



Australian  
National  
University

**Imported Block 1 - Jul 29, 2024**

## **ALGORITHM DEBT IN MACHINE AND DEEP LEARNING SYSTEMS**

You are invited to participate in this study conducted by Iko-Ojo Simon, a PhD candidate at the School of Computing at the Australian National University, as part of his thesis. The focus of his research is on Algorithm Debt, a new type of Technical Debt that arises from decisions made by developers during algorithm implementation. The aim of this particular study is to investigate the causes, effects, and mitigation strategies of Algorithm Debt within the Machine and Deep Learning domain.

This survey consists of both open and closed-ended questions about Algorithm Debt in Machine and Deep Learning Systems. If you do not have experience in developing Machine and Deep Learning systems, please do not proceed with the questionnaire. For clarity, Machine and Deep Learning systems are defined as systems that integrate Machine or Deep Learning models or algorithms with other non-Machine and Deep Learning components. The survey should take no more than 10 minutes to complete and is only required to be

completed once.

Please review the [participant information sheet](#) which can be viewed in any browser, before proceeding. The ethical aspects of this research have been approved by ANU's Human Ethics Committee (Protocol: H/2024/0982).

## Consent

By continuing, I confirm that:

- I have read and understood the Participant Information Sheet provided to me regarding the research project.
- I have at least one year of hands-on experience with frameworks such as TensorFlow, Keras, DL4J, PyTorch, MXNet, or Microsoft Cognitive Toolkit e.t.c.
- Any questions or concerns I had about the project have been addressed to my satisfaction.
- I agree to participate in the study and understand that I may withdraw at any time before submitting my data.
- I consent to the information I provide, or that is obtained with my permission during the project, being included in a thesis, presentations, publications, and stored in an open repository, under the condition that I will not be personally identified.

- ☐ Yes, I meet all the requirements and agree to participate in this research.
- ☐ No, I do not agree or do not meet the criteria mentioned.

## Participant Characterisation

The aim of this section is to gather information about your background and experience. By understanding factors such as your region, role, and level of expertise in Machine and Deep Learning projects, we can better contextualise your responses related to Algorithm Debt.

In which region are you currently employed?

- ☐ Africa
- ☐ Antarctica
- ☐ Oceania
- ☐ Asia
- ☐ Europe
- ☐ North America
- ☐ South America

How would you rate your experience in Machine and Deep Learning systems development?

- ☐ Novice (Minimal or “textbook” knowledge without connecting it to practice)
- ☐ Beginner (Working knowledge of key aspects of practice)
- ☐ Competent (Good working and background knowledge of area of practice)
- ☐ Proficient (Depth of understanding of discipline and area of practice)
- ☐ Expert (Authoritative knowledge of domain and deep tacit understanding across area of practice)

Are you answering the following questions based on your current role or your previous hands-on technical role?

- ☐ Current Role
- ☐ Previous Role
- ☐ Both

Which of the following closely aligns with your current or previous role?

- ☐ Machine and Deep Learning Model Developer
- ☐ Machine and Deep Learning Builder / Engineer
- ☐ Machine and Deep Learning Model Analyst
- ☐ Machine and Deep Learning Researcher
- ☐  Others

### Introduction to Technical Debt and Algorithm Debt

In this section, we introduce you to the concepts of Technical Debt and Algorithm Debt.

**Technical Debt:** Technical Debt refers to the additional work or cost incurred due to shortcuts or suboptimal choices made during the software development process.

**Explanation:** When developers make quick fixes or choose easier solutions to meet deadlines or immediate needs, they may create issues that will need to be addressed later. These shortcuts or solutions can accumulate over time and make future changes or maintenance more difficult and costly.

**Algorithm Debt:** Algorithm Debt is a newly uncovered type of Technical Debt that is incurred when an algorithm or logic is implemented in a way that it may be efficient in the short term but require rework in the future as the system evolves, ultimately degrading overall system performance. It may be identified through prior decisions, or in discussions of potential actions or solutions . Algorithm Debt is characterised by its negative impact on system performance, efficiency, and other quality attributes as the system evolves over time.

**Explanation:** In Machine and Deep Learning systems, Algorithm Debt arises when suboptimal algorithms are implemented, resulting in performance issues such as scalability challenges as the system evolves. This may necessitate revisiting and improving the code in the future. The example provided (from an open source) can be classified as Algorithm Debt, as the developer acknowledges in the TODO comment that the current pooling implementation may not be the most efficient for the "channel-first" case.

```
// TODO(Yangqing): Is there a faster way to do
pooling in the channel-first case?

template <typename Dtype>
void PoolingLayer<Dtype>::Forward_cpu(const
vector<Blob<Dtype>*>& bottom,
    const vector<Blob<Dtype>*>& top) {
const Dtype* bottom_data = bottom[0]-
>cpu_data();
Dtype* top_data = top[0]->mutable_cpu_data();
const int top_count = top[0]->count();
```

Before now, how familiar are you with the concept of Algorithm Debt?

- ☐ Not familiar at all (I have never heard of Algorithm Debt)
- ☐ Slightly familiar (I have heard of Algorithm Debt but do not understand it well)
- ☐ Moderately familiar (I have some understanding of Algorithm Debt but need more information)
- ☐ Very familiar (I understand Algorithm Debt and its implications well)
- ☐ Extremely familiar (I have a thorough understanding of Algorithm Debt and have worked with it extensively)

How often do you encounter Algorithm Debt in your projects?

- ☐ Never
- ☐ Rarely
- ☐ Sometimes
- ☐ Often
- ☐ Always

# Causes of Algorithm Debt

In this section, we are interested in understanding the factors that contribute to Algorithm Debt in Machine and Deep Learning systems. Please reflect on your experience and provide insights on the common causes or conditions that lead to Algorithm Debt in the projects you've worked on

	Never	Rarely	Sometimes	Often
How often do hardware limitations impact your ability to implement optimal algorithms in Machine Learning and Deep Learning systems, resulting in the need for future rework?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
To what extent have API changes or compatibility issues led you to implement temporary algorithm solutions that required future rework?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
To what extent do data issues force you to implement algorithms that you knew were inadequate in the long run?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
How often do model training processes lead to a need for future rework related to your algorithms?	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Which other factors could lead to Algorithm Debt? Select all that apply.

- ☐ Complexity of Machine and Deep Learning algorithms
- ☐ Lack of knowledge from the developer
- ☐ Tight deadlines and pressure for quick delivery
- ☐ Lack of proper documentation
- ☐ Insufficient testing
- ☐ Lack of collaboration between teams

☐

Others

## Effects of Algorithm Debt

In this section, we are interested in understanding the effects that Algorithm Debt can have in Machine and Deep Learning systems. Please reflect on your experience and provide insights on the common impacts that Algorithm Debt had in the projects you've worked on.

If you have encountered Algorithm Debt, how would you rate the impact on the system?

- ☐ No Impact
- ☐ Minor Impact
- ☐ Moderate Impact
- ☐ Significant Impact
- ☐ Not applicable

What are some of the effects of Algorithm Debt in a Machine Deep Learning software projects? Select all that apply.

- ☐ Poor model scalability
- ☐ Decreased model performance
- ☐ Model bias



- ☐ Increased maintenance costs
- ☐ Delayed project timelines
- ☐ Increased model complexity
- ☐ Loss of stakeholder confidence
- ☐  Others

## Resolving/Mitigating Algorithm Debt

In this section, we are looking for insights on strategies or practices that can help resolve or mitigate Algorithm Debt in Machine and Deep Learning systems.

Do you think that the Algorithm Debt you encountered could have been prevented?

- ☐ No
- ☐ Yes

How do you think that Algorithm Debt in Machine and Deep Learning systems can be mitigated? Select all that apply.

- ☐ Code reviews and refactoring
- ☐ Better resource allocation
- ☐ Improved team communication
- ☐ Systemic testing practices

☐ Adopting automated management tools

☐  Others

## Final Thoughts

In this final section, we invite you to share your thoughts, insights, or experiences related to Algorithm Debt in Machine and Deep Learning systems.

What are your thoughts on Algorithm Debt? Select all that apply.

☐ I believe Algorithm Debt is an important concept that needs more attention in Machine and Deep Learning projects.

☐ I think Algorithm Debt is overemphasised and not a major concern in practice.

☐ Algorithm Debt poses significant challenges in maintaining and scaling Machine and Deep Learning Systems.

☐ I am concerned about the long-term impact of Algorithm Debt on system performance.

☐ There should be tools to manage Algorithm Debt effectively.

☐  Others