Confirmation of Candidature

Design and Evaluation of Explainable Methods for Predictive Process Analytics

Student: Mythreyi Velmurugan
Student ID: 9455647

Submitted in partial fulfillment of the requirement for the degree of
IF49 Doctor of Philosophy

School of Information Systems
Faculty of Science
Queensland University of Technology

August 11, 2022

Contents

List of Figures

List of Tables

Proposed Thesis Title

Design and Evaluation of Explainable Methods for Predictive Process Analytics

Proposed Supervisory Team

Principal Supervisor: Dr Chun Ouyang

Dr Ouyang is a Senior Lecturer in the School of Information Systems at QUT, and has supervised three PhD students to completion. She has strong research interest in Process-aware Information Systems, particularly in data-driven process analytics; process automation; and standards and methods used in industry.

Associate Supervisor: Dr Catarina Pinto Moreira

Dr Moreira is a lecturer in the Information Science discipline at the School of Information Systems. She specialises in Enterprise Systems and Artificial Intelligence, and the use of quantum-like models in describing decision-making.

Associate Supervisor: Dr Renuka Sindhgatta Rajan

Dr Sindhgatta is a lecturer in the Service Science discipline at the School of Information Systems. She has previously worked in industry research labs, notably including IBM. Her research focuses on the development of service delivery systems, particularly BPM solutions, and interpretable machine learning.

Thesis Type

Thesis by Monograph

1 Introduction

Business Process Management (BPM) methods are increasing in technical complexity and sophistication. An emerging BPM technique is predictive process analytics (PPA), which, at runtime, attempts to predict some future state of a process using machine learning or AI-enabled systems [29]. Modern data analytics techniques and increased availability of machine-generated process data has enabled PPA which applies predictive analytics to business processes, for example, to predict the most likely outcome of a running process instance [47]. A common input for PPA is business process event logs [29], though the use of contextual information and other data is also possible [54]. PPA can be used for better decision-making in organisations, optimisation or other process management activities, and discovering previously unknown factors or relations between factors affecting process performance [38].

However, there are concerns regarding the opaque nature of some machine learning and AI systems. While the complexity and sophistication of prediction algorithms are often correlated with the system's prediction accuracy, the resulting predictive models become less transparent due to their sophisticated internal representations - an issue that could affect an organisation's transparency, ethical conduct, accountability and liability, as well as raising potential issues with regards to safety and fairness when dealing with stakeholders [16]. Lack of transparency in machine decision making can also reduce trust in predictions made, and, as such, lead to reluctance in using machine learning and AI-enabled predictions and recommendations [38,43]. More transparency regarding the decision-making processes of automated models can assist in understanding why a particular prediction or decision was made by a machine; encouraging social acceptance of a prediction or prediction models; and ensuring the safety of a system, especially in domains where predictions may have significant impact, such as in medical or legal decision making [6].

Moreover, the need for interpretability in algorithmic decision-making has become a key part of standards, guidelines and legislation in a number of countries. In particular, the European Union has mandated the right of stakeholders to be provided ”meaningful explanations of the logic involved” in algorithmic decisions as part of the General Data Protection Regulation (GDPR) [16], which is relevant not only to European businesses, but any organisation operating in the EU or monitoring the behaviour of individuals in the EU [36]. Moreover, in May 2020, the International Standards Organisation released the ISO/IEC TR 24028 on the trustworthiness of AI systems, which also highlights the opaqueness of AI systems, and advises the use of explainability to mitigate this opaqueness [1].

Methods and techniques have been proposed in machine learning to explain such opaque “black box” models, forming a new research theme known as explainable AI (XAI) [16]. “Interpretability” refers to the ability to provide meaning regarding black box decision making in a human-understandable way [16], and refers not only to interpreting black box algorithms, but also inherently intepretable “white box” models [5, 6, 16]. Moreover, interpretations can be made at the global level (covering the whole model) or a small region of data, and can provide explanations regarding not only why a particular prediction or decision was returned, but also steps that can be taken to change a prediction or decision [33].

Several recent studies in predictive process analytics have attempted to apply existing XAI techniques to interpret process prediction models (for example, in [14, 41, 43]). However, especially given the variety of XAI techniques available and emerging, it is unclear how fit for purpose such intepretability techniques are when applied to PPA. Frameworks and metrics for determining XAI fitness for PPA are also so far under-explored. Further investigation is also needed to understand the impacts that characteristics of different datasets and black box models can have on explanation quality, and in understanding how to best provide explanations to support decision-making.

The proposed project, therefore, will address methods to improve AI and machine learning transparency in the field of predictive process analytics, provide a framework for evaluating these methods and attempt to provide a set of guidelines or recommendations for explainability in predictive process analytics. More specifically, there will be a focus on ensuring explanations can be used to empower decision-making, with emphasis on the understandability and comprehensibility of explanations. The following literature review (section 2) will provide an overview of PPA and prediction model explainability. Using this review, current research will be evaluated for any gaps and further research problems will be identified (section 3), followed by the proposed research design (section 4) and progress made to date (section 5). Planned future work will also be outlined (section 6).

2 Literature Review

2.1 Predictive Process Analytics

Process mining is a technique that attempts to derive process information from event logs recorded in information systems. Process mining has been described as, "...the missing link between data analytics and process science...", where data analytics techniques are applied to discover business process information [49]. This information could include activities undertaken, actors involved in process instances, and process metrics, among others, to effectively understand and reconstruct the process [49]. This technique forms the backbone of PPA, where event data is extracted from historical event logs and used by a prediction model to predict some future state of running process execution (known as the prediction target) [29].

Most techniques for PPA involve data processing, followed by learning and prediction, though there may be some variations depending on the data and learning algorithms used. While event logs are the most common data source for PPA, it can be supplemented by contextual information, such as case documentation, to provide more information to the learning algorithm that might prove useful in making the final prediction [29]. One form of context information that has been used in predictive process monitoring is documents associated with a process instance, such as order information or invoices [54]. Trials have also been conducted using media content [58] and context information associated with an instance, such information concerning the movement of goods in a logistic process [13].

Two PPA benchmark papers [47, 50] set out a two-phase approach for creating predictions that begins with an offline processing and learning phase where the predictor is created and trained, followed by deploying the predictor at runtime to make predictions. Various AI-enabled learning techniques and statistical methods have been used to create prediction models, though it is generally agreed that the most appropriate learning algorithm will depend on the process under examination, the needs of the end user and the prediction target [3, 29, 46, 47, 50]. Bucketing and encoding techniques may also be used to pre-process the data, depending on the type of learning algorithm used [47], though some algorithms, such as LSTM and other deep learning methods, do not require bucketing of data [50].

In PPA, the prediction target varies greatly. The outcome of a process instance [47], the time needed to complete the process instance [50] and the future activities of the process instance [46] have all been explored in detail, and risk prediction, Service Level Agreement violation prediction, abnormal event prediction and other targets have been trialled [29]. The underlying prediction models have achieved a relatively high accuracy, generally averaging 70% accuracy or more [29, 46, 47, 50]. Nevertheless, there are concerns regarding the use of PPA in real-world settings, particularly with regards to actionability [8, 56]. Actionability of predictions is often neglected in favour of quality and accuracy in literature [20], and it has been suggested that interpretation of predictions could be key to actionability [38] and to increasing trust in the predictor [38, 43].

State-of-the-art explainable predictive process analytics have generally attempted to use existing explainable methods in XAI. For example, the use of LIME and SHAP to evaluate and improve black box models [41, 43] has been explored, and SHAP has been used to create explainable dashboards for informed decision-making [14]. A method of interpreting process predictive models underpinned by neural networks using layer-wise relevance propagation is presented in [55]. In contrast to these works which use existing methods, a semi-interpretable, process-aware approach was proposed for remaining time predictions, where predictive models were trained for every activity and decision point in the process, and a prediction was made at each step to better understand how each activity and decision point contributed to the remaining time [51]. However, this cannot be considered to be fully interpretable, as it does not highlight the specific attributes that contributed to the final prediction.

2.2 Interpretability of Prediction Models

It has been suggested that a clearer understanding of the decision making of prediction models, or explanations for predictions, can be useful in increasing trust in a returned prediction [5,16,34,38,39,43–45], and in doing so, increase confidence in the prediction [5, 10,59]. In addition to this, the mandation of "the right to an explanation" by the GDPR has made machine learning explainability crucial [6,19,22]. Moreoever, explainability of corporate and commercialised AI systems are suggested to be of benefit to enterprises for decision-making and ensuring system quality [6, 22], often by promoting understanding of the limitations and boundaries of a system [59]. However, the "black box problem" of AI makes this difficult as many prediction models and algorithms applied can be difficult for humans to understand [22]. As such, explainable ML and AI systems have become a key topic.

Interpretability as a concept in itself is nebulously defined in literature [25], and other terms have often been used in its place, such as, "intelligbility", "explainability" and "transparency" [6]. In this document, the terms "interpretability" and "explanations" will be used as defined in [16], where "interpretability" refers to the ability to provide meaning in human understandable terms, and "explanations" are the interface between the human and the predictor. The term "explainability" will be used to refer to the ability to present interpretations in a meaningful and effective way.

Interpretability in machine learning is generally broken down into two categories: interpretable prediction models and post-hoc interpretation. Interpretable prediction models are those that are generated in such a way as to be immediately interpretable by a human [16, 34], though this often means that the models are simpler, and so may have reduced predictive power [34]. Some rule-based models, decision-tree-based models, decision lists, among other model types may fall into this category [5], though they may also become difficult to interpret if they grow too large [28].

Post-hoc interpretation refers to the interpretation of a model or a prediction after the creation of the model. It is suggested in [16] that post-hoc interpretation methods generally fall into one of three categories:

- Model explanation: Providing a global explanation of the entire black box model using an interpretable model that approximates and mimics it.

- Outcome explanation: Providing a local explanation for a prediction made using a specific input, or some small region of data surrounding a specific input.

- Model inspection: Providing a means of understanding some specific property or neighbourhood covered by the model.

This is partly expanded in [28], where post-hoc interpretation methods are classified based on both the scale and the purpose of the explanation. They determine that XAI methods could be used for the following:

- Local:
  - Prediction parts: Analyses and considers the effects of different components for a single prediction. LIME and SHAP, two methods commonly used in explainable PPA, fall into this category.
  - Prediction profile: Profiling variables and sensitivity of predictions for a single prediction.
  - Prediction diagnostics: Evaluating the performance of a black box model from the perspective of a single prediction or the region around the input of a prediction.
- Global:
  - Model parts: Analysing the importance of variables or groups of variables across the entire dataset, not just a single prediction.
  - Model profile: Profiling how a single variable or group of variables affect the response of a model.
  - Model diagnostics: Methods used to evaluate the performance of a model, improve prediction quality or otherwise present the structure of the model.

Another consideration that may be made when choosing a post-hoc interpretability technique is whether the technique is model-agnostic (i.e. able to interpret all black boxes) or model-specific (i.e. only able to interpret certain types of black boxes) [16].

Work by an IBM Research group [4] created a taxonomy of interpretable models that can be used to choose an approach based on the situations in which they may be used, and an open-source toolkit was developed based on this taxonomy, though it is targeted at data scientists, rather than business users. This taxonomy determines the appropriate method of prediction and interpretation based on the following:

- Whether the explanation needs to be static or interactive;
- Whether a global or local explanation would be more appropriate;
- Whether the model must be directly interpretable (i.e. a white box model) or interpreted post-hoc; and
- If the explanation must be local and post-hoc, whether it will be generated based on several sample cases used, or simply looking at the relevant features of a specific case to determine output; or
- If the explanation must be global and post-hoc, whether a surrogate model is required or whether a visualisation of the relevant parts of the model would suffice.

As such, given that the suitability of any interpretability method or explanation is dependent on the context and the needs of the end user, various considerations must be taken into account when considering the usefulness of interpretations, both at the functional level and the user level. Therefore, in addition to the interpretation method, the method of communicating the interpretation (i.e. the explanation) must also be considered.

2.3 Explanations and Stakeholder Communication

As per the definition in [16], explanations can be viewed as the interface between the machine and the human, and it is the user's needs that determines whether something is an explanation [19]. Explanations are classified in [6] as being either non-pragmatic (answering the question posed) or pragmatic (answering the question with respect to the audience's knowledge and expertise). Therefore, explanations can generally be evaluated both in context and decoupled from the context [19]. There has been a demonstrated link between the quality of the explanation and trust, with increased access to information generally associated with increased trust [10], though not in all circumstances [22, 59]. In addition to this, explanations are suggested to be necessary to facilitate learning, facilitate understanding of real-world phenomenon, improve knowledge structures and provide assurance in fairness and safety in predictions [6, 31, 34].

The explainee, their needs and experiences and the context in which the explanation is provided are all key considerations in developing an explanation. Users may require explanations of algorithms for many reasons, including to gain a better understanding of a system [19, 31], improve their ability to use a system [19], or have enough information to make a decision [25]. Explanations are highly contextual and the explainer must select and present the subset of causes that are the most useful and interesting to the explainee [31]. Moreover, trust may be situational [25], and explanations may be useful in allowing a decision-maker to understand the boundaries and weaknesses of a system [59].

As such, there has been emphasis on the importance of ensuring that explanations are customised based on the needs of the audience. It has been suggested that explanation methods, particularly interactive explanation methods in which the human element is able to query the explanation, should be done in a way that emulates human explanations [27]. In [44], it is suggested that an explanation method to be applied to should consider functional, operational, usability, safety and validation requirements necessary to the context, the audience and the predictor being explained. In a later paper [45], the authors suggest an efficient way of doing so is to allow the end user to provide feedback to the explainer in order to customise and personalise the explanation. As such, it has been suggested that end user perspectives must be considered in explanation research [10,45], and any determination of quality should consider the end user.

2.4 Evaluating Explanations

There are a number of ways to evaluate explanations, many of which depend on the purpose of the explanation and the explainee [33]. While explanation goodness can be evaluated without regard to context, a decontextualised evaluation will not necessarily provide an indication of explanation usefulness or user satisfaction [19]. A three-level system of evaluation that considers context to differing levels is proposed in [9], which comprises of:

- *Application-Grounded Evaluation*: Evaluating explanations in full context with end users;
- *Human-Grounded Evaluation*: Evaluating explanations with laypeople doing simple or simplified tasks; and
- *Functionally-Grounded Evaluation*: Using functional tasks to evaluate without subjective, user-based evaluation.

Functionally-grounded evaluation measures include those that do not require human input, but still allow some judgement of the explanation to be made. An example of such a measure would be computational efficiency [33]. Two common functionally-grounded evaluation measures in literature are *stability* and *fidelity*.

Explanation *stability* is defined as the level of similarity between explanations for similar or identical instances [52]. Several measures of stability for XAI have been proposed, but many are specific to a single explainable method, such as stability indices for LIME [52]. However, measures proposed to assess the stability of feature selection algorithms in feature engineering literature can be adapted to evaluate explanation stability where explanations involve feature importance attribution. The stability of these feature attribution methods can be evaluated in several ways, including by measuring the stability of a returned subset of features (*stability by subset*) and the stability of the measure of importance provided to each feature (*stability by weight*) [32]. These measures typically use sets of features or feature weights, and measure the stability of the sets by averaging the pairwise similarity for each possible pair in the set [32].

Explanation *fidelity* is generally defined as how faithful the explanation is to the black box. Two ways of measuring fidelity are defined in [30]: external and internal fidelity. *External fidelity* measures the similarity of decisions made by a surrogate model or interpretation of a black box and the black box itself [30]. While this approach can be used to measure how often a black box and its interpreter agree, it does not measure how well the interpreter mimics the black box's decision-making process – defined as *internal fidelity* [30]. Approaches that have been used to measure internal fidelity include:

- Creating explanations for the black box and surrogate models using an existing explainable method to determine how often the explanations concur [30];
- Removing or changing features identified by the explanation and comparing the changes in prediction probability of the black box [11, 21]; and
- Creating explanations for a white box model to see how well the explanations match the decision-making of the white box [39] (though this does not necessarily imply that the interpreter will perform as well as with a more complex model).

Non-functional evaluation measures generally rely on some response from a human user, whether that response be performance of a task or a subjective evaluation of the explanations provided. Examples of non-functional evaluation include:

- *Human simulatability*: How accurately the user is able to predict the black box's predictions by mimicking its decision processes [23, 44].
- *Task Effectiveness*: How well the user can apply the provided information, for example the performance of a particular task given an explanation [9, 53].

- *Time Efficiency*: The time taken by the user to complete a task given an explanation [23].
- *Mental efficiency*: How much effort the user had to expend to apply the explanation [53].
- *Satisfaction*: The user's evaluation of how well the explanation satisfies their needs [24, 33]
- *Trust*: The user's evaluation of the system's trustworthiness [57]

3 Research Problem

While there have been attempts to apply or create XAI methods to interpret and explain process predictions (as described in section 2.1), the fitness of these methods to explain PPA is unknown, and there are few frameworks available to evaluate this fitness. And although attempts have been made to create explainable interfaces and methods specifically for PPA [14, 55], it is as yet unclear how to best ensure that PPA interpretations are useful and informative. Therefore, the proposed research problem for this project is: How can explainable methods be best utilised for predictive process analytics, particularly in supporting decision making? This can be broken down into the following research questions:

- **RQ1**: What AI-enabled models and techniques can be used to present explanations to users in the context of process prediction?

  As seen in section 2.2, there are a number of interpretation methods available, all intended for different purposes and functioning in different ways. As such, it becomes necessary to identify interpretability methods and tools relevant to the datasets and methods used in PPA. Therefore, the state-of-the-art must be first understood, and XAI methods that are commonly applied to PPA or are applicable to PPA must be identified.

- **RQ2**: What metrics and/or criteria would be suitable to assess the quality of process prediction explanations generated from the models and techniques identified in RQ1?

  While a number of evaluation frameworks and taxonomies exist for evaluating interpretability and explainability tools, they are highly generalised and act as broad categorisations, rather than an evaluation method. For example, in [44], while a number of dimensions of categorising and evaluating XAI are presented, there are no specific metrics and methods that can be applied for evaluation. As such, it becomes necessary to define metrics that are relevant to the datasets, methods and explanations that are used in PPA, with an emphasis on creating an evaluation framework that is extensible and flexible enough to account for the different prediction problems, datasets, explanation types and users that may be involved.

- **RQ3**: Using the metrics and/or criteria determined in RQ2, what gaps and drawbacks do existing explainable methods have when explaining process predictions?

  - **RQ3.1:** At a functional level, using the interpretability methods identified as part of RQ1, what gaps, disadvantages and drawbacks exist when explaining process predictions?

  - **RQ3.2**: From the perspective of a user, using the interpretability methods identified as part of RQ1, what gaps, disadvantages and drawbacks exist when interpreting process predictions?

  RQ3 is necessary in order to better understand how fit for purpose the identified XAI methods are for PPA. Firstly, functionally-grounded evaluation is necessary in order to determine how well-suited an XAI method inherently is to solving the problem of explaining PPA, even before users are considered. Following this, it is important to understand how the explanations provided by the XAI methods are received by decision makers. As such, human-grounded or application-grounded evaluation becomes necessary to determine the usefulness of explanations.

- **RQ4**: How could explainable methods be utilised in order to more effectively interpret process predictions?

  Once evaluation has been completed and gaps and drawbacks in the chosen interpretation methods are identified, guidelines and/or recommendations can be developed to ensure that these effects of these gaps and drawbacks are minimised or eliminated altogether. This research question addresses the ultimate aim of this project – understanding and ensuring that process predictions can be effectively explained to support human decision-making.

The following section (section 4) will explain in detail the proposal for addressing these questions.

4 Program of Research

4.1 Objectives

The proposed research project attempts to understand how to best design explainable methods to promote human understanding of process predictions. More specifically, the aim of the project is to empower the decision-making of individuals in an organisational setting by effectively interpreting and explaining PPA models. The objectives of this research project are threefold to address the research questions specified in section 3:

- **Objective 1**: Investigate how process predictions can be explained to a human audience, and how the goodness of these explanations can be judged (RQ1 and 2)
- **Objective 2**: Evaluate relevant existing explainable methods to determine how fit-for-purpose they are in explaining process predictions (RQ3)
- **Objective 3**: Determine how to best create explanations to support decision-making regarding PPA (RQ4)

4.2 Chosen Methodology

Design Science Research (DSR) will be used to guide the methods used in the proposed project. Hevner et al. [18] define DSR as a problem solving paradigm that creates innovations that can be used to formalise practices, ideas and products, and in doing so facilitate the effective and efficient creation, usage and maintenance of information systems in businesses. They further clarify that the environment of the context and the business determines the business needs (problem space) to be explored by research [18], which, for the proposed research project is the research problem identified in section 3: determining how best explainable methods can be used to most effectively interpret predictive process analytics. The researcher's knowledge base is expected to provide the "raw material" required to design and/or build an appropriate artefact or theory, and this knowledge base will consist of foundational knowledge, such as theories and frameworks, as well as methodologies [18]. Existing explanation models and methods will be used as the knowledge base for the proposed project.

In a later paper, Hevner [17] identifies three research cycles in DSR, which connect the three facets of environment, knowledge base and the research itself together in an iterative way (see figure 1):

- **The Relevance Cycle** provides the problem domain for which the artefact or theory is designed, as well as the acceptance criteria for this output. The output must also be studied and evaluated within the context of this problem domain to determine its effectiveness in addressing the research problem and decide whether another iteration of the relevance cycle is necessary to improve the output. In the proposed project, the problem domain is predictive process analytics, and the outcomes of the project (explanations created using XAI methods) will be evaluated within the context of process predictions.
- **The Rigor Cycle** provides the knowledge required to create and innovate the output of the research, including the knowledge required to create the output, as

well as the knowledge required to evaluate the output. An effective DSR project will further add to the knowledge base, either by extending existing constructs or providing new ones. In the proposed project, existing prediction and explainability methods and models will be used to create process prediction explanations. The evaluation framework that will be used to assess these explanations, the results of the evaluation and the adaptations made, as well as any new constructs created, will be added to the knowledge base.

- **The Design Cycle** is described as, "...the heart of any design science research project," [17] where the output is continuously created, evaluated and adapted based on evaluation. The creation of the output and its evaluation are both rooted in the relevance and rigor cycles. This cycle will form the major part of the proposed project, where explainability models and methods will be created, adapted and evaluated.

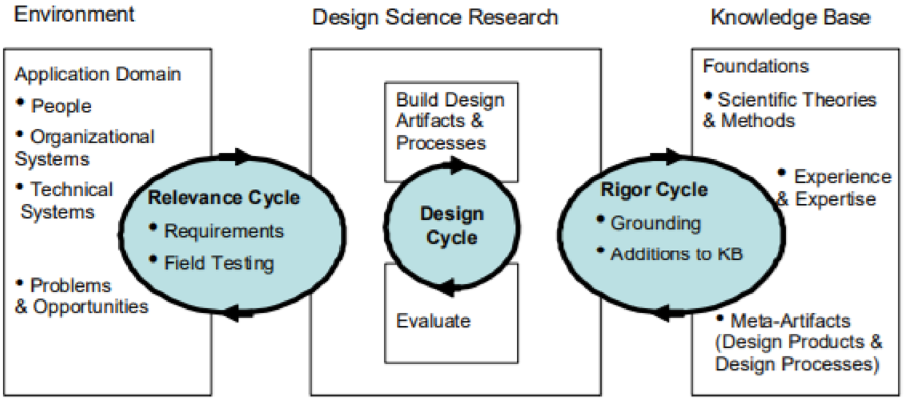


Figure 1: The three cycle view of DSR [17]

4.3 Research Plan

The proposed project will be split into three phases. The first phase will cover RQs 1 and 2, and focus on understanding the state of the art in explanations, evaluating explanations and the relevance of both to the problem domain. The second phase covers RQ3, where existing explanation methods and models will be evaluated for fitness within the problem domain. The third phase will cover RQ4 and will focus on creating a set of guidelines to ensure explanation quality for PPA interpretations.

Peffers et al. [37] provide a specific, iterative process for conducting DSR (see figure 2). This process begins with the identification of the problem, followed by the formulation of objectives. Then, an artefact can be designed and/or built, and tested in context, allowing for an evaluation of the artefact. The resulting knowledge is then communicated. The researcher can iterate back to an earlier stage as necessary to change the objectives or the output. The application of this process in the proposed project is outlined in table 1. A more complete timeline is presented in section 6.1.

Table 1: Peffers et al.'s DSRM process, as applied to the proposed project

Phase	**DSRM Phase**	**Activities**
Phase 1	Problem Identification and Motivation	• Investigation of interpretability of PPA • Investigation of existing AI explanation models and methods • Investigation of explanation needs in PPA
	Define Objectives of Solution	• Investigation of characteristics needed for process prediction explanations • Investigation of evaluation measures of XAI
	Design and Development	• Creation of an evaluation framework to assess explanations of process predictions • Proposal of evaluation metrics to assess explanations of process predictions
Phase 2	Demonstration	• Selection of appropriate context for investigation, including appropriate domains, datasets and contexts captured by the datasets • Creation of process prediction explanations using XAI methods identified in Phase 1
	Evaluation	• Evaluation of existing AI explanation models and methods using framework created in Phase 1 • Identification of gaps and drawbacks in the generated interpretations • Identification of specific needs of user groups
Phase 3	Design and Development	• Creation of design principles based on findings from Phase 2 • Adaptation and/or extension of existing explainable methods and models
	Demonstration	• Implementation of adapted and/or extended models and methods to the chosen datasets and contexts
	Evaluation	• Evaluation of adapted and/or extended AI explanation models and methods using framework created in Phase 1
Throughout	Communication	• Identify and publish results as appropriate.

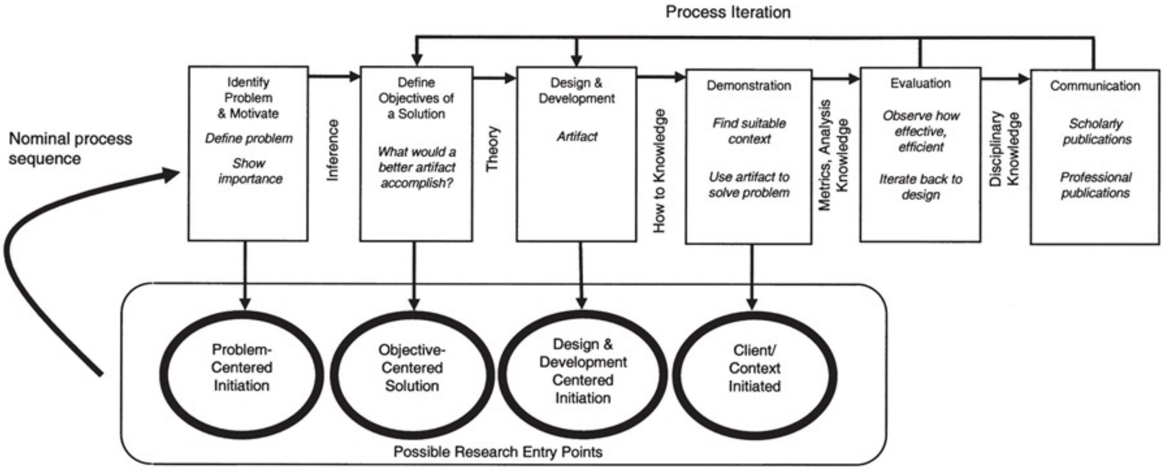

Process Iteration
Nominal process sequence
Identify Problem & Motivate
Define problem
Show importance
Inference
Define Objectives of a Solution
What would a better artifact accomplish?
Theory
Design & Development
Artifact
How to Knowledge
Demonstration
Find suitable context
Use artifact to solve problem
Metrics, Analysis Knowledge
Evaluation
Observe how effective, efficient
Iterate back to design
Disciplinary Knowledge
Communication
Scholarly publications
Professional publications
Problem-Centered Initiation
Objective-Centered Solution
Design & Development Centered Initiation
Client/ Context Initiated
Possible Research Entry Points

Figure 2: The process for design science research as provided by Peffers et al. in [37]

4.4 Resources and Funding Required

There will be some specialised equipment or other resources required for this PhD project. To ensure that results can be replicated and verified by other researchers, open source datasets will be used for creating predictions to be explained. These datasets will involve real-life event logs of process executions made publicly available for research purposes, such as those provided by the 4TU Centre for Research Data[1]. Access to QUT's High Performance Computing environment has already been obtained for the purpose of running evaluations, and access to the Research Data Storage Service will be requested for the purposes of storing any interview and survey data that will be collected.

Some specialised software or existing code will be necessary for processing and using the datasets, but open source software and code can be used for this. ProM is an open source process mining tool that can be used to view, analyse and edit event logs, and will be of use in extracting process information from and pre-processing the dataset used. Open source code created by researchers can be used to train, interpret and create explanations for a PPA model, likely using a Jupyter Notebook environment.

4.5 Individual Contribution to the Research Team

There are currently multiple PhD projects aiming to address different aspects of PPA. One project will investigate the adaptation of event log datasets for PPA, while another project will investigate how to best utilise interpretation techniques in conjunction with data processing and prediction models to provide useful interpretations that can then be used to improve the prediction model and/or data used. The project proposed in this document aims to investigate how to best utilise explainable techniques – both post-hoc methods and white box methods – to support decision-making.

It is expected that the following outcomes will be achieved as a result of this project:

- Proposal of evaluation metrics and criteria for evaluating explanations and expla-

[1] https://researchdata.4tu.nl/en/

nation methods of process predictions;

- Evaluation of existing explainable methods, both functionally-grounded and evaluated by user testing;
- An adaptation or extension of existing interpretation methods based on gaps identified; and
- A set of guidelines for developing explanations for PPA.

5 Progress To date

5.1 Techniques Used

Process Analytics A business process is defined as, "...a collection of inter-related events, activities and decision points that involve a number of actors and objects, which collectively lead to an outcome that is of value to at least one customer." [12] To generate process prediction models, events logs are used. An event log contains a sequence of activities recorded by an information system, with each activity (or event) in the event log belonging to a particular case and having a number of associated attributes, such as a time stamp or the ID of the resources (i.e, person, machine or system) that performed the activity. Some attributes of an event may be static and do not change over the course of the case (for example, the Case ID), and some may be dynamic (for example, the resources who performed each event or the timestamp of each event).

When creating machine learning models, event logs need to be processed in order to make the data within an event log more suitable for prediction algorithms. Two forms of processing commonly used in PPA are data bucketing and data encoding. The following bucketing and encoding combinations were used in the evaluations that have been conducted:

- Aggregate encoding for dynamic attributes with no bucketing
- Aggregate encoding for dynamic attributes with prefix-length-based bucketing
- Index-based encoding for dynamic attributes with prefix-length-based bucketing

Bucketing refers to the grouping of data, where each group - or bucket - is used to train one classifier. When the "no bucketing" method is used, all data is compiled as one (i.e. in a single bucket) before being used to train a single classifier. When "prefix-length bucketing " is used, data is bucketed based on the number of prefixes in a trace (i.e. number of events in the case) and a single classifier is trained for each prefix length. For example, in process where the smallest trace is one event, and the largest trace is 25, 25 buckets of data, each of a specific prefix length, will be generated, and 25 classifiers will be trained.

Typically, static and dynamic attributes of events are encoded differently. Static encoding is used in all of the bucketing and encoding combinations listed to encode static attributes. In this type of encoding, numeric attributes are encoded as they are, and one-hot encoding is used for categorical attributes. Aggregate encoding and index-based encoding were used for dynamic attributes. When using the former, the frequency of occurrence across a case (instance of the process) is used to record information about dynamic categorical attributes (for example, how many times a particular activity occurred or a resource was involved in the process) and the minimum, maximum, mean and standard deviation of values for dynamic numerical attributes across one instance are all used as features for that instance. When using index-based encoding, at each index (prefix in the process trace), numeric attributes are encoded as-is and categorical attributes are one-hot encoded.

Outcome-oriented prediction was the prediction problem that was chosen for testing due to its relative simplicity and prevalence in literature.

Machine Learning The machine learning technique chosen for initial evaluations was XGBoost. XGBoost is a decision-tree based algorithm, which creates multiple decision trees in sequence, with each subsequent tree accounting for the weakness of previous trees (a method known as “boosting”) [7]. While numerous machine learning techniques have been applied to PPA [29,46,47,50], XGBoost was chosen for the initial evaluations as it was shown to be the most effective for outcome-oriented predictions [47].

Interpretability Techniques Two popular interpretation methods in PPA literature are LIME and SHAP [14,41,43]. Both techniques are model-agnostic and produce local interpretations that fall into the ”prediction parts” category, though the methods they use are different. LIME creates perturbations of the dataset and uses the predictions of the black box model on these perturbations as ground truth to create a surrogate linear model that captures the black box model’s behaviour at a specific neighbourhood. This surrogate model is then used to weight and rank the importance of each feature in producing the result of the original instance, and this ranking and associated weights are presented as an explanation [39]. An example of such an explanation is presented in figure 3.

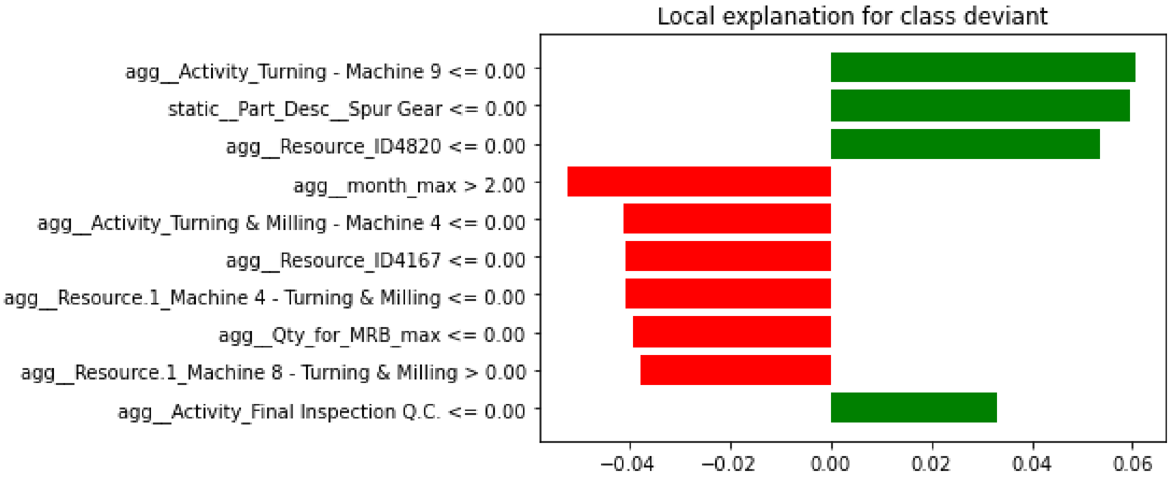


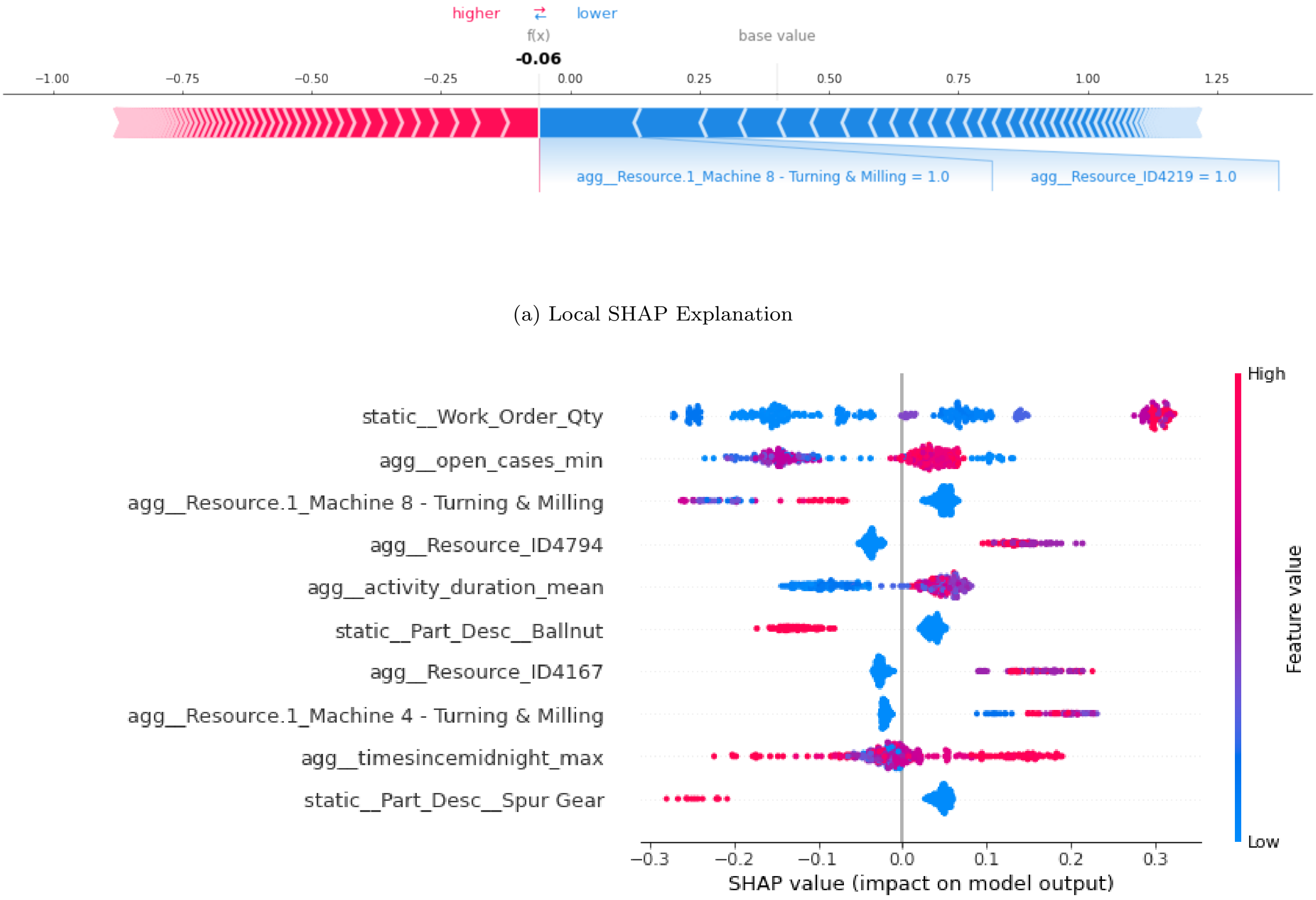
Figure 3: An explanation generated by LIME for a single instance in the Production dataset.

By contrast, SHAP uses a game theoretic approach. When using SHAP, a value known as SHAP value is assigned to each feature in an instance, describing its contribution to the final model output (i.e. a prediction) [26]. A notable feature of SHAP is that the SHAP values of each feature across all instances can be aggregated to provide a global explanations of the black box. An example of both a local and a global SHAP explanation is provided in figure 4.

5.2 Proposed Evaluation Framework and Metrics

5.2.1 Proposed Evaluation Framework

Based on past works that have attempted to evaluate XAI (see section 2.4), an interpretability framework for evaluating XAI has been developed (table 2). Specific metrics have only been defined for functionally-grounded measures.



(a) Local SHAP Explanation

(b) Aggregated SHAP values (global explanation)

Figure 4: A local explanation generated by SHAP for a single instance in the Production dataset (a) and a global explanation generated by aggregating SHAP values for the Production dataset (b).

5.2.2 Proposed Evaluation Metrics

It is important to note that the definition of each measure in this section focuses on the local explanations at the process instance level for any given event log dataset. Multiple explanations are required for each individual process instance before evaluation can occur. The overall evaluation of each measure for each classifier can be calculated as the average of the evaluations score for all individual process instances in the log.

Explanation Stability will be evaluated in terms of both stability by subset and the stability of feature weights. The *stability by subset* measure was adapted from a measure proposed in [35] for evaluating the stability of feature selection algorithms, and determines stability based on the presence or absence of each feature across multiple subsets. The stability of feature subsets ($\phi(\mathcal{Z})$) for a single process instance in an event log can be calculated as follows:

$$\phi(\mathcal{Z}) = 1 - \frac{\frac{1}{d}\sum_{i=1}^{d} s_{f_i}^2}{\frac{k}{d}(1 - \frac{k}{d})} \tag{1}$$

Table 2: Proposed Evaluation Framework for Explainable PPA.

Evaluation Level	Evaluation	Measure
Functionally-Grounded Evaluation	**Stability**	Stability by Subset
		Stability by Weight
	Fidelity	Internal Fidelity
	Efficiency	Time Required
Human-Grounded and Application-Grounded Evaluation	**Comprehension**	Human Simulatability
		Task Effectiveness
	Usability	Time Efficiency
		Mental Efficiency
	Sastisfaction	User Satisfaction
		User Trust

where:

- d = number of features encoded from event attributes in the log
- M = number of explanations generated for the process instance
- k = number of most relevant features, where relevance or level of importance is determined by an explanation generated for the process instance
- $\overline{k}$ = average number of features selected across all M explanations for the process instance
- $s^2_{f_i}$ = sample variance of the presence of feature f_i across all M explanations for the process instance
- $\mathcal{Z}$ = binary matrix of size M x d. Each row of the binary matrix represents a feature subset, where a 1 at the i^{th} position means feature f_i is present in the explanation and a 0 means it is not present.

This measure is bounded between 0 and 1, where 0 indicates no similarity in the feature subsets, and 1 indicates that all subsets are identical.

Pearson's correlation coefficient is generally used to measure stability of feature weights in feature selection algorithms [32], but this measures only a general trend of importance, and does not quantify how much a feature's weight may vary. Therefore, the statistical measure of relative variance was adapted to measure *stability by weight*. The stability of feature weights ($\phi(\mathcal{W})$) for a single process instance in an event log can be calculated as follows:

$$\phi(\mathcal{W}) = 1 - \frac{1}{d}\sum_{i=1}^{d} \frac{\sigma^2_{w_i}}{|\mu_{w_i}|} \qquad (2)$$

where:

- d = number of features encoded from event attributes in the log
- M = number of explanations generated for the process instance

- μ_{w_i} = mean of the weights of feature f_i across all M explanations for the process instance
- $\sigma^2_{w_i}$ = variance of the weights of feature f_i across all M explanations for the process instance
- $\mathcal{W}$ = matrix of size M x d. Each row of the matrix records the weight of each feature as quantified by an explanation

This measure also has an upper bound of 1 (indicating perfect stability), but no lower bound.

Explanation Fidelity will be evaluated by the *internal fidelity* of explanations, to ensure that it is the interpreter's decision-making, not the decision itself, is being evaluated. In literature, error functions are generally used to quantify internal fidelity, averaged out over the size of a dataset [11]. As such, the mean absolute percentage error (MAPE) can be used to measure the fidelity of explanations, and the fidelity ($\mathcal{F}$) of the interpreter for a single process instance in an event log can be calculated as follows:

$$\mathcal{F} = \frac{\sum_{1}^{|X'|} \frac{|Y(x)-Y(x')|}{Y(x)}}{|X'|} \tag{3}$$

where:

- x = original feature vector for the process instance
- X' = Set of perturbations for x and $x' \in X'$
- $Y(x)$ = Prediction probability for the predicted class given input x
- $Y(x')$ = Prediction probability for the originally predicted class given input x'

This measure is naturally bounded by the fact that prediction probabilities fall between 0 and 1.

5.3 Implementation of Framework

5.3.1 Evaluating Stability

To measure the stability, a sample of the test set was first selected, and ten explanations were generated for each instance (i.e. $M = 10$ for each instance). Stability scores are first calculated for each instance, then averaged out over the dataset to calculate the stability of the entire dataset.When computing stability by subset, features in the explanation are ranked based on the importance value attached to that feature (the importance score when using LIME and SHAP values when using SHAP), and the ten most influential features are chosen as the subset to be measured using Equation 1 (i.e. $k = 10$ and $\overline{k} = 10$). Equation 2 is used to measure stability by weight, and the quantified feature importance provided in each explanation for all features (not just a subset of features) are used to calculate stability.

5.3.2 Evaluating Fidelity

Of the three internal fidelity evaluation approaches outlined in section 2.4, the second one is the most appropriate for this context. The first approach was developed to evaluate the custom-made surrogate models, not existing methods, while the third one attempts to determine fidelity by explaining a relatively simple white box model and determining the explanation correctness. However, this does not necessarily imply that the explanation will also be correct for a more complex black box model. As such, the second method of evaluating fidelity by altering or eliminating features based on an explanation is the most appropriate approach in this context.

While an ablation approach is the most common approach in literature [11, 21], this is generally used for image or textual data where features (pixels in images or words in textual data) can simply be removed from the input. However, for tabular data like event logs, this will not be possible, as a machine learning algorithm will require a feature vector input with same length as instances in the training set. In some cases, such as with XGBoost, the prediction model may automatically impute missing data if a "blank spot" appears in the data. Therefore, altering features (a perturbation strategy) will be more effective to measure internal fidelity when using event logs. Once again, ten explanations were generated for each instance – to account for instability – and the top 10% of features (i.e. 10% of d) that appeared the most often across all ten explanations were identified.

For each feature, the feature value distribution that caused the initial prediction result was identified using the explanations. This was relatively simple with LIME, which presents the feature value or feature value distribution that caused the black box's prediction (see figure 3). For example, an explanation including "1 < Activity_A < 3" indicates that the occurrence of activity A more than once, but fewer than three times was influential on the result.

However, SHAP presents only the feature's influence on the end result as a SHAP value. Therefore, the distribution had to be generated by determining what feature values created SHAP values similar to the SHAP value for the instance under examination. This was done by aggregating SHAP values across the dataset (i.e. creating a global explanation), then identifying the feature values across the entire dataset that produced a SHAP value similar to the explanation for the original instance (i.e. identifying feature values for a small size of the global explanation). The minimum and maximum values in across these feature values became the minimum and maximum of the distribution. For example, if "Activity_A" has a SHAP value of 0.54, using the global explanation, all instances where "Activity_A" has a SHAP value between 0.5 and 0.55 would be identified, and the maximum and minimum values for "Activity_A" across these instances would be used as the distribution.

Once the relevant feature value distributions were identified for all relevant features, for each instance:

1. A prediction using the original input vector x was generated, along with the prediction probability for the predicted class ($Y(x)$)

2. For each feature to be perturbed, a new, uniform distribution outside of the existing distribution was created to draw new feature values from (for example, if the initial distribution was $1 < Activity\_A <= 3$, new feature values from "Activ-

ity_A” would be drawn from between 3 and 5)

3. For each feature to be perturbed, a new value was randomly sampled from the new distribution to replace the original value for that feature to create the perturbed feature vector x'
4. The prediction probability for the originally predicted class was determined for input x' (i.e. $Y(x')$)
5. The difference between $Y(x)$ and $Y(x')$ was computed

Ten perturbations were created for each instance (i.e. $|X'| = 10$), and the differences in prediction probability between the original instance and each of the ten perturbed instances were used to calculate the MAPE for each instance (see Equation 3).

5.3.3 Datasets Used

The evaluation was conducted with three open-source, real-world event logs that were used in a predictive process monitoring benchmark on outcome prediction, following the preprocessing, training and testing methods used in that paper [47]. All three event logs have different contexts, are different in size and have different types of attributes present as shown in table 3.

Table 3: A summary of statistics of three event log datasets

Event Log		Production[2]	Sepsis Cases[3]	BPIC2012[4]
Description		A manufacturing process	Hospital event log showing sepsis cases	Loan application process
No. of Cases (before encoding)		220	782	4,685
Proportion of Positive Cases		55.0%	16.0%	53.4%
Maximum Prefix Length		23	29	40
Prefix Lengths Used		1 – 20	1 – 25	1 – 25
Feature Vector Shape	**Single Bucket & Aggregate Encoding**	162	274	134
	Prefix-length buckets & aggregate encoding	Min: 137 Max: 156	Min: 153 Max: 218	Min: 43 Max: 134
	Prefix-length buckets & Index-Based Encoding	Min: 100 Max: 844	Min: 147 Max: 535	Min: 11 Max: 1654

The outcomes to be predicted are whether a case is “deviant” (positive) or “normal” (negative). In the Production process, a deviant case is one where at least one work order in the case is rejected, while in the Sepsis Cases process, a deviant case is one where a patient returns to the ER within two weeks of discharge. Because this dataset is also highly unbalanced (only 16% of cases are deviant), the data was downsampled before training. As this process contains considerably more static than dynamic attributes, feature vector lengths were longest when using aggregate encoding, but comparatively shorter feature vectors for longer prefix lengths when index-based encoding is used. By

[2] https://doi.org/10.4121/uuid:68726926-5ac5-4fab-b873-ee76ea412399

[3] https://doi.org/10.4121/uuid:915d2bfb-7e84-49ad-a286-dc35f063a460

[4] https://doi.org/10.4121/uuid:3926db30-f712-4394-aebc-75976070e91f

contrast, BPIC2012 dataset has only a single static attribute, but a significant number of dynamic attributes (many of which are categorical), and so will have relatively short feature vectors when using aggregate encoding, but longer feature vectors at high prefix lengths when using index-based encoding. A deviant case in this process is one where a loan isn't accepted.

5.4 Outcomes of Implementation

The source code implementing the proposed metrics and approach, and the results of the initial experiments are available at: https://git.io/JIYtH.

5.4.1 Results

Once the evaluations were conducted as per the approach outlined in section 5.3, the results were averaged out across the dataset in order to provide an overview of the performance of the interpreters for each combination of dataset and bucketing and encoding.

Table 4: Overall stability by subset results for each dataset

		Production	**Sepsis Cases**	**BPIC 2012**
Single bucket	LIME	0.72	0.20	0.81
Aggregate Encoding	SHAP	**1.00**	**1.00**	**1.00**
Prefix-length buckets	LIME	0.82	0.50	0.57
Aggregate Encoding	SHAP	**1.00**	**1.00**	**1.00**
Prefix-length buckets	LIME	0.61	0.33	0.24
Index-based encoding	SHAP	**1.00**	**1.00**	**1.00**

Table 5: Overall stability by weight results for each dataset

		Production	**Sepsis Cases**	**BPIC 2012**
Single bucket	LIME	0.95	0.36	0.85
Aggregate Encoding	SHAP	**1.00**	**1.00**	**1.00**
Prefix-length buckets	LIME	0.89	0.63	0.67
Aggregate Encoding	SHAP	**1.00**	**1.00**	**1.00**
Prefix-length buckets	LIME	0.80	-0.04	0.30
Index-based encoding	SHAP	**1.00**	**1.00**	**1.00**

Stability Overall, SHAP is significantly more stable than LIME (see tables 4 and 5), with perfect stability. LIME is least stable when using index-based encoding and most stable when using aggregate encoding, but the most effective bucketing techniques vary. For BPIC2012, single bucketing produces the best results, while prefix-length bucketing yields the best result for Sepsis Cases. LIME is almost always most stable for the Production dataset.

Fidelity Fidelity was generally poor for both LIME and SHAP (see table 6). Though SHAP explanations were generally more faithful, the difference between SHAP's and LIME's fidelity scores were generally small. Both interpretation techniques generally fare the worst when aggregate encoding is used with no bucketing, but the fidelity of the interpreters when using the other two combinations varied across datasets. When using the BPIC2012 event log, prefix-length bucketing and aggregate encoding produced the best results, while prefix-length bucketing and index-based encoding produced the best results for the Sepsis dataset. LIME performed better with the former on the Production dataset, but SHAP performance was on par for both.

Table 6: Overall fidelity results for each dataset

		Production	**Sepsis Cases**	**BPIC 2012**
Single bucket	LIME	0.26	0.36	0.37
Aggregate Encoding	SHAP	**0.27**	**0.46**	**0.41**
Prefix-length buckets	LIME	0.47	0.37	0.38
Aggregate Encoding	SHAP	**0.51**	**0.49**	**0.42**
Prefix-length buckets	LIME	0.36	0.51	0.32
Index-based encoding	SHAP	**0.51**	**0.56**	**0.4**

5.4.2 Analysis and Findings

In this section, the dataset-level results from the previous section will be unfolded to better understand how the characteristics of each instance affected the quality of the explanations (i.e. the initial results provided by the equations defined in section 5.2.2). The instance-level characteristics we explore the effects of are the prefix length of the instance and the prediction probability for the original input vector. The results are, once again, separated by the bucketing and encoding techniques used.

Running Time The running time of the interpreters for each combination of dataset, bucketing and encoding was recorded. SHAP was the faster interpreter, generally producing explanations in less than a second. LIME's running time was reasonably consistent when using aggregate encoding, but increased with prefix length when index-based encoding was used (see figure 5). It is apparent from these results that the characteristics of the dataset, including bucketing and encoding techniques, had an effect on running time, especially when using LIME. Feature vector length, in particular, seems to have affected LIME, which is most obvious when considering the running time for index-based encoding where running time increases considerably as the feature vector's length also increases. SHAP seems to be able to better handle this complexity.

Stability There is a clear relationship between instance-level characteristics and the stability of LIME. As with running time, SHAP is almost perfectly stable for all combinations of dataset and bucketing and encoding, and unfolding the results of SHAP produces no new insights. However, when unfolding the results, it becomes clear that the stability of LIME is dependent on the size of the feature vector (see figure 6). This is likely because of LIME's permutation of new feature vectors that are used to generate a surrogate model, which are randomly sampled from the neighbourhood of the input x [39]. However, as the size of the feature vector increases, LIME's sampling efficiency decreases, and the samples become less representative, and so the surrogate models generated is different every time a new explanation is created, causing instability [52]. This instability is more prominent when measuring stability by subset, but instability of weights is also present, and and is most apparent when using index-based encoding (see figure 7).

However, contrary to the observations regarding feature vector length and instability, when using index-based encoding the Production dataset is more stable than the Sepsis

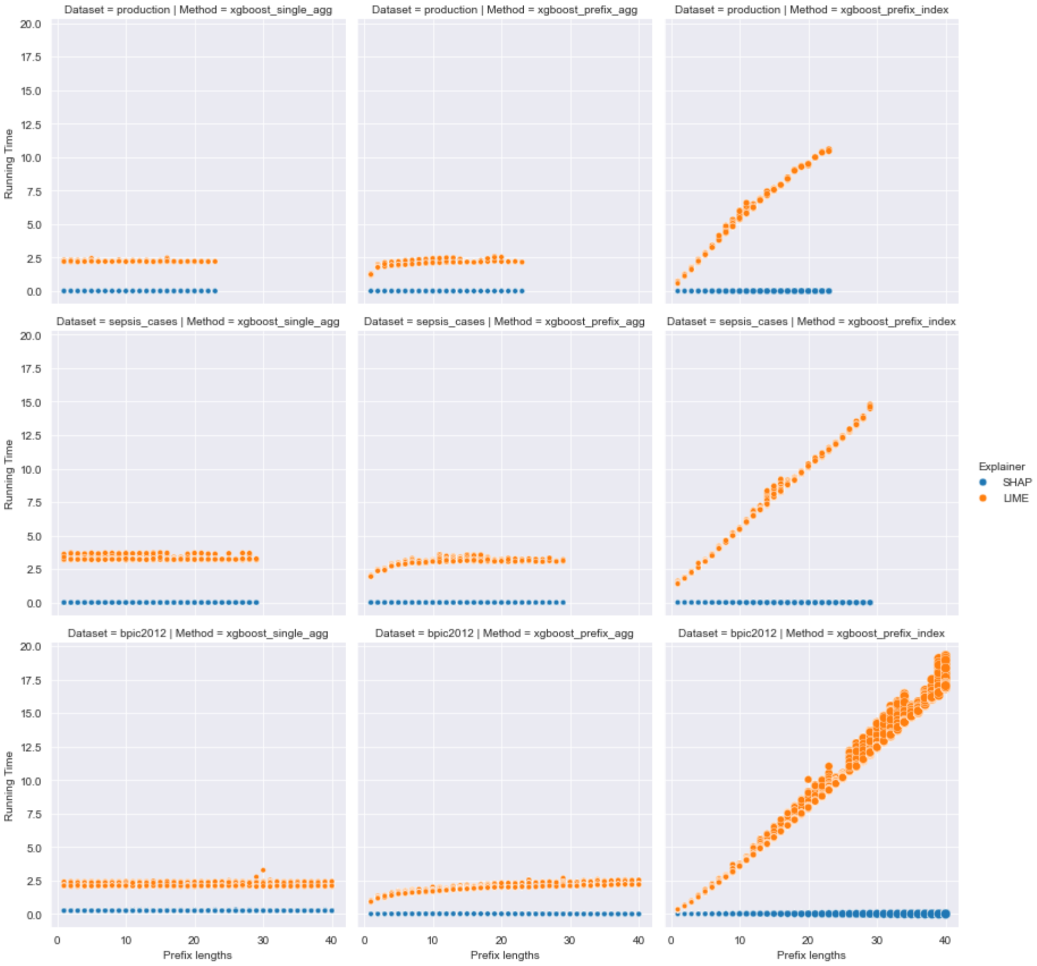
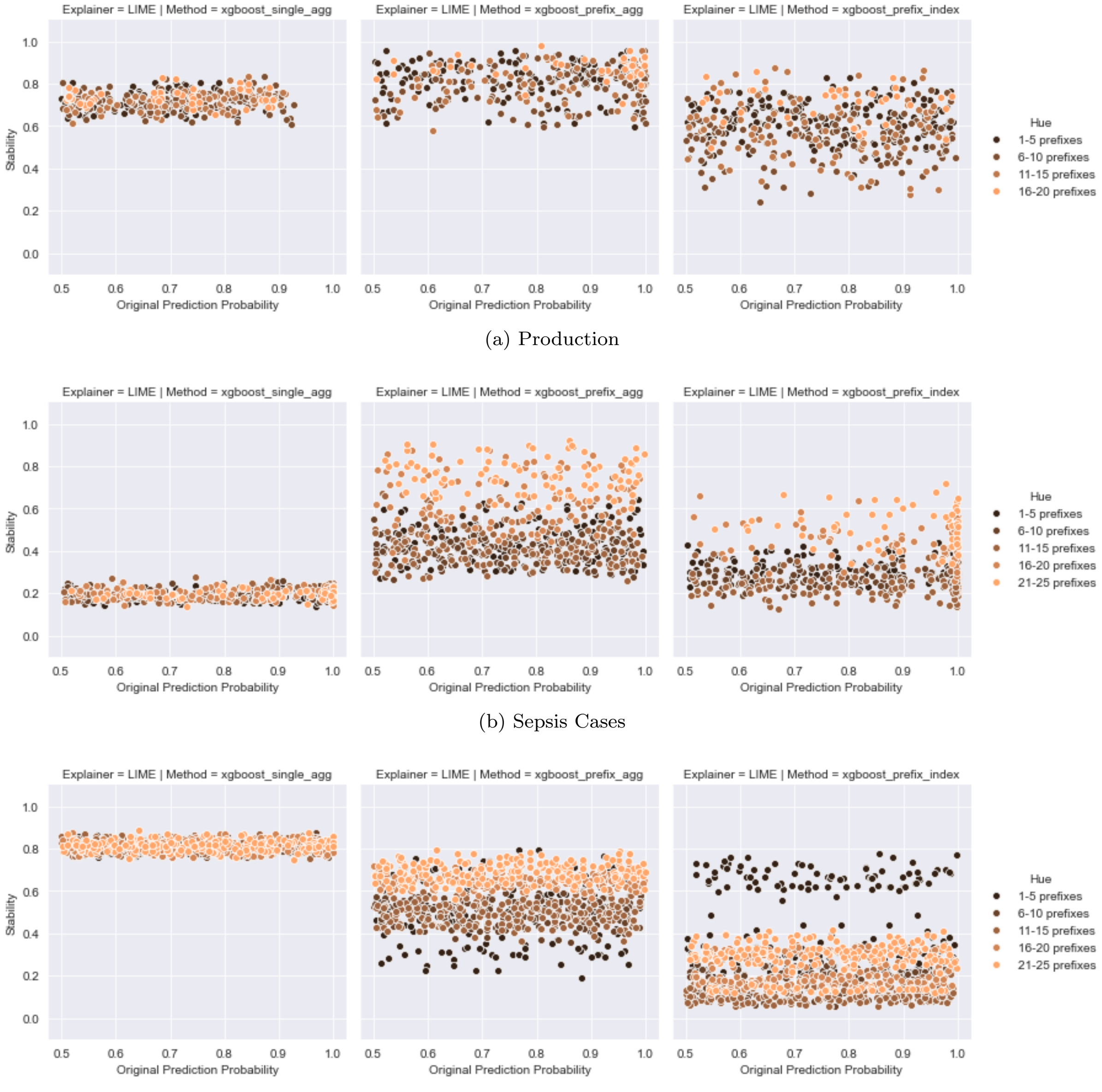


Figure 5: Time taken to create explanations (marker size indicates size of feature vector)

dataset though it has longer prefix lengths. This is likely because of the dataset's sparsity in each bucket. Once LIME has generated a set of input points X' based on x, it uses the black box to create a set of labels $Y(x')$ that are used to fit the surrogate model. Since the black box classifiers for the Production dataset are overfit when using index-based encoding (testing set accuracy is 61.8% and training set accuracy is 92.5%), it's likely that $Y(x')$ were near-randomly chosen by the black box. This is likely to have resulted in an under-fit surrogate model that changed very little each time a new explanation was generated.

Fidelity LIME and SHAP show similar results for fidelity. In many cases, the faithfulness of explanations varies so much across instances that the fidelity scores are almost uniform in distribution (for example, the Production dataset when prefix-length bucketing is used or with SHAP for the BPIC2012 dataset, see figure 8). This suggests that some explanations are faithful and others are not, but there is no pattern or trend associated with the instance's prefix length or initial prediction probability, and there is no consistency across bucketing or encoding methods or datasets. Some combinations (for example, Sepsis Cases with index-based encoding for both LIME and SHAP) have high overall fidelity scores, but once unfolded, it becomes apparent that is caused by a handful of highly faithful explanation and that for the most part, changes in prediction

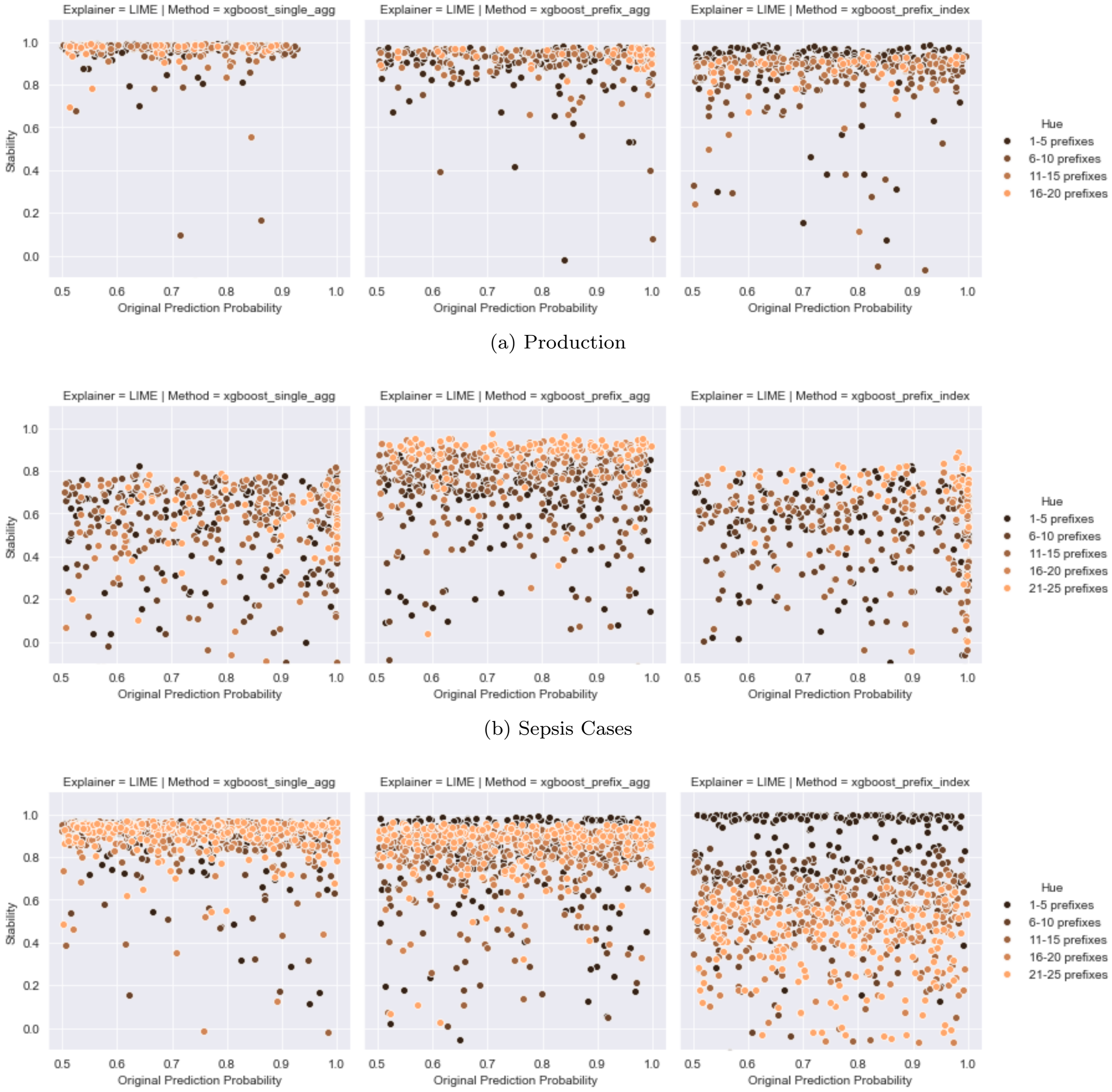


(a) Production

(b) Sepsis Cases

(c) BPIC2012

Figure 6: Stability by subset results over original prediction probability and prefix length of LIME, respectively, for Production (a), Sepsis Cases (b) and BPIC2012 (c)

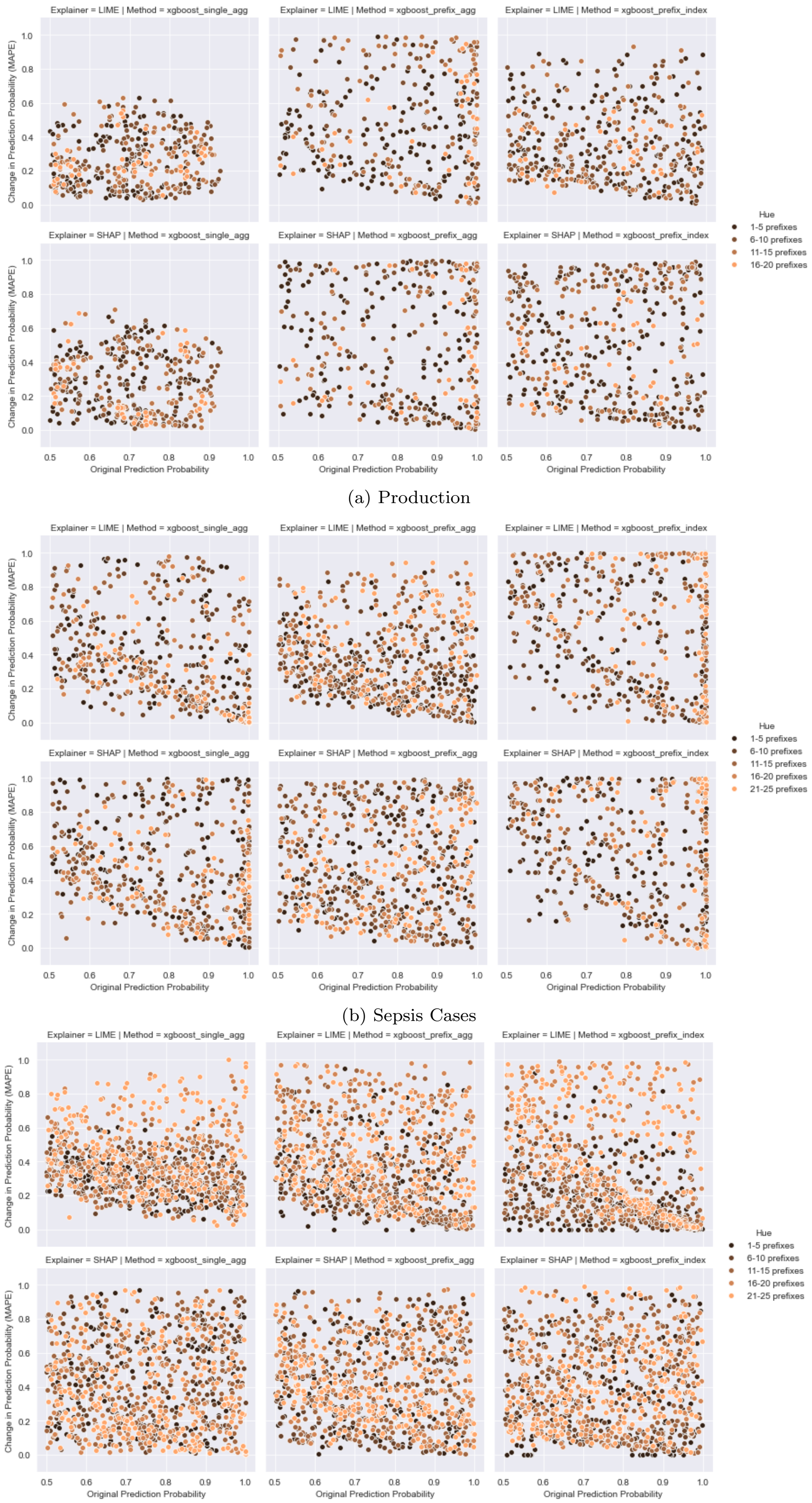


(a) Production

(b) Sepsis Cases

(c) BPIC2012

Figure 7: Stability by weight results over original prediction probability and prefix length of LIME, respectively, for Production (a), Sepsis Cases (b) and BPIC2012 (c)



(a) Production

(b) Sepsis Cases

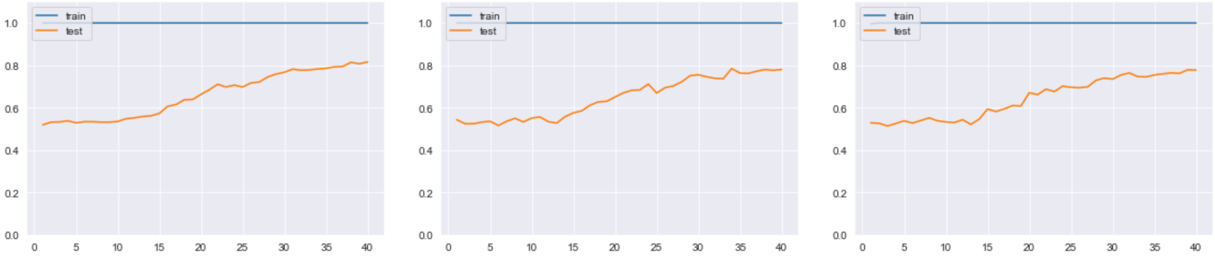
(c) BPIC2012

Figure 8: Fidelity results over original prediction probability and prefix length of LIME and SHAP, respectively, for Production (a), Sepsis Cases (b) and BPIC2012 (c)

probability were small.

One notable observation is that both LIME and SHAP are least faithful when aggregate encoding is used without any bucketing. This combination of bucketing and encoding has been shown to produce the most accurate black boxes for outcome prediction [47], so it may be difficult to apply LIME and SHAP in practice unless other bucketing and encoding techniques are reasonably accurate for that dataset and black box. But it is important to note that while other combinations yield more faithful explanations, fidelity is generally low-to-moderate for all combinations.

Unlike stability, there is generally no link between prefix length and fidelity, except with BPIC2012 (most noticeable in LIME), where a higher prefix length generally results in a more faithful explanation. The larger size of the BPIC2012 dataset has resulted in more reasonable black box accuracy at the higher prefix lengths (see Fig. 9), which in turn appears to have ensured a better fit for LIME's surrogate models. Interestingly, there is also no pattern between stability and fidelity. It may be expected that poor stability will also result in poor fidelity, since there are so many possible explanations. Generating multiple explanations to choose the most common features appears to have mitigated the effects of any instability. This is particularly clear in the Sepsis Cases dataset, which had the most unstable explanations, but has comparable fidelity results with the other datasets. It is possible that this approach can be used to mitigate stability issues in practice (provided that the interpretation method is efficient enough to make the generation of multiple explanations feasible for timely decision-making).



(a) Single Bucket, Aggregate Encoding

(b) Prefix-length Bucket, Aggregate Encoding

(c) Prefix-length Bucket, Index-based Encoding

Figure 9: Accuracy of predictive models at each prefix length for the BPIC2012 dataset

The Sepsis Cases dataset presents an interesting pattern when index-based encoding is used. From the results, it becomes apparent that the fidelity of explanations from both LIME and SHAP are more consistent for instances closer to the decision boundary (i.e. when the initial prediction probability is closer to 0.5), and less consistent when moving away from the decision boundary. This is likely, once again, a combination of data sparsity in each bucket (caused by the down-sampling) and relatively long feature vector lengths (in comparison to aggregate encoding) resulting in an inaccurate black box model (training accuracy and testing accuracy are both low at 67.1% and 45.8% respectively) that make near-random predictions when unfamiliar instances are presented.

5.5 Work In Progress

Given the lackluster results for fidelity outlined in the previous section, a closer examination of fidelity was conducted using a toy dataset. This examination attempted to uncover how the prediction probability of a classifier for the initially predicted class varied as the value of any given feature changed. In one case, the prediction probability stayed relatively constant before changing dramatically when the feature value reached 0.5 (see figure 10). However, the explanation for this instance only identified the feature's initial value being between 0.25 and 0.5 as the cause of the prediction. While this is true, it does not account for all of the feature values that could have contributed to the prediction (i.e. the other values between 0 and 0.5). As such, when permuting close to the distribution there was no change in prediction probability, leading to low fidelity scores for that instance, and that classifier overall, as this pattern was repeated across most instances.

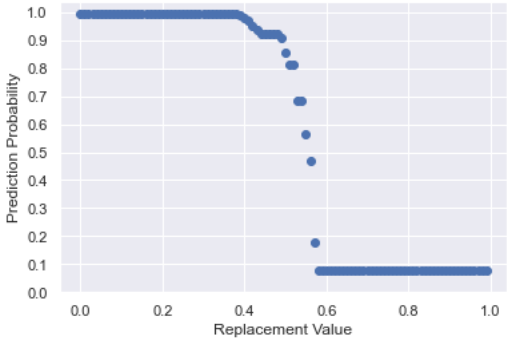


Figure 10: The prediction probability changes caused by a single variable for one instance in a toy dataset.

Namely, for that explanation, while the fidelity of the explanation was correct in the sense that the feature values within the distribution identified provided similar results to the original feature value, this does not imply that feature values outside of the distribution would provide a different result. Moreover, the importance of each feature (associated with weight) seemed to have some impact, with more important features causing extreme changes in prediction probability, such as the one in figure 10.

As such, the following changes are currently being made to the method for determining fidelity:

- Creation of two measures of fidelity:
  - ***Explanation-supporting fidelity***, measuring the correctness of the distribution provided by the explanation, i.e. checking whether the prediction probability remains unchanged when using feature values within the distribution.
  - ***Explanation-contrary fidelity***, to determine the firmness of the boundaries set by the explanation. This is the form of fidelity that was used in the evaluation that has already been conducted.
- Accounting for feature weights, as provided by the explanation, when calculating fidelity.
- Adjusting the perturbation method for explanation-contrary fidelity to include a greater range of feature values than those just outside of the distribution.

The effects of the classifier type being used is also being investigated. Initial experiments were conducted with XGBoost classifiers, and Logistic Regression and SVM classifiers are currently being trialled.

6 Future Work

There are a number of further activities that need to be undertaken to fully answer the research questions outlined in section 3.

6.1 Timeline

The proposed timeline for the project is presented in table 7.

Task	Started By	Finished By	Current Status
Complete Stage 2 Report	March 2020	June 2020	Complete
Stage 2 Submission	May 2020	May 2020	Complete
Phase 1	May 2020	April 2021	RQ1: Complete RQ2: In Progress
Phase 2	September 2020	January 2022	RQ3.1: In Progress RQ3.2: TBC
Confirmation Seminar	March 2021	March 2021	Scheduled
Phase 3	January 2022	September 2022	TBC
Progress Report Due	May 2022	May 2022	TBC
Write Initial Thesis Draft	September 2022	December 2022	TBC
Final Seminar	December 2022	December 2022	TBC
Expected Completion Date	March 2023	March 2023	TBC

Table 7: Proposed Timeline

6.2 Phase One

Currently, evaluations have been focused on LIME and SHAP, which are feature attribution methods – local methods that explain the effects of different components for a single prediction [28]. Past studies with SHAP have shown that such explanations have little to no effect on user's trust of a black box model [59] or, when attempting to determine the correctness of a machine-generated alert, little to no effect on task effectiveness or user's mental efficiency [53]. Therefore, it's been suggested that comparative explanations or counterfactual explanations could be more useful than feature attribution methods in helping users understand and evaluate prediction algorithms [53].

As such, other XAI methods must also be evaluated for usefulness. One candidate is Anchors – a post-hoc method that is an extension of LIME, and which produces decision-rule explanations [40]. User testing of Anchors has suggested that it provides a better understanding of the black box than LIME, with users being better able to predict the decisions a black box will make when given an Anchors explanation, rather than a LIME explanation [40]. Another post-hoc method that may be of use is LORE, a local explanation method that provides both factual and counterfactual decision-rule explanations for a single instance [15]. While testing suggests that Anchors is more precise than LORE, LORE is more generalisable and stable, in addition to providing a counterfactual explanation (i.e. what changes to features can provide a different prediction) [15].

The prediction algorithms currently used have also been primarily black-box machine learning algorithms, which require extensive bucketing and encoding. As deep learning methods do not require bucketing and need less extensive encoding, and have accuracy equal to or better accuracy than machine learning models [50], it would also be useful to evaluate post-hoc XAI for deep learning methods. The most commonly used method in PPA literature is LSTM [14,42,48,50], although Transformers has also been proposed as a viable method, given that the required data encoding is more compact and the resulting prediction model is inherently interpretable [2]. Similarly, an inherently interpretable attention-based LSTM model can also be generated [42]. As such, in addition to using post-hoc methods, some deep learning process prediction models can also be evaluated as white-box algorithms.

The inclusion of these additional explainable methods will also necessitate an extension to the evaluation framework. The framework currently presented in section 5.2.1 was developed primarily for post-hoc feature attribution methods. As such, if the techniques outlined above are to be evaluated, the framework must be extended to include methods of evaluating these. The following measures and metrics would need to be defined or adapted from existing metrics:

- Evaluation metrics that are comparable for both post-hoc and white box methods;
- Evaluation metrics that can be applied to rule-based explanations; and
- Evaluation metrics for non-functional evaluation (while several measures have been identified in table 2, associated metrics and methods have not yet been defined).

6.3 Phase Two

This phase is partly complete. Several of the defined evaluation measures have been applied for LIME and SHAP, allowing for some conclusions, and some evaluation metrics are currently being refined based on investigation of the obtained results. Furthermore, evaluation will also need to be conducted of the methods outlined in section 6.2 once the evaluation framework has been appropriately extended and all metrics defined. User evaluation will also be required. This will likely take the form of a human-grounded evaluation conducted with student volunteers to test the explanations provided by the chosen interpretability techniques.

6.4 Phase Three

Based on the data collected in phase two, a set of design principles for creating an explanation interface will be developed. The design principles with be evaluated with user testing of a prototype developed based on the design principles. Once again, this will likely be a human-grounded evaluation.

6.5 Publication Strategy

The evaluations already conducted have been submitted as a conference paper to the International Conference on Advanced Information Systems Engineering (CAiSE) and is currently under review. The proposed framework and metrics will also be used to evaluate interpretations generated for a medical prediction problem (not a PPA problem) for validation, and it is hoped that the results of this can be submitted to Data Mining and Knowledge Discovery's forthcoming special issue on explainable and interpretable machine learning and data mining. The Journal of Responsible Technology's special issue on accountability mechanisms in socio-technical systems is also being considered as a potential venue for publishing the work outlined in section 5.5.

A number of conferences have been considered for future publications. In addition to CAiSE, the International Conference on Service Oriented Computing is another Information Systems conference with a wide audience. Two prominent conferences in the area of BPM are the International Conference on Business Process Management and the International Conference on Process Mining. The International Conference on Very Large Data Bases also has a Process Mining track and has submission deadlines throughout the year (though the conference only occurs once a year). The IEEE International Conference on Data Engineering may also be relevant.

Various journals have also been considered for future publications. One notable journal is Decision Support Systems, which focuses on systems that can aid enhanced decision making. This journal may be a potential venue for publication once Phase 3 is complete. Other relevant journals are Expert Systems with Applications, Machine Learning with Applications, IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Service Computing and Collective Intelligence (a new journal published by ACM). One extremely ambitious goal is to publish a paper in Nature Machine Intelligence, which may be possible if a standardised evaluation framework for XAI, along with associated metrics and methods, can be developed.

References

[1] ISOIEC TR 24028:2020: Information technology – Artificial intelligence – Overview of trustworthiness in artificial intelligence. Tech. rep., The British Standards Institution, 2020.

[2] Agarwal, P., Swarup, D., Prasannakumar, S., Dechu, S., and Gupta, M. Unsupervised contextual state representation for improved business process models. In *Business Process Management Workshops*. Springer International Publishing, 2020, pp. 142–154.

[3] Appice, A., Mauro, N. D., and Malerba, D. Leveraging shallow machine learning to predict business process behavior. In *2019 IEEE International Conference on Services Computing* (Milan, Italy, 8-13 July 2019).

[4] Arya, V., Bellamy, R. K. E., Chen, P.-Y., Dhurandhar, A., Hind, M., Hoffman, S. C., Houde, S., Liao, Q. V., Luss, R., Mojsilović, A., Mourad, S., Pedemonte, P., Raghavendra, R., Richards, J., Sattigeri, P., Shanmugam, K., Singh, M., Varshney, K. R., Wei, D., and Zhang, Y. One Explanation Does Not Fit All: A Toolkit and Taxonomy of AI Explainability Techniques. arXiV:1909.03012v2, 2019.

[5] Biran, O., and Cotton, C. Explanation and justification in machine learning: A survey. In *ICJAI-17 workshop on explainable AI* (Melbourne, Australia, 20 August 2017).

[6] Carvalho, D. V., Pereira, E. M., and Cardoso, J. S. Machine Learning Interpretability: A Survey on Methods and Metrics. *Electronics 8, Article 832*, 8 (July 2019).

[7] Chen, T., and Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Aug. 2016), ACM.

[8] Dees, M., de Leoni, M., van der Aalst, W. M. P., and Reijers, H. A. What if process predictions are not followed by good recommendations? arXiv:1905.10173v3, 2019.

[9] Doshi-Velez, F., and Kim, B. Towards a rigorous science of interpretable machine learning. arXiv: 1702.08608v2.

[10] Drozdal, J., Weisz, J., Wang, D., Dass, G., Yao, B., Zhao, C., Muller, M., Ju, L., and Su, H. Trust in AutoML: Exploring information needs. In *Proceedings of the 25th International Conference on Intelligent User Interfaces* (Cagliari, Italy, 17-20 March 2020).

[11] Du, M., Liu, N., Yang, F., Ji, S., and Hu, X. On attribution of recurrent neural network predictions via additive decomposition. In *The World Wide Web Conference - WWW '19* (2019), ACM Press.

[12] Dumas, M., La Rosa, M., Mendling, J., and Reijers, H. A. *Fundamentals of Business Process Management*. Springer, Berlin New York, 2018.

[13] Folino, F., Guarascio, M., and Pontieri, L. Discovering context-aware models for predicting business process performances. In *On the Move to Meaningful Internet Systems: OTM 2012*. Springer Berlin Heidelberg, 2012, pp. 287–304.

[14] Galanti, R., Coma-Puig, B., de Leoni, M., Carmona, J., and Navarin, N. Explainable predictive process monitoring. arXiv: 2008.01807.

[15] Guidotti, R., Monreale, A., Giannotti, F., Pedreschi, D., Ruggieri, S., and Turini, F. Factual and counterfactual explanations for black box decision making. *IEEE Intelligent Systems 34*, 6 (nov 2019), 14–23.

[16] Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. A survey of methods for explaining black box models. *ACM Computing Surveys 51, Article 93*, 93 (Jan. 2018).

[17] Hevner, A. R. A three cycle view of design science research. *Scandinavian Journal of Information Systems 19* (2007), 87–92.

[18] Hevner, A. R., March, S. T., Park, J., and Ram, S. Design science in information systems research. *MIS Quarterly 28* (2004), 75–105.

[19] Hoffman, R. R., Mueller, S. T., Klein, G., and Litman, J. Metrics for Explainable AI: Challenges and Prospects. arXiV:1812.04608v2, 2018.

[20] Kim, J. Enhacing the quality of predictions in predictive business process monitoring. In *ICPM 2019 Doctoral Consortium* (Aachen, Germany, 23 June 2019).

[21] Kindermans, P.-J., Hooker, S., Adebayo, J., Alber, M., Schütt, K. T., Dähne, S., Erhan, D., and Kim, B. The (un)reliability of saliency methods. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer International Publishing, 2019, pp. 267–280.

[22] Koska, C., and Filipović, A. Blackbox AI — state regulation or corporate responsibility?, Sept. 2019.

[23] Lage, I., Chen, E., He, J., Narayanan, M., Kim, B., Gershamn, S. J., and Doshi-Velez, F. Human evaluation of models built for interpretability. In *Proceedings of the Seventh AAAI Conference on Human Computation and Crowdsourcing* (Washington State, USA, 28-30 October 2019).

[24] Li, X.-H., Cao, C. C., Shi, Y., Bai, W., Gao, H., Qiu, L., Wang, C., Gao, Y., Zhang, S., Xue, X., and Chen, L. A survey of data-driven and knowledge-aware eXplainable AI. *IEEE Transactions on Knowledge and Data Engineering TBD* (2020), 1–1.

[25] Lipton, Z. C. The Mythos of Model Interpretability. *Queue 16* (June 2018), 31–57.

[26] Lundberg, S. M., and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 2017 Neural Jnformation Processing Systems Conference* (Long Beach, USA, 4-9 December 2017).

[27] Madumal, P., Miller, T., Sonenberg, L., and Vetere, F. A grounded interaction protocol for explainable artificial intelligence. In *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems* (Montreal QC, Canada, 13-17 May 2019).

[28] Maksymiuk, S., Gosiewska, A., and Biecek, P. Landscape of r packages for explainable artificial intelligence. arXiv: 2009.13248, Sept. 2020.

[29] Marquez-Chamorro, A. E., Resinas, M., and Ruiz-Cortes, A. Predictive monitoring of business processes: A survey. *IEEE Transactions on Services Computing 11*, 6 (Nov. 2017), 962–977.

[30] Messalas, A., Kanellopoulos, Y., and Makris, C. Model-agnostic interpretability with shapley values. In *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)* (Patras, Greece, jul 2019), IEEE.

[31] Miller, T. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence 267* (Feb. 2019), 1–38.

[32] Mohana Chelvan, P., and Perumal, K. A Survey of Feature Selection Stability Measures. *International Journal of Computer and Information Technology 5*, 14 (2016), Article 14.

[33] Mohseni, S., Zarei, N., and Ragan, E. D. A multidisciplinary survey and framework for design and evaluation of explainable ai systems. arXiv: 1811.11839v4.

[34] Molnar, C. *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable.* Lean Publishing, 2020. Accessed: 29 March, 2020. [Online]. Available: https://christophm.github.io/interpretable-ml-book.

[35] Nogueira, S., Sechidis, K., and Brown, G. On the Stability of Feature Selection Algorithms. *Journal of Machine Learning Research 18*, 174 (2018), Article 174.

[36] Office of the Australian Information Commmissioner. Australian entities and the EU General Data Protection Regulation (GDPR), June 2018. Retrieved from https://www.oaic.gov.au/privacy/guidance-and-advice/australian-entities-and-the-eu-general-data-protection-regulation/.

[37] Peffers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. A design science research methodology for information systems research. *Journal of Management Information Systems 24*, 3 (Dec. 2007), 45–77.

[38] Rehse, J.-R., Mehdiyev, N., and Fettke, P. Towards Explainable Process Predictions for Industry 4.0 in the DFKI-Smart-Lego-Factory. *Künstliche Intelligenz 33*, 2 (Apr. 2019), 181–187.

[39] Ribeiro, M. T., Singh, S., and Guestrin, C. ”Why Should I Trust You?”: Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, 13-17 August 2016).

[40] Ribeiro, M. T., Singh, S., and Guestrin, C. Anchors: High-precision model-agnostic explanations. In *Proceeding of the 32nd AAAI Conference on Artificial Intelligence* (New Orleans, USA, 2-7 February 2018).

[41] Rizzi, W., Francescomarino, C. D., and Maggi, F. M. Explainability in predictive process monitoring: When understanding helps improving. In *International Congerence on Business Process Management 2020* (Seville, Spain, 13-18 September 2020), Springer International Publishing, pp. 141–158.

[42] Sindhgatta, R., Moreira, C., Ouyang, C., and Barros, A. Exploring interpretable predictive models for business processes. In *International Conference on Business process Management 2020* (Seville, Spain, 13-18 September 2020), Springer International Publishing, pp. 257–272.

[43] Sindhgatta, R., Ouyang, C., Moreira, C., and Liao, Y. Interpreting predictive process monitoring benchmarks. arXiv:1912.10558v2, 2020.

[44] Sokol, K., and Flach, P. Explainability fact sheets. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain, 27-30 January 2020).

[45] Sokol, K., and Flach, P. One explanation does not fit all: The promise of interactive explanations for machine learning transparency. arXiV: 2001.09734v1, 2020.

[46] Tama, B. A., and Comuzzi, M. An empirical comparison of classification techniques for next event prediction using business process event logs. *Expert Systems with Applications 129* (Sept. 2019), 233–245.

[47] Teinemaa, I., Dumas, M., La Rosa, M., and Maggi, F. M. Outcome-oriented predictive process monitoring: review and benchmark. *ACM Transactions on Knowledge Discovery in Data 13, Article 17*, 17 (June 2019).

[48] Teinemaa, I., Dumas, M., Leontjeva, A., and Maggi, F. M. Temporal stability in predictive process monitoring. *Data Mining and Knowledge Discovery 32*, 5 (jun 2018), 1306–1338.

[49] van der Aalst, W. M. P. *Process mining: data science in action.* Springer, Heidelberg, 2016.

[50] Verenich, I., Dumas, M., Rosa, M. L., Maggi, F. M., and Teinemaa, I. Survey and cross-benchmark comparison of remaining time prediction methods in business process monitoring. *ACM Transactions on Intelligent Systems and Technology 10, Article 34*, 34 (Aug. 2019).

[51] Verenich, I., Dumas, M., Rosa, M. L., and Nguyen, H. Predicting process performance: A white-box approach based on process models. *Journal of Software: Evolution and Process 31*, 6 (mar 2019), e2170–e2196.

[52] Visani, G., Bagli, E., Chesani, F., Poluzzi, A., and Capuzzo, D. Statistical stability indices for lime: obtaining reliable explanations for machine learning models. arXiv: 2001.11757v1.

[53] Weerts, H. J. P., van Ipenburg, W., and Pechenizkiy, M. A human-grounded evaluation of shap for alert processing. arXiv:1907.03324.

[54] Weinzierl, S., Revoredo, K. C., and Matzner, M. Predictive business process monitoring with context information from documents. In *Proceedings of the 2019 European Conference on Information Systems* (Stockholm and Uppsala, Sweden, 8-14 June 2019).

[55] Weinzierl, S., Zilker, S., Brunk, J., Revoredo, K., Matzner, M., and Becker, J. XNAP: Making LSTM-based Next Activity Predictions Explainable by Using LRP. arXiv preprint: 2008.07993v1.

[56] Weinzierl, S., Zilker, S., Stierle, M., Park, G., and Matzner, M. From predictive to prescriptive process monitoring: Recommending the next best actions instead ofcalculating the next most likely events. In *Proceedings of the International Conference on Wirtschaftsinformatik* (Potsdam, Germany, 08-11 March 2020).

[57] Yang, F., Du, M., and Hu, X. Evaluating explanation without ground truth in interpretable machine learning. arXiv:1907.06831v2.

[58] Yeshchenko, A., Durier, F., Revoredo, K., Mendling, J., and Santoro, F. Context-aware predictive process monitoring: The impact of news sentiment. In *Lecture Notes in Computer Science*. Springer International Publishing, 2018, pp. 586–603.

[59] Zhang, Y., Liao, Q. V., and Bellamy, R. K. E. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency* (Barcelona, Spain, 27-30 January 2020).

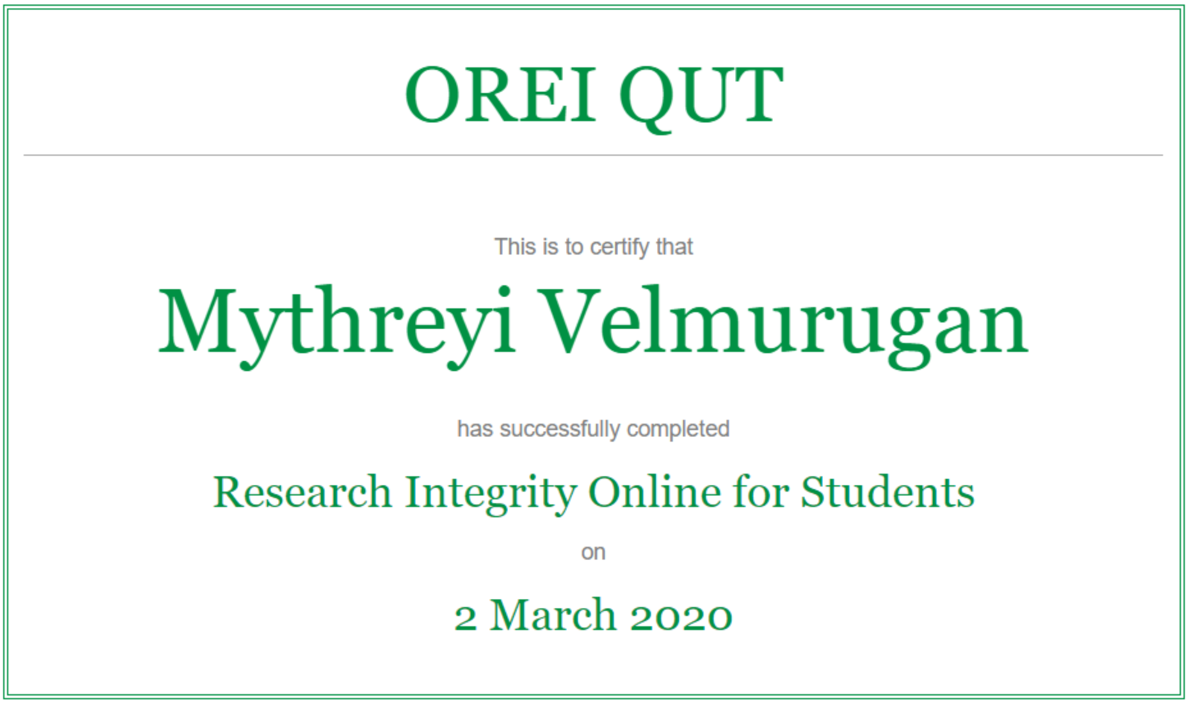
A Appendices

A.1 Plagiarism Checking

This report was submitted to iThenticate, which returned a similarity score of 4%, excluding references.

A.2 Research Integrity Online (RIO)

I have completed the Research Integrity Online (RIO) Blackboard Module. I attach my certificate of completion below.



OREI QUT

This is to certify that

Mythreyi Velmurugan

has successfully completed

Research Integrity Online for Students

on

2 March 2020

A.3 Coursework

I am required to the following coursework as part of the of the completion of this course:

- **IFN001 Advanced Information Retrieval Skill (AIRS)**: Assessment submitted prior to submission of Stage 2 Report
- **INN700 Introduction to Research**: Completed in Semester 1, 2019 as part of a previous course.
- **INN701 Advanced Research Topics**: Completed in Semester 2, 2019 as part of a previous course.

A.4 Ethics

Ethics approval has not currently been sought. An ethics application will be submitted once the methods for the human-grounded evaluation have been finalised.

A.5 Intellectual Property

I am still discussing my IP Assignment Agreement with my supervisory team. My intention is to report the research outcomes by publishing academic papers and to present them at academic conferences when available. Any software developed will be released under open-source licenses.

A.6 Health and Safety

I do not need access to labs while conducting my research, and I will not be handling biological, microbiological, biomedical or biochemical material as part of my research. As such, I have only completed the required Health and Safety Training and general evacuation training.

A.7 Data Management

A data management plan has been created online (DMP Identifier 4801).