Characterizing and Detecting Mismatch in ML–Enabled Systems

Project Introduction

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution.

Copyright 2019 Carnegie Mellon University.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

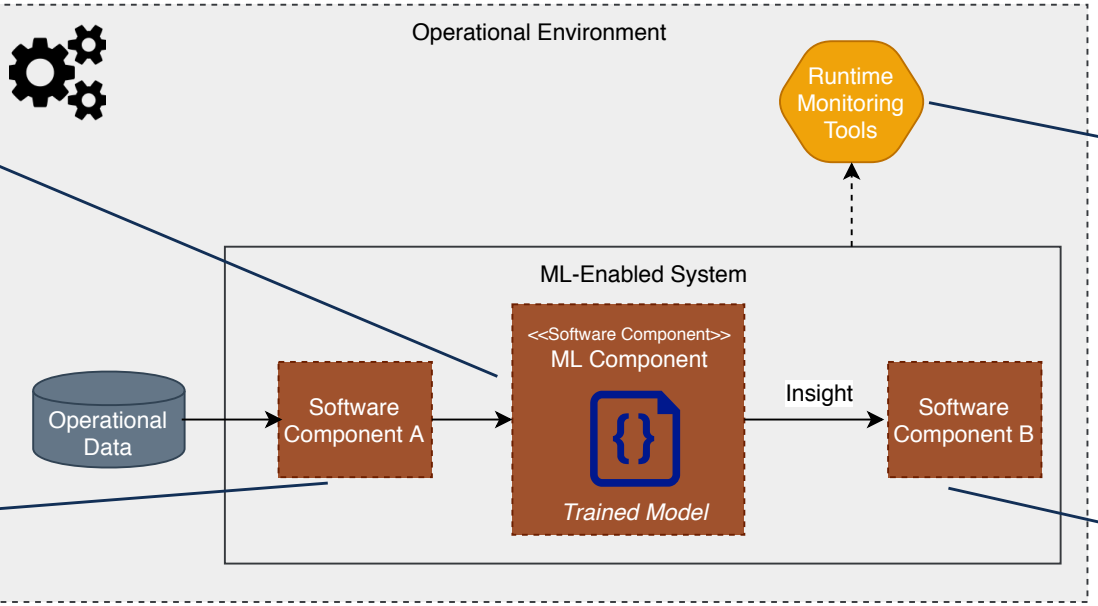
DM19-1090

ML-Enabled System

We define an ML-enabled system as a software system that relies on one or more ML software components to provide required capabilities

A trained model(s) is wrapped as a special type of Software Component called an ML Component

ML component receives (processed) operational data from one software component …



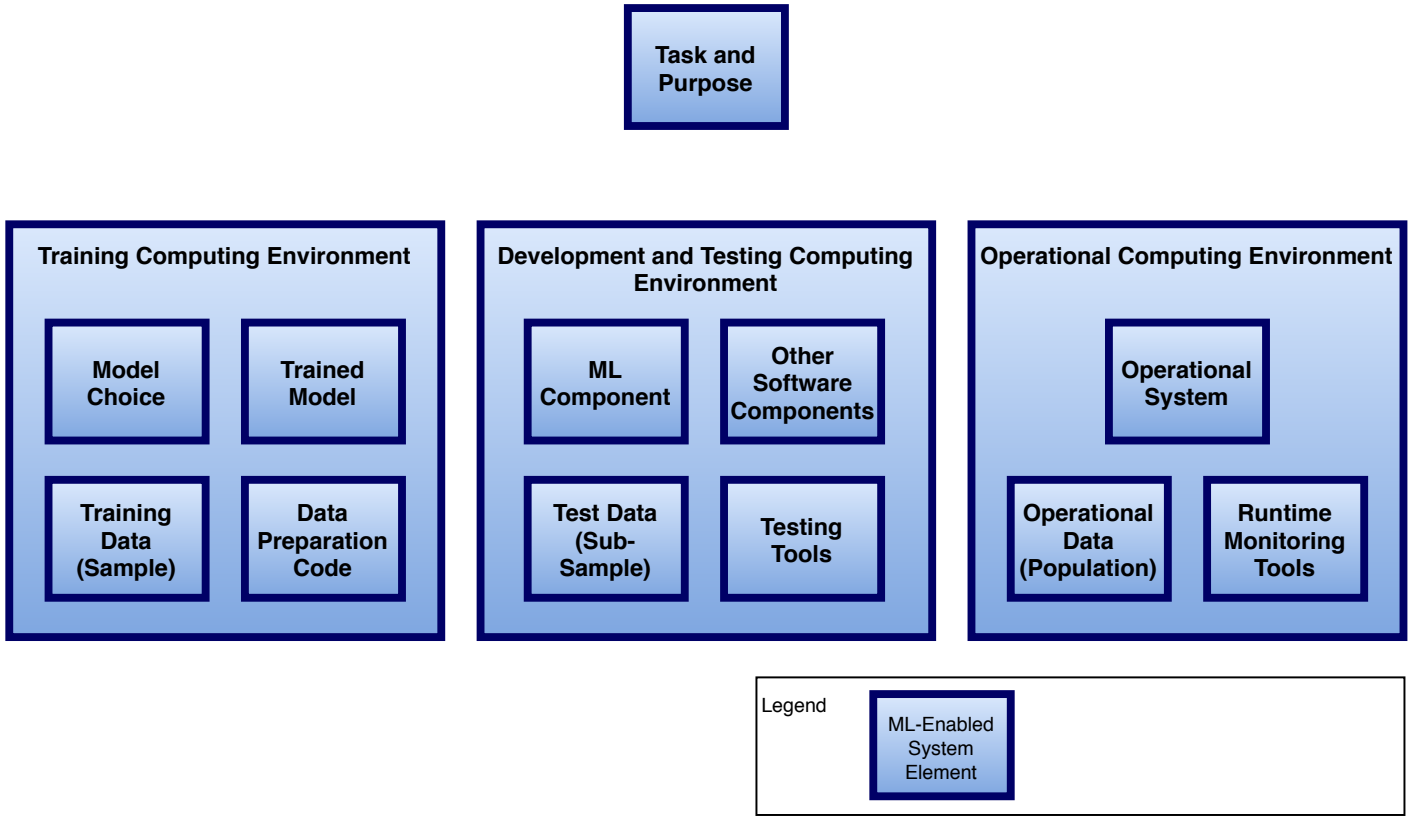
Runtime Monitoring Tools in the operational environment obtain and react to measures produced by the ML-Enabled System

… and generates an insight that is consumed by another software component.

© 2019 Carnegie Mellon University

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution.

Elements of ML-Enabled Systems



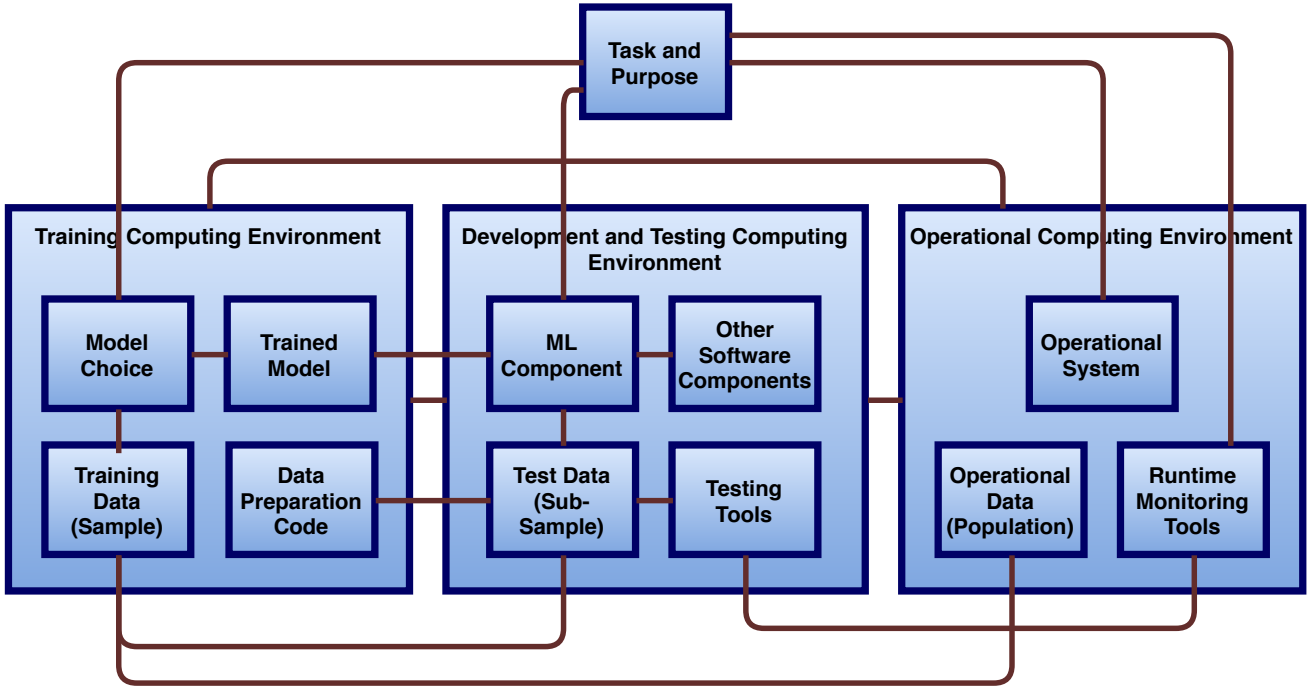
We define elements of ML-enabled systems as the non-human entities involved in the training, integration and operation of ML-enabled systems.

© 2019 Carnegie Mellon University

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution.

Motivation

Many of the challenges that we see in trying to deploy ML-enabled systems into operational environments is due to mismatch, or lack of alignment, between elements of ML-enabled systems.



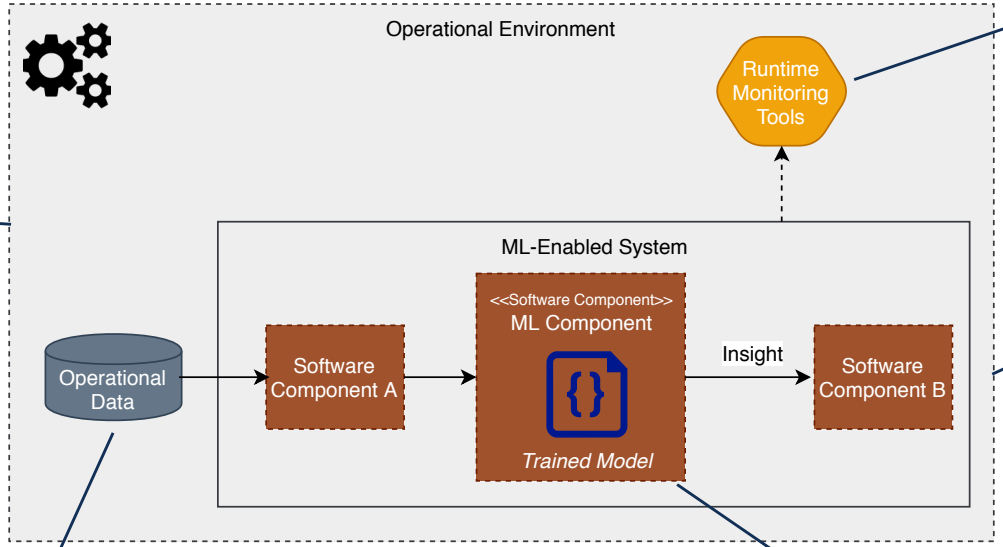
There is very little existing guidance because development of ML and AI capabilities is still mainly a research activity or a stand-alone project, with the exception of large companies.

© 2019 Carnegie Mellon University

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution.

Examples of Mismatch

Poor system performance because computing resources for model testing different from operational computing resources (**computing resource mismatch**)



Tools not set up to detect diminishing model accuracy, which is the "goodness" metric defined for the ML component (**metric mismatch**)

System failure due to poor testing — developers not able to replicate testing done during model training (**test environment mismatch**)

Poor model accuracy because model training data different from operational data (**data distribution mismatch**)

Large amounts of glue code because trained model input/output very different from operational data types (**API mismatch**)

© 2019 Carnegie Mellon University

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution.

Hypothesis

We hypothesize that a reason for mismatch is because ML-enabled systems typically involve three different and separate workflows

- Model training
- Model integration
- Model operation

… performed by three different sets of people

- Data scientists
- Software engineers
- Operations staff

… with different skills,

… who often make assumptions about what the other people know or need.

The goal of the interview is to

1. Elicit examples of mismatch that you have encountered (or can likely occur) in the development and deployment of ML-enabled systems
2. Identify key information that needs to be shared in order to avoid mismatch

© 2019 Carnegie Mellon University
[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution.