2022.
https://doi.org/10.1007/978-3-030-98581-3_21

generalizes the possible event sequences of the system from which the event log originates, is less explored. The origin of such event logs is usually real-world systems in domains such as business [2], manufacturing [19,23], or healthcare [9,11,21]. Studies have shown that measuring the generalization of discovered process models is of importance [16] and that only a few methods focus on this objective. Meanwhile, the research community is aware that existing methods do not fully address requirements and present individual shortcomings [12,18].

Adversarial System Variant Approximation (AVATAR) is a method [20] to overcome some of the known issues in measuring generalization. This method leverages a Generative Adversarial Network (GAN) that is trained on the same event log that is used to discover a process model. AVATAR is based on the fact that GANs successfully demonstrated the ability to unveil underlying data distributions, including discrete sequences, and transfers the approach to the context of measuring the generalization of process models. By sampling from the GAN, a baseline of supposedly generalizing event sequences is obtained. Experimental evaluations have been performed using ground truth systems which have shown that the GAN of AVATAR can model observed event sequences of the event log, and unobserved event sequences of the ground truth system accurately.

Whereas the experimental evaluation of AVATAR demonstrated that GANs are suitable and promising neural network architectures that can be used to measure the generalization of a process model, further research is required to understand the working conditions of those GANs in depth. This paper contributes to this objective by conducting performance analyses on the GANs of AVATAR using the same ground truth systems that were used in the original publication. First, the performance analysis includes an experimental evaluation of the proposed sampling parameter k value of $10,000$ of the AVATAR GAN. Second, experiments are performed on limited event log sizes. The original publication used a constant 70% split ratio of the event sequences of the ground truth systems that were used as the event log for process discovery and AVATAR. Under real-world conditions, such a constant 70% split ratio is usually infeasible. Hence, it is necessary to investigate the GAN performance of AVATAR using different split ratios. Third, an experimental evaluation is performed on the robustness of AVATAR towards bias. Specifically, this paper investigates if event logs that are biased affect the ability of the GAN to unveil unobserved event sequences of the ground truth system. The results of the experiments are used to draw conclusions and to raise awareness about the working conditions of the GANs of AVATAR. The results and source codes are available on Github[1].

2 Related Work

2.1 Generalization Metric

Generalization describes that a process model, such as a Petri net (PN), models ideally all possible event sequences of a system that can realistically occur.

[1] https://github.com/ProminentLab/AVATAR.

This means that a process model should allow for the event sequences that are recorded in an event log when observing a system under investigation. These event sequences are usually used to automatically discover a process model using a process discovery algorithm. Additionally, the process model should not allow for unrealistic event sequences beyond the observed ones. It is obvious that the difficulty of measuring the generalization of a process model reduces to classifying if given unobserved event sequences are either realistic or unrealistic in the context of the system under investigation.

A significant amount of research has been spent on measuring how well a process model allows for event sequences contained in an event log (i.e., measuring the *fitness*) and how well a process model restricts to allow for event sequences beyond the ones contained in an event log (i.e., measuring the *precision*). However, research on measuring the generalization of process models is scarce due to the difficulty of deriving realistic and unobserved event sequences from an event log. Nonetheless, the process mining research community is aware that the quality dimension of generalization is of importance [7,12,18].

Historically, one of the first approaches to quantify the extent to which a process model generalizes event sequences beyond the ones contained in an event log has been introduced by Buijs et al. [8]. The proposed approach is based on quantifying the trustworthiness of the precision of a process model using alignments. Highly frequent used areas of a process model are considered well generalizing whereas low frequent parts of the model are less generalizing.

Van der Aalst et al. [1] built a measurement to quantify that a process model does not overfit on a given event log. Specifically, their approach is based on the probability of observing a new event in any given state of the model based on the observations contained in the event log. If the likelihood of observing a new event in a given state is small, then the generalization is good.

Vanden Broucke et al. [6] introduced a method to measure the generalization of a process model using weighted artificial negative events. In comparison to an actual event, an artificial negative event prevents the occurrence of a specific event at a given time. This concept enables to derive allowed and disallowed generalized event sequences.

A method proposed by van Dongen et al. [10] is based on anti-alignments which are event sequences that are disparate from a set of given event sequences. This notion is used to measure the generalization by relating the state space of a process model. A generalizing process model has therefore a maximally different set of anti-alignments without introducing unseen states.

A comparative study by Janssenswillen et al. [13] led to the conclusion that metrics that quantify the generalization with respect to a given event log do usually not assess the quality of a process model concerning the underlying system correctly. Hence, generalization metrics need to be developed that do not solely relate modeled event sequences to the ones contained in an event log. Such metrics should be evaluated using ground truth systems.

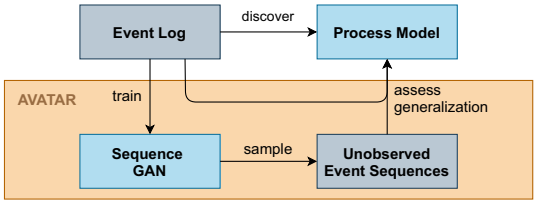

Fig. 1. Flow chat of the AVATAR methodology, derived from [20]

2.2 Adversarial System Variant Approximation

AVATAR is a recently proposed approach to quantify the extent to which a process model generalizes [20]. The idea of this method is to unveil realistic but unobserved event sequences of a system using Generative Adversarial Networks (GANs). If it is possible to confidently model unobserved event sequences using GANs, then measuring the generalization reduces to measuring the fitness and precision of a process model using the observed event log in combination with the unobserved event sequences that are modeled by the GAN. This is motivated by the generalization capabilities of GANs [3]. A flow chart of the methodology is provided in Fig. 1. A given set of event sequences that is used for automated process discovery is also used to train a Sequence GAN (SGAN). AVATAR leverages a RelGAN [15] architecture that is enhanced with an additional standard discriminator neural network. A major hyperparameter of this SGAN architecture is the temperature control β of the RelGAN that controls the tradeoff between sample diversity and quality. The trained SGAN is then used to sample unobserved event sequences. AVATAR proposes therefore two sampling methodologies. The first is *naive sampling* controlled by the parameter k which means that k samples are drawn from the generator of the SGAN. The intuition is that the number of unique event sequences converges with an increasing number of sampling iterations. This also means that the relative frequency of an event sequence indicates the modeling confidence of this particular event sequence. The second sampling methodology uses the *Metropolis-Hastings algorithm* [14] and is inspired by the work of Turner et al. [22]. It is assumed that by sampling from the SGAN, the unobserved event sequences of a system can be unveiled. Here, quantifying the generalization of a process model reduces to measuring the fitness and precision of the process model with respect to the set of observed and approximated unobserved event sequences from the GAN.

The AVATAR methodology has been statistically evaluated using the finite set of event sequences of 15 ground truth PNs. These PNs were created artificially as part of a comparative study of process discovery quality measures [13] and are publicly available[2]. Each of the 15 PNs has different, but realistic characteristics. 10 of the PNs can be classified as *moderately complex* with a small number of transitions and comparatively few parallelisms whereas 5 PNs

[2] https://github.com/gertjanssenswillen/processquality/.

are *highly complex* with a larger number of transitions and parallel structures. The *highly complex* PNs are supposed to reflect the complexity of real-world systems. For each ground truth PN, a random and unbiased 70% random split of the modeled unique event sequences was considered as an event log. These event logs were used to discover process models using two process discovery algorithms [4,5]. The remaining 30% were withheld as the set of unobserved event sequences that the GAN should be able to model.

The results of the experimental evaluation showed that SGANs are well suited to obtain realistic unobserved event sequences with a relatively small number of unrealistic event sequences. Moreover, the AVATAR generalization scores were compared to existing generalization metrics on the discovered process models. The obtained AVATAR scores on those models were perceived more appropriate than the scores of existing generalization measures based on the ground truth event sequence information. All experimental results were obtained under ideal working conditions.

3 Notations

The notations that are used throughout this paper are based on and consistent with the ones of the original AVATAR publication. The reader is referred to [20] for comprehensive introductions.

A system is denoted by S. An event $a \in \mathcal{A}$ describes an instantaneous change of the state of S where $\mathcal{A}$ is the finite set of all possible events. The cardinality of a set is denoted by $|\cdot|$. An event instance E is a vector and describes the occurrence of a specific a along with its occurrence timestamp and optional additional information. A trace is a finite and chronologically ordered sequence of event instances. A *variant* $v \in \mathcal{V}$ is a sequence of events where $\mathcal{V}$ is the infinite set of all variants. A trace maps to exactly one variant. Whereas an event log is a set of traces, denoted by $\mathcal{L}$, a variant log is a sample of variants denoted by $\mathcal{L}^*$. A *unique variant log* is denoted by $\mathcal{L}^+$ and equals to the set of $\mathcal{L}^*$. The set of all variants that can be observed during the runtime of S is denoted by $\mathcal{V}_S$. The functions $\mu(\mathcal{V})$ and $mean(\mathcal{V})$ return the maximum and mean variant lengths of a given set of variants, respectively.

Following the AVATAR methodology, a SGAN architecture is trained on $\mathcal{L}^+$ with a hyperparameter β, i.e., GAN_β. The SGAN can be used to naively sample variants. The number of sampling iterations from GAN_β is denoted by k.

When training a GAN, all variants of $\mathcal{L}^+ \subseteq \mathcal{V}_S$ are considered. A subset of variants $\mathcal{V}_u$ might exist such that $\mathcal{V}_S = (\mathcal{L}^+ \cup \mathcal{V}_u)$ and $(\mathcal{L}^+ \cap \mathcal{V}_u) = \emptyset$. $\mathcal{V}_u$ is intuitively the set of unobserved behavior. Ideally, when sampling k times from GAN_β, it is desired to obtain an estimated set of system variants, i.e., $\hat{\mathcal{V}}_S$ that equals to $\mathcal{V}_S$. How well the GAN performs to reach this goal is quantified using the true positive ratios $tp = \frac{|\hat{\mathcal{V}}_S \cap \mathcal{V}_S|}{|\mathcal{V}_S|}$ and $tp_u = \frac{|\hat{\mathcal{V}}_S \cap \mathcal{V}_u|}{|\mathcal{V}_u|}$. tp describes the proportion of realistic variants sampled using GAN_β over all possible system variants. tp_u describes the ratio of sampled variants using GAN_β over all unobserved variants. Moreover, the number of unique sampled variants is recorded. Ideally,

tp and tp_u should be equal to 1 while the number of unique sampled variants should equal $|\mathcal{V}_S|$. The score function $s(tp, tp_u) = \frac{tp+tp_u}{\sqrt{2}}$ is used, as proposed and reasoned in [20], to quantify how well the GAN of AVATAR performs.

4 Problem Statement

The AVATAR methodology [20] demonstrated that SGANs can model $\mathcal{V}_u$ which builds a foundation to measure the generalization of process models. The evaluation setup of AVATAR consisted of a 70/30 split ratio of $\mathcal{V}_S$ to obtain $\mathcal{L}^+$ and $\mathcal{V}_u$ and a sampling parameter k value $10,000$ for each of the ground truth systems. This setup raises multiple research questions, including the following.

RQ1: *Is the parameter k with a value of $10,000$ optimally defined and is there a relationship between k and the GAN performance of AVATAR?* The parameter k describes the number of variants that are drawn naively from the trained SGAN without leveraging the Metropolis-Hastings algorithm. Whereas [20] states that preliminary results showed that setting k to the value of $10,000$ is a good choice, a proven justification for this value is missing. Moreover, it remains unclear if a relationship between k and the performance of the GAN of AVATAR exists. This paper experimentally assesses the performance of the GANs with multiple values for k to validate the statement made in the original publication and investigates the relationship between S, k, and the GAN to model $\mathcal{V}_S$.

RQ2: *How does the size of $\mathcal{L}^+$ relate to the performance of modeling $\mathcal{V}_S$?* The AVATAR methodology has been evaluated using a 70/30 split ratio of $\mathcal{V}_S$ to obtain $\mathcal{L}^+$ and $\mathcal{V}_u$ across all used ground truth systems. However, it remains unclear how the GAN of AVATAR performs if less information of a system is given. In real-world scenarios, an exact 70% split of all possible variants of a system is usually unrealistic. The ratio of variants contained in $\mathcal{L}^+$ to all variants in $\mathcal{V}_S$ can be guessed at its best. Hence, this paper experimentally assesses the performance of the GANs of AVATAR at different split ratios to investigate the working conditions of AVATAR when the given event log size is limited.

RQ3: *Are the GANs of AVATAR sensitive to biased variant logs?* The GANs of AVATAR have been evaluated using a random and unbiased split of $\mathcal{V}_S$. In real-world scenarios though, $\mathcal{L}^+$ might be biased due to a limited observation duration of the system or adverse environmental situations. Whereas research has been conducted on the impact of biased event logs on process discovery algorithms [17], it remains unclear how the GANs of AVATAR perform when being trained on a biased set of variants. Bias can be expressed, e.g., in terms of variant lengths. In this paper, preliminary experiments are performed to investigate if the performance of the GANs are affected when being trained on specific biased variant logs.

5 Experimental Setup

5.1 Sampling Parameter

To investigate the relationship between k and the performance of the SGANs (*RQ1*), multiple values for k are investigated. Specifically, k is set to $1,000$, $2,000$, $4,000$, $6,000$, up to $20,000$, with an increment of $2,000$ each. This includes the originally proposed $k = 10,000$ value. These specific values are chosen such that performance changes can be observed when increasing and decreasing the proposed value of k. It is expected that the performance of the GANs decreases with a very small value, such as $k = 1,000$, but it remains unclear if the performance increases with an increased value of k. It is not expected that a granularity finer than $2,000$ will unveil significant differences.

Training and sampling of the SGANs is performed on the five highly complex PNs that were also used to evaluate the AVATAR methodology according to the original publication. These systems are denoted as S_{11-15} and correspond to Systems 11–15 in [20]. For each of the five systems, two SGANs are trained with $\beta = 100$ and $\beta = 1000$, respectively. These GANs are trained using a random 70% split of $\mathcal{V}_S$ which corresponds to $\mathcal{L}^+$. The remaining 30% results in $\mathcal{V}_u$ and are used to evaluate the performance of the SGAN to approximate the unobserved system variants, as in the original publication. This is called a *70/30 split ratio*. The setup results in ten different SGAN models and, due to 11 different values for k, in a total of 110 observation values for evaluation.

5.2 Variant Log Size

To investigate the performance of the GANs of AVATAR when limited variant log sizes are given (*RQ2*), two SGANs per system are trained with different split ratios compared to the 70/30 ratio of the original evaluation. In this setup, the 70/30 split ratio is used as a baseline for comparison. Moreover, experiments are performed using 10/90, 20/80, 30/70, 40/60, 50/50, and 60/40 split ratios. It is expected that the performance of the GANs in modeling $\mathcal{V}_u$ decreases with smaller $|\mathcal{L}^+|$ values. As before, the systems S_{11-15} are used for experimental evaluation due to their realistic complexity. The SGANs are trained with $\beta = 100$ and $\beta = 1000$ to be consistent with the original AVATAR work. This results in 70 SGANs for evaluation. Variants are generated from the SGANs using the originally proposed $k = 10,000$ value.

5.3 Biased Variant Logs

This experiment investigates the performance of the GANs of AVATAR in detecting $\mathcal{V}_u$ when being trained on a biased $\mathcal{L}^+$ to provide an answer to *RQ3*. Bias is expressed using the length of variants. The baseline is obtained using a random and unbiased 70/30 split ratio on $\mathcal{V}_S$ such that $mean(\mathcal{L}^+)$ and $mean(\mathcal{V}_u)$ are almost equal. Four bias setups are defined and denoted by $b1$ to $b4$.

The first bias setup $b1$ is defined such that $\mathcal{L}^+$ contains the shortest 70% of $\mathcal{V}_s$ and $\mathcal{V}_u$ contains the remaining 30%. This means that a SGAN is trained on short variants, but is supposed to generalize to long variants. The setup $b2$ is defined such that $\mathcal{L}^+$ contains the longest 70% of $\mathcal{V}_s$ and $\mathcal{V}_u$ contains the remaining variants. In this case, a SGAN is trained on long variants and is supposed to generalize short variants. The setups $b3$ and $b4$ are leaky variations of $b1$ and $b2$, respectively. For both setups, 20% of the variants in $\mathcal{V}_u$ are randomly exchanged with a randomly chosen variant from $\mathcal{L}^+$. This means that the corresponding SGAN is not trained on strictly short or strictly long variants. However, bias in terms of the lengths of variants contained in $\mathcal{L}^+$ and $\mathcal{V}_u$ persists.

For all setups $b1$ to $b4$, the longest possible variant of a corresponding system is contained in $\mathcal{L}^+$ rather than $\mathcal{V}_u$. This is required to satisfy the assumption that the maximum possible system variant length is known to train an SGAN [20]. Therefore, at least one variant with a length equal to $\mu(\mathcal{V}_S)$ must be known. Like before, two SGANs are trained with $\beta = 100$ and $\beta = 1000$, respectively, for each of the systems S_{11-15} and each setup plus the baseline setup. Consequently, the total number of SGAN models under investigation equals 50.

6 Results

6.1 Sampling Parameter Results

For S_{11} and GAN_{100}, the number of approximated system variants increases with the value of k. This GAN setup is closest to the desired $|\mathcal{V}_S|$ value when using $k = 8,000$. In the meantime, the tp ratio decreases with an increasing value of k. With an increasing value of k, the tp_u ratio converges to 0.6. Similar behavior is observed for the SGANs for S_{12}. However, with $k = 2,000$, $\hat{\mathcal{V}}_S$ already exceeds the desired value of $\mathcal{V}_S$. Accordingly, tp decreases and tp_u converges with increasing k to about 0.8. The overestimation of variants can be explained by the complexity of the underlying system. The second most complex system is S_{14} with a much smaller maximum variant length. Accordingly, the SGANs of S_{14} are better in approximating $\mathcal{V}_S$ compared to the ones of S_{12}. Systems S_{13-15} perform similarly to S_{11} with an optimal variant number approximation around $k = 10000$. The tp_u ratios seem to converge around 0.7 and 0.9.

The results look similar for GANs with $\beta = 1,000$. In general, $\hat{\mathcal{V}}_S$ is overestimated with an increasing value of k and when $k > 10,000$. Only for S_{11}, the corresponding SGAN underestimates $|\mathcal{V}_S|$ when using any of the considered values for k. However, for $k = 20,000$, GAN_{1000} almost perfectly estimates $|\mathcal{V}_S|$ with a decently high tp and tp_u ratio. Generally, the tp ratio reduces with a more gentle slope compared to GAN_{100} while tp_u converges to a fixed value similar to GAN_{100}. The tp_u convergence value lies between 0.75 and 1.0.

Since it can be observed that the performance of the GANs on more complex systems, such as S_{12}, can be weaker, a linear regression model is fit using the features k, $\mu(\mathcal{V}_S)$, and $|\mathcal{V}_S|$ to model the resulting scoring value for s. With linear features, this leads to an R^2 value of 1.4% indicating a bad fit. With the corresponding quadratic features, the R^2 score improves to 40%. The quadratic

relationship could be an initial step to develop a rule-of-thumb to select an individual and optimized value for k. The required values for $|\mathcal{V}_S|$, $\mu(\mathcal{V}_S)$ and a desired minimum score value s can be guessed by using expert knowledge.

The median value of k that corresponds to the best obtained scores for the SGAN models under consideration, equals $10,000$. This validates the general suitability of $k = 10,000$ as proposed in [20] and answers *RQ1*.

6.2 Variant Log Size Results

For the GANs that were trained using $\beta = 100$, it can be generally noted that fewer unique variants are sampled with decreasing sizes of $\mathcal{L}^+$. At the same time, it can also be observed that tp and tp_u generally tend to decrease. A similar, but less significant behavior can be observed for the SGANs that are trained using $\beta = 1000$. This confirms the expectations.

The same trend can be observed when visualizing the 90% confidence intervals (CIs) of the obtained scores s for each SGAN and variant log size setup over all systems, as visualized in Fig. 2. Whereas this visualization cannot provide statistical proof due to the small sample size, it shows the decreasing trend satisfyingly. Since the CIs for a 10/90 split ratio and the baseline 70/30 split ratio for both SGAN setups are non-overlapping, it can be concluded that a 10/90 split ratio performs statistically poorer than a 70/30 split ratio with 90% confidence.

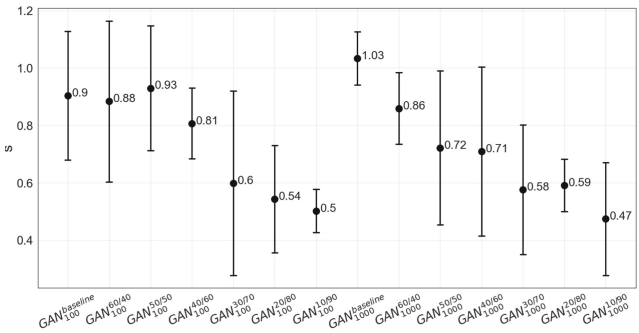

Fig. 2. 90% CIs of the mean scores s for each SGAN setup of different $\mathcal{L}^+$ sizes over all systems S_{11-15}

To provide an answer to *RQ2*, the GAN performance decreases with less variants contained in $\mathcal{L}^+$ with respect to $|\mathcal{V}_S|$. These experiments prove that the SGANs of AVATAR trained with a 70/30 split ratio perform statistically significantly better compared to a 10/90 split ratio. For GAN_{1000}, the experiments show that a 70/30 split ratio leads to statistically significantly better performance compared to 30/70, 20/80, and 10/90 split ratios. Further experiments with a larger sample size are required to provide statistical proof.

6.3 Biased Variant Log Results

For all systems, the SGAN using $\beta = 100$ on the biased setup $b1$ performs poorly. However, when training using $\beta = 1000$, the performance seems to be increasing. The β parameter indicates an impact on the performance when $\mathcal{L}^+$ is biased. However, the details of the impact remain unclear. Overall, the SGANs trained with $\beta = 1000$ seem to perform better in general.

Furthermore, the performance seems to increase when $\mathcal{L}^+$ is less restrictively biased, i.e., with the setups $b3$ and $b4$ compared to $b1$ and $b2$, respectively. This indicates that less bias leads to better performance. Additionally, $b2$ seems to perform better than $b1$, and $b4$ performs better than $b3$. The same can be observed when visualizing the 90% CIs of the scores s per SGAN setup over all systems in Fig. 3. Comments on the statistical significance of each CI cannot be made due to the small sample size. However, the CI mean values indicate the observed trend. The baseline SGANs are the best-performing models. When introducing leaky bias with $b3$ and $b4$, the performance reduces on average. Strict bias, such as with setups $b1$ and $b2$, leads to a further decrease of performance in unveiling $\mathcal{V}_S$. The large CIs for the SGANs trained using $\beta = 1000$ and using the setups $b3$ and $b4$ can be either a randomness artifact or a sign that the β hyperparameter can accommodate for non-strict bias in specific situations.

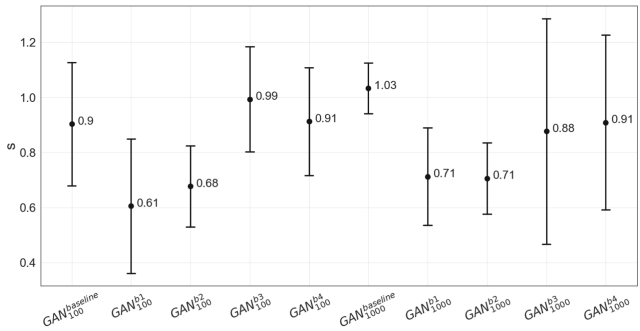

Fig. 3. 90% CIs of the mean scores s for each biased $\mathcal{L}^+$ and baseline SGAN setup over all systems S_{11-15}

To answer *RQ3*, the GANs are sensitive to bias and perform with a s value that decreases proportionally to the significance of present variant length bias in $\mathcal{L}^+$. Further experiments with larger sample sizes of ground truth systems are anticipated to provide statistical evidence and insights on the potential impact of β to accommodate for bias.

7 Conclusion

Regarding *RQ1*, the experiments have shown that $k = 10,000$ is generally a good choice. However, an individual value for k is required depending on the underlying system complexity to fine-tune the GAN performance. Linear regression

with quadratic features indicated a good fit to estimate an optimized value for k given the desired performance score s, the total number of system variants, and the maximum variant length of the underlying system. For *RQ2*, the GAN performance in modeling $\mathcal{V}_S$ generally tends to decrease when fewer variants of the system are contained in $\mathcal{L}^+$. Finally, the GANs of AVATAR seem to be sensitive towards biased variant logs, as an answer to *RQ3*. The performance of the underlying SGANs decreases the more significant the bias in $\mathcal{L}^+$ is. Moreover, the experimental results show the potential that the SGAN hyperparameter β might be able to accommodate for bias in specific situations.

While the experimental results unequivocally highlight certain conditions of the GANs that need to be considered when applying AVATAR, detailed statistical evidence remains mostly missing due to limited sample sizes. Hence, the results of this paper should raise awareness to the research community and provide the following three research directions. First, the results of the parameter k investigations motivate future experimental evaluations to derive a rule-of-thumb to select an optimal value k. This requires an experimental evaluation using a large set of different ground truth systems to derive a robust rule-of-thumb. Second, a larger set of experiments need to be conducted to investigate the required variant log size to train a converging GAN such that AVATAR can be applied confidently. Third, the bias sensitivity of the GANs of AVATAR needs to be investigated with a larger set of ground truth systems and with different β hyperparameter values to unveil a potential relationship between β and the GAN sensitivity towards bias.

References

1. van der Aalst, W., Adriansyah, A., van Dongen, B.: Replaying history on process models for conformance checking and performance analysis. Wiley Interdisc. Rev. Data Mining Knowl. Disc. **2**(2), 182–192 (2012). https://doi.org/10.1002/widm.1045
2. van der Aalst, W.M., et al.: Business process mining: an industrial application. Inf. Syst. **32**(5), 713–732 (2007)
3. Arora, S., Ge, R., Liang, Y., Ma, T., Zhang, Y.: Generalization and equilibrium in generative adversarial nets (GANs). In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 224–232. JMLR. org (2017)
4. Augusto, A., Conforti, R., Dumas, M., La Rosa, M., Polyvyanyy, A.: Split miner: automated discovery of accurate and simple business process models from event logs. Knowl. Inf. Syst. **59**(2), 251–284 (2018). https://doi.org/10.1007/s10115-018-1214-x
5. vanden Broucke, S.K., De Weerdt, J.: Fodina: a robust and flexible heuristic process discovery technique. Decis. Support. Syst. **100**, 109–118 (2017)
6. vanden Broucke, S.K., De Weerdt, J., Vanthienen, J., Baesens, B.: Determining process model precision and generalization with weighted artificial negative events. IEEE Trans. Knowl. Data. Eng. **26**(8), 1877–1889 (2013)
7. Buijs, J.C., Van Dongen, B.F., Van Der Aalst, W.M.: Quality dimensions in process discovery: the importance of fitness, precision, generalization and simplicity. Int. J. Cooper. Inf. Syst. **23**(1), 1440001 (2014). https://doi.org/10.1142/S0218843014400012

8. Buijs, J.C.A.M., van Dongen, B.F., van der Aalst, W.M.P.: On the role of fitness, precision, generalization and simplicity in process discovery. In: Meersman, R., et al. (eds.) OTM 2012. LNCS, vol. 7565, pp. 305–322. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33606-5_19
9. Darabi, H., Galanter, W.L., Lin, J.Y., Buy, U., Sampath, R.: Modeling and integration of hospital information systems with Petri nets. In: 2009 IEEE/INFORMS International Conference on Service Operations, Logistics and Informatics, pp. 190–195, July 2009. https://doi.org/10.1109/SOLI.2009.5203928
10. van Dongen, B.F., Carmona, J., Chatain, T.: A unified approach for measuring precision and generalization based on anti-alignments. In: La Rosa, M., Loos, P., Pastor, O. (eds.) BPM 2016. LNCS, vol. 9850, pp. 39–56. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-45348-4_3
11. Ghasemi, M., Amyot, D.: Process mining in healthcare: a systematised literature review. Int. J. Electron. Healthcare **9**(1), 60–88 (2016)
12. Janssenswillen, G., Depaire, B.: Towards confirmatory process discovery: making assertions about the underlying system. Bus. Inf. Syst. Eng. **61**(6), 713–728 (2018). https://doi.org/10.1007/s12599-018-0567-8
13. Janssenswillen, G., Donders, N., Jouck, T., Depaire, B.: A comparative study of existing quality measures for process discovery. Inf. Syst. **71**, 1–15 (2017). https://doi.org/10.1016/j.is.2017.06.002
14. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equation of state calculations by fast computing machines. J. Chem. Phys. **21**(6), 1087–1092 (1953)
15. Nie, W., Narodytska, N., Patel, A.: RelGAN: relational generative adversarial networks for text generation. In: International Conference on Learning Representations (2018)
16. Rehse, J.-R., Fettke, P., Loos, P.: Process mining and the black swan: an empirical analysis of the influence of unobserved behavior on the quality of mined process models. In: Teniente, E., Weidlich, M. (eds.) BPM 2017. LNBIP, vol. 308, pp. 256–268. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-74030-0_19
17. Fani Sani, M., van Zelst, S.J., van der Aalst, W.M.P.: The impact of biased sampling of event logs on the performance of process discovery. Computing **103**(6), 1085–1104 (2021). https://doi.org/10.1007/s00607-021-00910-4
18. Syring, A.F., Tax, N., van der Aalst, W.M.P.: Evaluating conformance measures in process mining using conformance propositions. In: Koutny, M., Pomello, L., Kristensen, L.M. (eds.) Transactions on Petri Nets and Other Models of Concurrency XIV. LNCS, vol. 11790, pp. 192–221. Springer, Heidelberg (2019). https://doi.org/10.1007/978-3-662-60651-3_8
19. Theis, J., Mokhtarian, I., Darabi, H.: Process mining of programmable logic controllers: input/output event logs. In: 2019 IEEE 15th International Conference on Automation Science and Engineering (CASE), pp. 216–221, August 2019. https://doi.org/10.1109/COASE.2019.8842900
20. Theis, J., Darabi, H.: Adversarial system variant approximation to quantify process model generalization. IEEE Access **8**, 194410–194427 (2020)
21. Theis, J., Galanter, W., Boyd, A., Darabi, H.: Improving the in-hospital mortality prediction of diabetes ICU patients using a process mining/deep learning architecture. IEEE J. Biomed. Health Inf. **26**(1), 388–399 (2022). https://doi.org/10.1109/JBHI.2021.3092969
22. Turner, R., Hung, J., Frank, E., Saatchi, Y., Yosinski, J.: Metropolis-Hastings Generative Adversarial Networks. In: International Conference on Machine Learning, pp. 6345–6353 (2019)

23. Yang, H., Park, M., Cho, M., Song, M., Kim, S.: A system architecture for manufacturing process analysis based on big data and process mining techniques. In: 2014 IEEE International Conference on Big Data (Big Data), pp. 1024–1029. IEEE (2014)