



Deposited via The University of Sheffield.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/229416/>

Version: Published Version

---

**Proceedings Paper:**

Xing, W.W., Chen, H., Chen, Z. et al. (2025) Adaptive LCI data completion: Integrating neural processes and active learning for enhanced life cycle assessment. In: Mativenga, P. and Gallego-Schmid, A., (eds.) Procedia CIRP. 32nd CIRP Conference on Life Cycle Engineering (LCE 2025), 07-09 Apr 2025, Manchester, United Kingdom. Elsevier, pp. 136-141. ISSN: 2212-8271. EISSN: 2212-8271.

<https://doi.org/10.1016/j.procir.2025.01.046>

---

**Reuse**

This article is distributed under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs (CC BY-NC-ND) licence. This licence only allows you to download this work and share it with others as long as you credit the authors, but you can't change the article in any way or use it commercially. More information and the full terms of the licence here: <https://creativecommons.org/licenses/>

**Takedown**

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing [eprints@whiterose.ac.uk](mailto:eprints@whiterose.ac.uk) including the URL of the record and the reason for the withdrawal request.

32nd CIRP Conference on Life Cycle Engineering (LCE 2025)

# Adaptive LCI Data Completion: Integrating Neural Processes and Active Learning for Enhanced Life Cycle Assessment

Wei W. Xing<sup>#a</sup>, Hong Chen<sup>#b</sup>, Zidong Chen<sup>a</sup>, Zhishan Quan<sup>c</sup>, Bertrand Laratte<sup>d</sup>, Mark Walsh<sup>e</sup>,  
Jing Pu<sup>f</sup>, Jose L. Casamayor<sup>f,\*</sup>

<sup>a</sup>School of Mathematical and Physical Sciences, University of Sheffield, Western Bank, Sheffield, S10 2TN, UK

<sup>b</sup>City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong

<sup>c</sup>College of Mechatronics and Control Engineering, Shenzhen University, Shenzhen, China

<sup>d</sup>Laval University, Department of Wood and Forest Sciences, 2425 De La Terrasse Street, Québec City, QC G1V 0A6, Canada

<sup>e</sup>Seco Tools UK, Adams Way, Alcester, B49 6PU, UK

<sup>f</sup>Advanced Manufacturing Research Centre (AMRC), University of Sheffield, Wallis Way, Catcliffe, Rotherham, S60 5TZ, UK

## Abstract

Accurate and comprehensive Life Cycle Inventory (LCI) data underpins the reliability and accuracy of Life Cycle Assessment (LCA) results. However, LCI data is often incomplete due to data unavailability, which affects the reliability and accuracy of LCA results. To address this issue, this paper introduces a novel approach for LCI data completion based on Neural Processes (NPs) combined with active learning for efficient adaptive refinement of LCI data completion. Experimental results demonstrate that the proposed approach outperforms the state-of-the-art XGBoost-based method significantly, achieving up to 99% improvement in prediction accuracy. This means that by reducing data requirements by approximately 50% whilst improving predictive accuracy, the proposed AI model can provide more reliable LCA results in less time.

© 2025 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the International Academy for Production Engineering (CIRP)

**Keywords:** Life Cycle Assessment, Life Cycle Inventory, Neural Processes, Active Learning

## 1. Introduction

Life Cycle Assessment (LCA) is one of the most widely used methods for evaluating the environmental impact of products and processes across their entire lifecycle [1]. As global attention to sustainable development and environmental protection intensifies, LCA has been widely adopted in industry, academia, and policy-making [2]. LCA enables a comprehensive assessment of potential environmental impacts throughout a product's or service's life cycle, from raw material acquisition through production, use, and disposal, thereby helping decision-makers understand and to take action towards reducing environmental burdens [3].

A key step in LCA methodology is Life Cycle Inventory (LCI) analysis, which involves compiling a detailed inventory of inputs (such as resources and energy flows) and outputs (such as emissions and other releases) for each unit process in a product's life cycle [4]. A comprehensive and accurate LCI is the foundation for accurate LCA results. However, the collection of LCI data can be challenging due to the intensive nature of collecting field data and confidential reasons, which poses the major challenge in compiling complete life cycle inventories [5].

\*Corresponding author

<sup>#</sup>These authors contributed equally to this work.

To address this data challenge, researchers have developed methods to estimate missing LCA data based on available data. Canals *et al.* suggested using proxy datasets or extrapolated data [6]. Olivetti *et al.* introduced the structured under-specification method, which allows for the inference of missing information by leveraging structural relationships between existing data [7]. Wernet *et al.* pioneered the use of neural network models focusing on molecular structure to predict the environmental burden of chemical production [8]. Specifically, they utilized molecular structures as input features, enabling the model to infer environmental impacts based on the chemical properties of substances. This approach was groundbreaking, and building on this, Song *et al.* developed a deep artificial neural network model that improved both the speed and accuracy of life cycle impact assessments for chemicals by using larger datasets and more complex network architectures [9]. Hou *et al.* employed various machine learning models, including random forests and support vector machines, to address the problem of missing ecotoxicity characterization factors for toxic chemicals in LCA [10]. Dai *et al.* adopted Gaussian process regression models to quantify the similarities between variables, addressing issues related to data gaps in agricultural data and the quantification of sec-

ondary data uncertainty [11]. However, due to the limited availability of training data and the significant variability in environmental impact factors across sectors, these machine learning methods are mostly restricted to specific industries and technological domains. Expanding the application to a broader range of industries remains an ongoing challenge.

To overcome these limitations, Hou et al. proposed a similarity-based link prediction method [12], which was further optimized using the XGBoost model [13]. This approach is considered state-of-the-art and aims to characterize the relationship between known information (predictors) and missing information (response variables) using existing LCI datasets. Although this method has improved predictive capabilities, its reliability may still be compromised in the absence of structured LCA data, as it relies on low-precision proxy data or assumptions.

Despite these advances, there are still aspects that need to be addressed: 1) **Process Correlations:** Existing methods often treat industrial processes as independent entities, failing to capture the inherent correlations between different processes in a product's life cycle. 2) **Adaptation to New Data:** Many current approaches struggle to efficiently incorporate new data (i.e., data updates) as it becomes available, requiring full model retraining. 3) **Targeted Data Collection:** Currently, there is insufficient guidance to help determine which data should be prioritized for collection, based on their influence on the precision of the LCI. 4) **Utilization of Process Information:** Many methods, including the latest XGBoost approach, rely solely on known flow values, ignoring potentially additional valuable information contained in process descriptions.

To address these challenges, a novel approach to LCI completion using Neural Processes (NPs) [14] is proposed, which is a state-of-the-art machine learning technique that combines the strengths of neural networks (NNs) and Gaussian processes (GPs). NPs define a distribution over functions like GPs, providing natural uncertainty estimates, which are crucial for reliable predictions. Additionally, they leverage NNs to learn flexible and efficient representations, enabling them to capture complex patterns in data that traditional GPs might struggle with. The proposed method offers several advantages over existing approaches: 1). **Incorporating Process Information:** The approach utilizes both process descriptions and known flow values, potentially leading to more accurate and generalizable predictions. 2). **Capturing Process Correlations:** NPs can learn and leverage the underlying relationships between different industrial processes, leading to more accurate predictions of missing LCI data. 3). **Uncertainty Quantification:** The method provides natural uncertainty estimates of LCA results based on different types of LCI data estimated, crucial to quantify the uncertainty of the LCA and to guide further data collection efforts. 4). **Efficient Adaptation to New Data:** The approach allows for the incorporation of new data without requiring full model retraining, enabling continuous update of LCI datasets. 5). **Adaptive Data Completion via Active Learning:** An active learning strategy is incorporated to guide data collection efforts

towards the most informative and relevant points, progressively improving the accuracy of LCI data completion.

## 2. Problem Formulation

LCI data forms the foundation of LCA, providing detailed information about the inputs and outputs of industrial processes. Typically, LCI data can be structured as a matrix where rows represent different flows (inputs and outputs) and columns represent distinct unit processes. Each entry in this matrix quantifies the amount of a specific flow associated with a particular process.

The LCI completion problem arises when this dataset is incomplete, containing missing entries that need to be predicted. Formally, given a set of observed process-flow pairs  $C = \{(x_i, y_i)\}_{i=1}^{n_c}$ , where  $x_i$  represents process characteristics and  $y_i$  represents flow values, the goal is to estimate missing entries. The objective is to identify a function  $f$  that can predict any missing flow value  $y_j$  for a given process  $x_j$ :

$$\hat{y}_j = f(C, x_j), \quad (1)$$

where  $C$  represents the observed process-flow pairs. It's important to note that  $x_i$  typically consists of natural language descriptions of the process, making it challenging to directly use as input to mathematical equations. The flow value  $y_i$ , on the other hand, is a numerical values that can be more readily incorporated into a machine learning pipeline.

Recently, Zhao et al. [13] approached this problem using the XGBoost algorithm, a decision tree-based supervised learning method. To circumvent the challenge of using process descriptions, their approach avoids using  $x$  as input and instead relies solely on the known flow values. For each process  $j$ , let  $p_j$  be the set of indices of missing flows, and  $q_j$  be the set of indices of known flows. The prediction function  $f_j$  for process  $j$  can be expressed:

$$\hat{y}_{ij} = f_j(\{y_{kj} : k \in q_j\}, i), \quad \text{for } i \in p_j. \quad (2)$$

The XGBoost method [13] does not consider the interdependencies between flows and processes, which do exist in real-world LCI datasets at different levels of complexity. Additionally, this approach lacks comprehensive uncertainty quantification, crucial to quantifying the uncertainty of each predicted LCI data value. Moreover, by not utilizing the process characteristics  $x$ , the method may miss important contextual information that could improve prediction accuracy and generalization to new processes.

## 3. Methodology

The LCI completion problem is formulated as a stochastic process over functions  $f : \mathcal{X} \rightarrow \mathcal{Y}$ , where  $\mathcal{X}$  represents process characteristics and  $\mathcal{Y}$  flow values. Given a set of observed context points  $C = \{(x_i, y_i)\}_{i=1}^{n_c}$  and target points  $T = \{(x_j, y_j)\}_{j=1}^{n_t}$ , a distribution over functions is learned to predict flow values for processes with unknown values.

In this paper, NPs are used to solve the above LCI completion problem. Its architecture consists of three main components: 1). Encoder:  $h_\theta : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$  The encoder maps each process-flow pair to a  $d$ -dimensional representation, where  $d$  is a hyperparameter determining the dimensionality of the latent space. This encoding captures the

essential features of each process-flow relationship. 2). Aggregator:  $a_\phi : \mathbb{R}^{d \times n_c} \rightarrow \mathbb{R}^d$  The aggregator combines the individual representations into a single global representation, allowing the model to capture dependencies between different processes and flows. 3). Decoder:  $g_\psi : \mathcal{X} \times \mathbb{R}^d \rightarrow \mathcal{Y}$  The decoder uses the global representation to generate predictions for target processes, producing both the predicted flow value and its associated uncertainty. For example, each observed process-flow pair  $(x_i, y_i)$ , is represented by:

$$r_i = h_\theta(x_i, y_i). \quad (3)$$

$r_i$  captures the essential features of the process-flow pair  $(x_i, y_i)$ . By learning these representations, the model can learn various patterns in the LCI data without requiring manual feature engineering. This flexibility is particularly suitable in LCI completion, where relationships between processes and flows can be complex and domain-specific.

These representations are then aggregated into a global latent variable:

$$z = a_\phi(\{r_1, \dots, r_{n_c}\}). \quad (4)$$

The latent variable  $z$  encapsulates information from all observed process-flow pairs, allowing the model to capture dependencies between different processes and flows. The decoder generates predictions for target processes:

$$\hat{y}_j = g_\psi(x_j, z). \quad (5)$$

To capture uncertainty, the output is modeled as a Gaussian distribution:  $p(y_j|x_j, z) = \mathcal{N}(y_j|\mu_j, \sigma_j^2)$ , where  $\mu_j$  and  $\sigma_j^2$  are produced by the decoder.

This formulation allows us to capture complex relationships between processes and flows while providing uncertainty estimates, addressing key limitations of previous approaches. The stochastic nature of NPs enables them to model the potential noise in the data, which is crucial for reliable environmental impact assessments.

Compared to the XGBoost used by Zhao *et al.* [13] and other regression-based approaches, NPs offer several significant advantages for LCI completion. Firstly, NPs capture global dependencies across all processes and flows through the latent variable  $z$ , providing a holistic view of the data, while XGBoost treats each variable independently. This global context allows NPs to better understand and leverage the interconnections within the LCI data. Additionally, NPs provide natural uncertainty estimates, which are crucial to understand the accuracy and confidence of the predicted values, whereas XGBoost requires additional techniques to quantify uncertainties. The adaptability of NPs is another key advantage; they can easily incorporate new data without full retraining, making them very suitable for LCI databases which are continuously being updated, and where new updates can change the LCA results significantly. In contrast, XGBoost typically requires retraining of the entire new datasets to incorporate new information, which can be computationally expensive and time-consuming. Furthermore, NPs excel in representation learning, developing flexible representations of processes and flows that can capture more complex relationships than the fixed feature representations used in XGBoost.

### 3.1. Implementation, Training, and Analysis

The encoder and decoder are implemented as multi-layer perceptrons (MLPs) with Rectified Linear Unit (ReLU) activation function. MLPs are for their ability to approximate complex functions, making them suitable for capturing the intricate relationships in LCI data. The ReLU helps mitigate the vanishing gradient during the model training, allowing for more effective training of deep networks.

The aggregator uses a permutation-invariant mean operation:  $z = \frac{1}{n_c} \sum_{i=1}^{n_c} r_i$ . A permutation-invariant operation is chosen because the order of processes in an LCI dataset should not affect the predictions. This property ensures that the model's output remains consistent regardless of how the input data is arranged, which is crucial for the reliability and reproducibility of LCA results.

Training is performed using variational inference, maximizing the evidence lower bound (ELBO):

$$\mathcal{L} = \mathbb{E}_{q(z|C,T)}[\log p(y_T|x_T, z)] - KL(q(z|C, T)||p(z|C)). \quad (6)$$

Variational inference, a method from Bayesian statistics, is used to approximate the true posterior distribution over the latent variables. This approach allows us to balance the model's fit to the observed data with its uncertainty in predictions. The ELBO provides a tractable objective for optimization, combining two key terms: 1). The expected log-likelihood  $\mathbb{E}_{q(z|C,T)}[\log p(y_T|x_T, z)]$ , which measures how well the model fits the observed data, and 2). The Kullback-Leibler (KL) divergence  $KL(q(z|C, T)||p(z|C))$ , which acts as a regularization term, encouraging the learned posterior  $q(z|C, T)$  to stay close to prior  $p(z|C)$ , preventing overfitting.

This formulation allows the model to be trained in a way that naturally balances accuracy and generalization, which is particularly important in the context of LCI data where overfitting to sparse observations could lead to unreliable predictions. The computational complexity of the method is  $O(mn)$  for both training and inference, where  $m$  is the number of flows and  $n$  is the number of processes. This is comparable to the XGBoost approach, but offers greater flexibility in handling missing data patterns, particularly in an incremental manner.

### 3.2. Adaptive LCI Completion with Active Learning

LCI databases are continually evolving, with new processes and updated measurements becoming available over time. To address this dynamic nature and maximize the efficiency of data collection efforts, an active learning strategy is incorporated. This approach allows for the iterative improvement of the model and guides data collection efforts, focusing on the most informative data points.

After initial training on the observed data  $X_\Omega$ , the most informative missing entries are iteratively selected for querying. An acquisition function is then proposed to balance the uncertainty of predictive values with the potential accuracy improvement of the model.

$$a(i, j) = \sigma_{ij} + \lambda \cdot \mathbb{E}[\Delta Y_{ij}], \quad (7)$$

where:  $\sigma_{ij}$  is the predicted uncertainty for  $i$  flows of  $j$  process, directly obtained from the Neural Processes model;

$\mathbb{E}[\Delta Y_{ij}]$  is the expected magnitude of change in all LCI target data to be predicted if  $(i, j)$  were observed;  $\lambda$  is a trade-off parameter balancing prioritizing the challenging values (with high predictive uncertainty) of those that potentially improve the model.

This acquisition function combines two key principles of active learning: 1. Uncertainty sampling: By including  $\sigma_{ij}$ , entries where the model is most uncertain are prioritized, potentially leading to the greatest improvement in model performance. 2. Expected model change:  $\mathbb{E}[\Delta Y_{ij}]$  prioritizes entries that are expected to have the largest improvement on the target LCI values, ensuring that the data collection efforts focus on influential entries.  $\mathbb{E}[\Delta Y_{ij}]$  is estimated using the current model's predictions and their uncertainties,  $\mathbb{E}[\Delta Y_{ij}] \approx \sum_{k,l} \left| \frac{\partial \mu_{kl}}{\partial x_{ij}} \right| \cdot \sigma_{ij}$ , where  $\frac{\partial \mu_{kl}}{\partial x_{ij}}$  captures the influence of entry  $(i, j)$  on other entries in the matrix. This term is computed using automatic differentiation, leveraging the end-to-end differentiable nature of the NP model. For models where the computation is infeasible, This term can be approximated using Monte Carlo sampling or simply set  $\lambda = 0$  to focus solely on uncertainty. At each iteration, the entry with the highest acquisition score is selected:

$$(i^*, j^*) = \underset{(i,j) \notin \Omega}{\operatorname{argmax}} a(i, j). \quad (8)$$

Compared to traditional static methods in LCI completion, the active learning strategy offers several advantages: 1. Adaptive data collection: The approach dynamically adjusts its focus based on the current state of the model and LCI matrix, ensuring optimally informative data collection throughout the process. 2. Uncertainty quantification: By incorporating uncertainty estimates, Areas of high uncertainty can be identified and prioritized, leading to more robust LCI completions in less time. 3. System-wide impact assessment: Considering the expected change in the entire LCI matrix allows us to prioritize entries with far-reaching effects, potentially leading to more efficient improvements in overall accuracy. 4. Iterative refinement: The approach allows for continuous model improvement as new data is collected, without requiring complete recomputation or re-training. 5. Scalability: By focusing on the most informative data points, the method is suitable for large-scale LCI databases where the data collection is impractical.

This adaptive approach represents a significant advancement in addressing the evolving nature of LCI databases and the need for efficient, targeted data collection in LCA.

#### 4. Experiments and Results

The proposed Neural Processes model was evaluated using the Tiangong database<sup>2</sup>, a comprehensive set of unit process data (UPR)[15]. The basic compositions of this database are the individual processes in human activity and their input-output exchange with the environment (elementary exchange) and technological systems (intermediate exchange). Each process typically refers to a specific activity, such as production, transportation, or waste disposal. For example, it could involve the production of 1 kg of cement.

<sup>2</sup><https://www.tiangong.earth/zh/data>

Flows in the process are categorized into two types: intermediate flows and elementary flows. Intermediate flows are material transfers between processes that are not directly exchanged with the environment. Elementary flows are directly exchanged with the environment, such as emissions or resource extraction. After preprocessing to remove duplicates and extraneous processes, the final dataset contains 3,804 unique processes (UPR) and 2,969 distinct flows, resulting in a sparse matrix with 0.3916% non-zero elements.

Following Zhao *et al.* [13], the LCI dataset is split into training and test sets. Ten processes are randomly selected for the test set, with the remaining 3,794 processes serving as the training set. For test processes, the missing data ratio of 1%, 5%, 10%, 20%, 30%, 50%, and 70% were selected to represent typical rates of incompleteness in LCI datasets. These rates span from minor gaps (1%-10%) to severe data scarcity (50%-70%), providing a balanced range to evaluate the model's performance under varying conditions and ensure its applicability across diverse scenarios.

The non-missing flow values serve as input for the model to predict the missing values. The method is compared with the state-of-the-art LCI completion using XGBoost [13], evaluating model accuracy using Normalized Root Mean Square Error (NRMSE), Normalized Mean Square Error (NMSE), and Normalized Mean Absolute Error (NMAE). To ensure robustness, each experiment is repeated 10 times, and the average errors are reported.

##### 4.1. Prediction Accuracy at Different Missing Data Ratio

Figure 1 compares the performance of the NP model and XGBoost across different missing data ratio. The NP model demonstrates superior robustness and consistency in handling missing data, with its error remaining relatively stable as the missing ratio increases, while XGBoost's errors rise significantly. As the missing ratio exceeds 5%, the NP model's advantage becomes evident, outperforming XGBoost across NRMSE, NMSE, and NMAE metrics. An exception occurs at the 20% missing ratio, where XGBoost slightly outperforms the NP model in terms of NMAE only. This could be attributed to XGBoost's tree-based structure capturing subtle variations more effectively. However, the NP model's superior performance in NRMSE and NMSE suggests it is better at suppressing large errors. At the extreme 70% missing ratio, the NP model demonstrates significant improvements over XGBoost. The NMSE of the NP model is approximately 1E-06, while that of XGBoost is around 1.1E-04, representing an improvement of about 99.09%. The NP model also outperforms XGBoost in NMAE at this high missing ratio, underscoring its robustness and accuracy in challenging scenarios.

Overall, the NP model exhibits stronger performance across most scenarios, particularly at high missing data ratio, surpassing XGBoost in both predictive accuracy and consistency. This performance advantage is crucial for practical LCI completion tasks, which often involve substantial amounts of missing data.

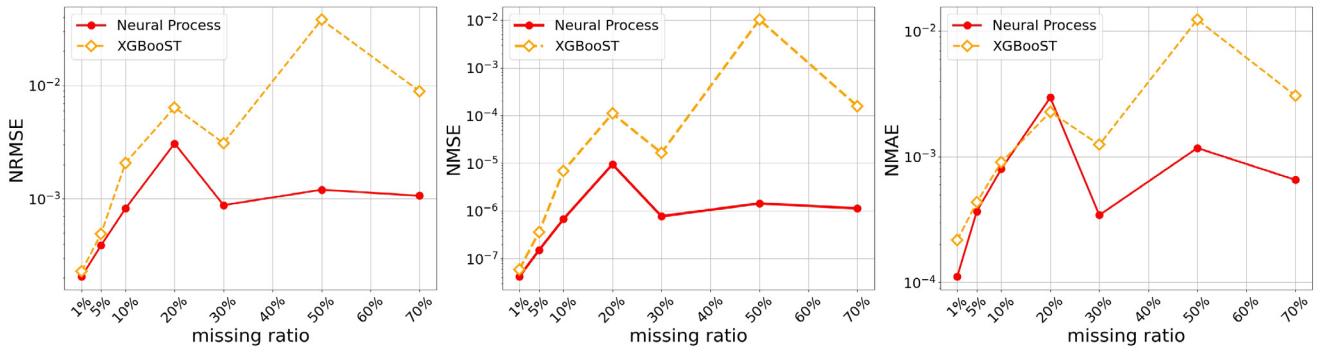


Figure 1. Predictive errors (NRMSE, NMSE, and NMAE) under different missing data ratio.

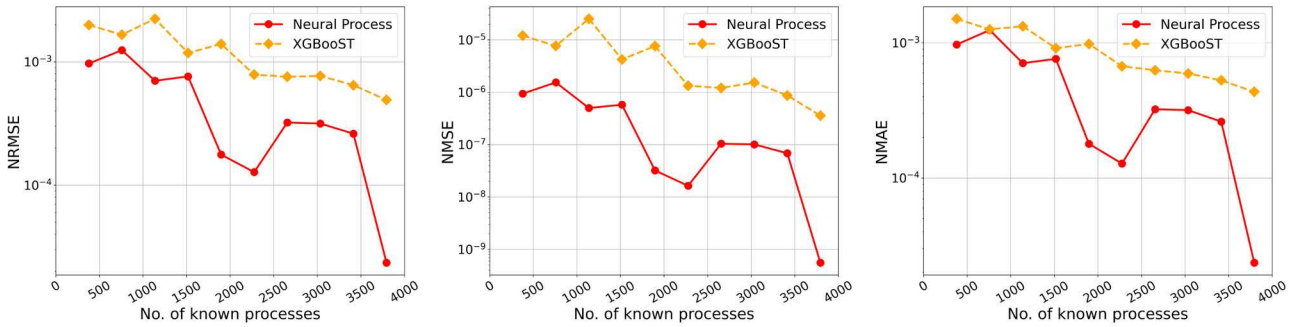


Figure 2. Predictive errors (NRMSE, NMSE, and NMAE) with increasing number of known processes.

#### 4.2. Prediction Accuracy With Varying Training Data Sizes

To evaluate the method’s adaptability to different training data sample sizes, simulating real-world scenarios of varying data availability, an experiment was conducted with incrementally increasing training set sizes. A fixed test set of 10 processes and a consistent missing data ratio of 5% were maintained to ensure comparability across set sizes.

Starting with a baseline of 10% of the known processes (379 processes), the training set is gradually expanded to the full 3,794 processes in nine steps, each increasing by approximately 10% (except the final step). Figure 2 shows the model predictions’ accuracy in terms of NRMSE, NMSE, and NMAE as the number of known processes grows.

The results show that the NP model consistently outperforms XGBoost across all training set sizes, especially as the number of known processes increases, with the NP model demonstrating a significant advantage in all metrics. When the training data is small, both models perform similarly, but as the number of known processes grows, the NP model’s advantage becomes more apparent. Notably, when the NP model is trained on only about 40% of the training set (approximately 1,517 processes), its performance already rivals that of XGBoost on the full dataset, highlighting its superior ability to extract and generalize patterns from limited data. As the training set approaches the full dataset (3,794 processes), the NP model’s performance improves further, achieving NMSE  $5.6E-10$ , NRMSE  $2.36E-5$ , and NMAE  $2.3E-05$ , accuracy improved by more than 90% compared to XGBoost.

However, in the range of 2200–2600 known processes, a slight performance degradation is observed in the NP model. This is likely due to the addition of less infor-

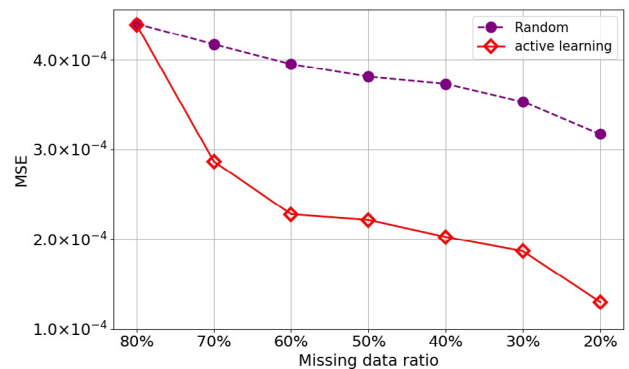


Figure 3. Comparison between active learning and random sampling.

native or redundant processes, which introduce noise and temporarily hinder the model’s learning. Unlike XGBoost, which relies heavily on data volume, the NP model is more sensitive to data quality. Despite this, the NP model quickly recovers and continues to improve as the training set grows. This behavior highlights the importance of data quality in NP-based approaches. While increasing training data quantity does not always guarantee better performance, the NP model’s ability to generalize effectively from high-quality data makes it suitable for LCI data collection scenarios.

#### 4.3. Prediction Accuracy With Adaptive Data Acquisition

In the simulation, it was assumed that only a limited subset of data points could be acquired. The dataset used for this study exhibits sparsity and variability, reflecting challenges encountered in real-world industrial applications, where certain processes have incomplete or unavailable records. This simulation is also applicable to scenarios in

which data collection efforts are constrained by economic or operational limitations. The active learning approach was compared with random sampling, using 2,794 fully observed processes as training data and 10 processes with 80% missing data as the test set. During the 6 rounds of experiments, an additional 10% of the missing data is incrementally selected for completion in each round, evaluating the model's performance by measuring the NMSE of the predicted remaining missing flows in the test set.

Figure 3 illustrates the MSE of the model's predictions after each iteration. Both methods started with an MSE of approximately  $4.2E-4$ , but active learning quickly demonstrated superior performance. At 70% missing data, active learning reduced the MSE to  $2.8E-4$ , while random sampling remained around  $4.1E-4$ . This performance gap continued to widen as iterations progressed. At lower missing data ratio, active learning showed more significant improvements. For example, at 20% missing data, the MSE was reduced to  $1.3E-4$ , while random sampling remained above  $3.1E-4$ . This resulted in a 58.06% improvement in accuracy with active learning.

In terms of efficiency, active learning achieves higher accuracy at a 70% missing data ratio compared to random sampling at a 20% missing data ratio, while also reducing the query count by approximately 50%. This translates directly to cost savings in real-world scenarios, where each query may represent an expensive or time-consuming data acquisition process. The consistent superior performance of active learning over random sampling underscores the value of intelligent data selection in LCI completion tasks. By identifying the most relevant data points, active learning enables faster improvements in model performance, making it particularly valuable in scenarios where data is available but challenging or costly to obtain.

This approach not only improves model accuracy but also optimizes resource allocation in data collection efforts, addressing one of the problems in LCA studies, which is the amount of resources required to collect the large amounts of data needed. By focusing on the most impactful and relevant data points, active learning provides a more efficient and cost-effective approach to enhancing LCI completion, particularly valuable in resource-intensive scenarios.

## 5. Conclusion

In this study, a novel NP-based approach is presented to enhance LCI data completion, which showed a significant improvement in accuracy (10x), compared to state-of-the-art methods, with less data (and time) requirements. In addition, the approach introduces a technique for selective data collection among unknown processes/flows via active learning to allow similar predictive accuracy with 50% less missing data. The findings have significant implications for the field of Life Cycle Engineering (LCE) and broader LCA. By addressing the challenge of incomplete LCI data, the proposed approach provides a scalable solution that reduces the reliance on comprehensive data collection, making LCA more accessible and reliable for industrial applications and policy evaluations. However, the adaptability of

the method to databases with significantly different structures remains a challenge. Future work will focus on enhancing the robustness of the model, particularly in handling diverse data distributions and missing data patterns.

## Acknowledgement

The authors would like to thank UKRI (EPSRC-IAA: Grant 187882) and Seco Tools UK Ltd. for their financial support; and Rebecca Holbach and Dr Bin Chen for their input and support in this research.

## References

- [1] International Organization for Standardization, Environmental management—life cycle assessment—principles and framework, ISO 14040:2006.
- [2] G. Finnveden, M. Z. Hauschild, T. Ekvall, J. Guinée, R. Heijungs, S. Hellweg, A. Koehler, D. Pennington, S. Suh, Recent developments in life cycle assessment, *Journal of Environmental Management* 91 (1) (2009) 1–21.
- [3] G. Rebitzer, T. Ekvall, R. Frischknecht, D. Hunkeler, G. Norris, T. Rydberg, W.-P. Schmidt, S. Suh, B. P. Weidema, D. W. Pennington, Life cycle assessment: Part 1: Framework, goal and scope definition, inventory analysis, and applications, *Environment International* 30 (5) (2004) 701–720.
- [4] S. Hellweg, L. Milà i Canals, Emerging approaches, challenges and opportunities in life cycle assessment, *Science* 344 (6188) (2014) 1109–1113.
- [5] S. Zargar, Y. Yao, Q. Tu, A review of inventory modeling methods for missing data in life cycle assessment, *Journal of Industrial Ecology* 26 (5) (2022) 1676–1689.
- [6] L. M. i. Canals, A. Azapagic, G. Doka, D. Jefferies, H. King, C. Mutel, T. Nemecek, A. Roches, S. Sim, H. Stichnothe, et al., Approaches for addressing life cycle assessment data gaps for bio-based products, *Journal of Industrial Ecology* 15 (5) (2011) 707–725.
- [7] E. Olivetti, S. Patanavanich, R. Kirchain, Exploring the viability of probabilistic under-specification to streamline life cycle assessment, *Environmental Science & Technology* 47 (10) (2013) 5208–5216.
- [8] G. Wernet, S. Hellweg, U. Fischer, S. Papadokonstantakis, K. Hungerbühler, Molecular-structure-based models of chemical inventories using neural networks, *Environmental Science & Technology* 42 (17) (2008) 6717–6722.
- [9] R. Song, A. A. Keller, S. Suh, Rapid life-cycle impact screening using artificial neural networks, *Environmental Science & Technology* 51 (18) (2017) 10777–10785.
- [10] P. Hou, O. Jolliet, J. Zhu, M. Xu, Estimate ecotoxicity characterization factors for chemicals in life cycle assessment using machine learning models, *Environment International* 135 (2020) 105393.
- [11] T. Dai, S. M. Jordaan, A. P. Wemhof, Gaussian process regression as a replicable, streamlined approach to inventory and uncertainty analysis in life cycle assessment, *Environmental Science & Technology* 56 (6) (2022) 3821–3829.
- [12] P. Hou, J. Cai, S. Qu, M. Xu, Estimating missing unit process data in life cycle assessment using a similarity-based approach, *Environmental Science & Technology* 52 (9) (2018) 5259–5267.
- [13] B. Zhao, C. Shuai, P. Hou, S. Qu, M. Xu, Estimation of unit process data for life cycle assessment using a decision tree-based approach, *Environmental science & technology* 55 (12) (2021) 8439–8446.
- [14] M. Garnelo, J. Schwarz, D. Rosenbaum, F. Viola, D. J. Rezende, S. Eslami, Y. W. Teh, Neural processes, *arXiv preprint arXiv:1807.01622*.
- [15] G. Wernet, C. Bauer, B. Steubing, J. Reinhard, E. Moreno-Ruiz, B. Weidema, The ecoinvent database version 3 (part i): overview and methodology, *The International Journal of Life Cycle Assessment* 21 (9) (2016) 1218–1230.