1. Introduction

Hello, thanks for taking your time to meet with us today.

We are doing a study on software maintenance differences between machine learning and non-machine learning systems. During the interview, I'd start by asking a few questions about your background with programming, and then we'll have a discussion about your experience with developing and maintaining ML systems. I will be reading from a script to make sure that we follow the same procedure across all participants.

Is it ok if I record this discussion so that I can review it later? In any data collected, or in reports or papers that are published, you will not be identified by name. Please be careful not to discuss any sensitive information about the company you work for. If you do mention any, we will do our best to remove it from our transcripts, but better if you don't mention such sensitive information at all.

2. Demography questions

- How many years of experience do you have in total with software development, and how many years with machine learning development?

	With Machine Learning	Total
Experience (in years)		

- What is your current job role right now?
- Which programming languages do you use most for your machine learning software development?

- Next, I will go through a series of job responsibilities in machine learning development, and I'd like you to tell me "yes" or "no" experience you have before.

Job responsibilities	**Yes**	**No**
Data Streaming Pipeline Establish		
Data Labeling		
Data Cleaning		

Feature Selection		
Model Training		
Model Validation		
Model Deployment		
Model Performance Monitoring		

3. Metric Related Questions

3.1 Question Related to Current project:

- What kind of machine learning projects are you working on now, for example, vision, speech recognition, recommendation system, etc.
- How long have you been working on this project?
- What kind of metrics do you usually use to measure your **source code quality**? Do you use any kind of threshold for these metrics? Do you use any publicly available tool to do this?
- What kind of metrics do you usually use to measure **data quality**? Do you use any kind of threshold for these metrics? Do you use any publicly available tool to do this?
- What kind of metrics do you usually use to measure your **model quality**? Do you use any kind of threshold for these metrics? Do you use any publicly available tool to do this?

3.2 Question Related to Deployed projects:

- After deployment, what metrics do you use to monitor the overall performance (data, model, source code) of the projects?
- What kind of maintenance activities do you do to maintain source code quality after deployment?
- What kind of maintenance activities do you do to maintain data quality after deployment?
- What kind of maintenance activities do you do to maintain the model after deployment?
- If you see a performance degradation in your deployed projects, what immediate steps do you take?

3.3 Question Related to Difference between ML and non-ML Maintenance:

- Do you think there are differences in the maintenance procedure between ML software and non-ML software? If yes, could you elaborate.

4 Question Related to ML Maintenance Issues verification:

- Have you ever faced the below mentioned machine learning related circumstances? If so, do you think these impact maintainability? Why ?

	Questions	**Meet before**	**Impact**	**Why**
Entanglement	Have you ever worked on a machine learning system in which a change to one component (such as a feature, hyper-parameter, etc.) required making changes to multiple other components?			
Correction Cascades	Have you ever worked with a machine learning system that involved running multiple distinct models sequentially (AKA the output of one model was used as the input for the next model)?			
Undeclared Consumers	Have you ever worked on a system in which results from your models may have been implemented for external parties to use in their systems but did not inform you?			
Unstable Data Dependencies	Have ever worked on a system in which input features were unstable, AKA they would change over time?			
Legacy Feature	Have you ever worked on a system that contained a feature that became redundant over time but may not have been immediately identified as redundant?			

ε - Feature	Have you ever worked on a system in which there was a push to improve model accuracy even though the accuracy gain would have been marginal or there would have been high complexity overhead?			
Glue Code	Have you ever worked on a system where the implementation of a supporting package within a code hindered the ability to improve or modify the model?			
Pipeline Jungle	Have you worked on a system in which the data pipeline appeared to have grown organically over time as opposed to being thoughtfully planned out beforehand?			
Bundled Features	Have you ever worked on a system that incorporated bundled components in which not all components were actually needed?			
Dead Experimental Codepaths	Have you ever worked on a system that contained obsolete code paths that may have been originally added for experimental or testing purposes?			
Multiple-Language Smell	Have you ever worked on a system that used multiple programming languages in a way that did not seem efficient?			