

Joora bachi

- Five open source bug repositories
- Their characteristics (six categories)
 - Quantitative

Def: Non-reproducible bug

Bugzilla and Jira

① Interbug dependencies (45%)

- ↳ dependent on other bugs (that have been fixed)
- ↳ Not clear what have been fixed
- duplicate / closely related

② Environmental differences (24%)

- Cached data / local data
- firewalls, plugins, extensions
- System configuration, IDE settings

③ Insufficient Information (14%)

- less of test case
- precise test cases

④ Conflicting expectation (12%)

- misunderstanding the purpose of a function
- change in requirement

⑤ Non-deterministic behavior (3%)

⑥ Other - (2%) mistake of reporter

Säker

- Stack overflow posts
- about Java
- qualitative

Guidelines
are provided

- ① class/interface/ method not found
- ② missing code
- ③ External library not found
- ④ Identifier/ object not found
- ⑤ Too short code Snippet
- ⑥ Miscellaneous
- ⑦ Database / File / UI dependencies
- ⑧ outdated code
- ⑨ Inaccurate claim
- ⑩ Ill-defined issues (not consistent)
Unknown source

Horton & Poirier

- GitHub Gist

- quantitative

① Mismatched name

② System dependencies

→ additional libraries

→ missing C library.

③ Custom environments

→ separate execution package

④ Unlisted package

→ used libraries that are
not listed

⑤ deprecated package

→ packages that are not available
anymore

⑥ Configuration settings

→ additional config file that is
not provided. (TWITTER
API-KEY)

⑦ Language Version

→ Python 2 vs Python 3

⑧ Operating System

→ Mac vs Windows

Responses from Developers

① Version errors

→ different version of softwares

② Unlisted/Unknown dependencies

MacOS

- bug repositories

① Bug duplication

→ duplicate might already be solved

② False positive Bugs

→ features that are not available in the software

→ simple tweaking / not bugs

③ Bug Intermittency

→ non-determinant

→ doesn't occur consistently

④ Missing info

→ Not all info is there to reproduce

→ Missing / not carefully reported

⑤ Ambiguous Specification

→ misidentify feature as bug

⑥ Performance Regression

- performance are subjective
- & dependent on machine
- Need exact configuration

⑦ Lack of Cooperation

- fail to gain attention of developers
- Auto closed by bots
- lack of cooperation from the reporter

⑧ Memory misuse

- leak of small object ~~but can~~ lead to unbounded memory
- mismanagement of memory

⑨ Third Party Defects

- third party library dependencies
- lack of control
- No way to install the same software
↓
easy

⑩ Restricted Security Access

- Many info can be shared
- lack of info make it challenging

⑪ Touch & Gesture

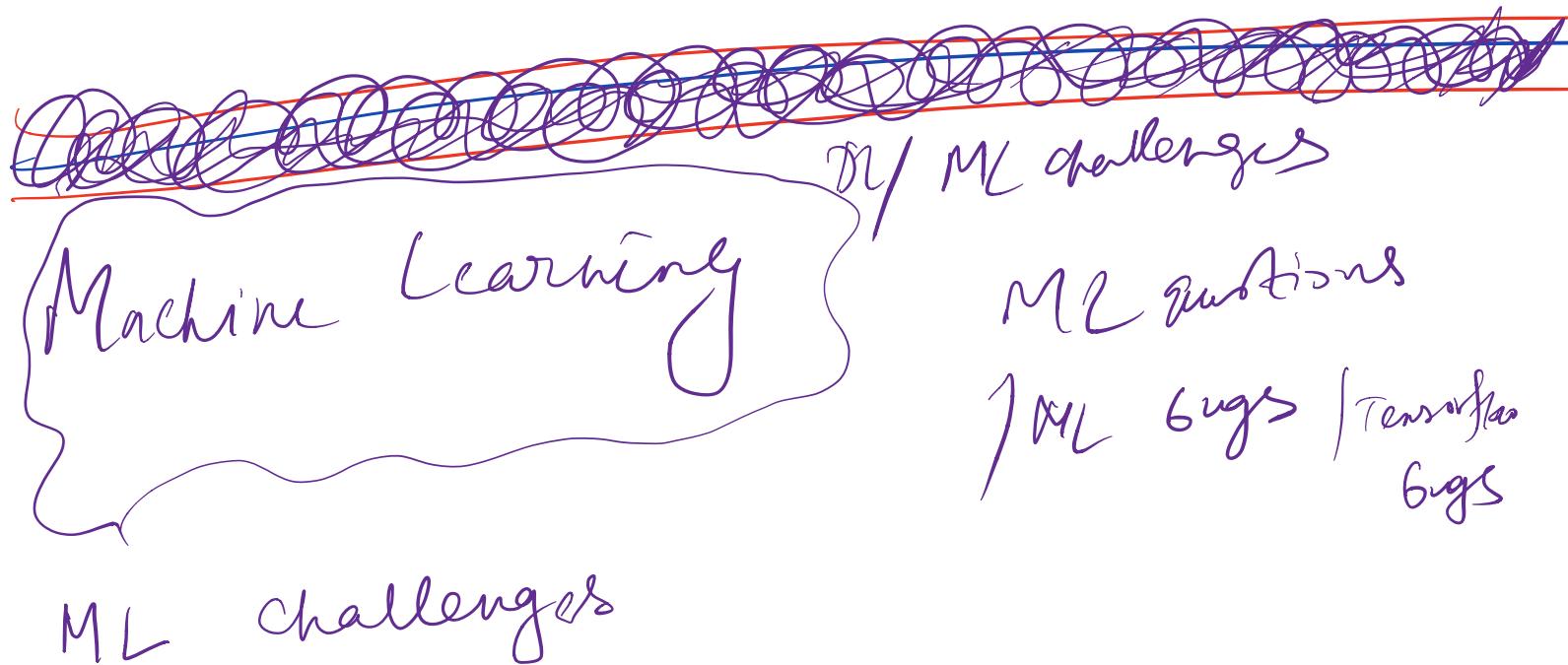
- Accessibility bugs
- Hard to imitate

75% → single factor NP

25% → Multiple NP

12

MISC



Alshabani et al (ML application challenges)

↳ ML skills

- Data preparation
- model deployment issues

[Early
late]

Early studies by
Patek

- ML suffers from no accepted answers
- ML .. is No answers.
- ML questions take 10 more times
- Lack of Experts in SO
- DP & MD most challenging
- Challenge : data & feature preprocessing
- NLP & DL are topics for SO
- TF has a high level of difficulty
- Can be answered with ML knowledge
- Question that need Concept + implementation
Suffers from lack of no accepted answers

Journal - ML Question

- ML question into several (7) pipelines

- different stage → specific problem
- which libraries are most difficult

- Model creation most difficult
- Data prep is also challenging
 - Type mismatch questions
 - shape mismatch
- Data cleaning is a common problem
- Questions on exceptions/error are prevalent
- Parameter selection is difficult
- choice of loss function is diff in DL

- Scikit-learn has more diff. in hyper-param tuning
- Scikit needs a deeper look in its API design to improve usability
- DL has more training time difficulties.

Sum → ML Bugs

- Seven categories of bug
- Twelve fix patterns
- 64% are small-fix

3 big ml projects on Github

→ Only "Reproducible" bugs

① Compatibility bug (23)

incompatible - incomp. OS, Version mismatch,
backward compatibility, hardware conflict

② Variable bug (30)

incorrect transfer, assignment, improper
data constraint, wrong format, abnormal initialization,

③ Documentation (19)

untimely update, interface error,

④ Performance optimization (n)

Performance related bugs.

⑤ Memory overflow (3)

→ Stack overflow

⑥ Design defect (20)

→ Potential defect in design due to lack of
consideration

⑦ Others (5.5)

Then they identify the bugs
fix patterns

Maintenance type

Zhang DL-challenges

Sample of 715 SO questions in DL

Six types of questions

3 main types of questions

five root causes

Programming Questions

① program crashes

② model migrations & deployment

③ implementation

④ training anomaly

⑤ comprehension

⑥ installation

⑦ performance

Program
model
implementation
are
not
discussed

① Implementation

- implement desired functionality
- how to use APIs of interest
- Effective use of GPU
- distributed training

② Program Crash

- runtime exceptions
- shape inconsistency
- numerical error
- GPU/CPU incompatibility

③ Training anomaly

(difficult
for junior
developer)

- Unreliable training behavior
- extreme accuracy
- loss value never dropping
- Overfitting

④ Model Migration

- model implementation between different frameworks

-

⑤ Performance

- training time
- memory usage
- optimize model performance

⑥ Comprehension

- clarifications on concept
- algorithms
- frameworks
- often answered by learning resources

⑦ Installation

- Software reliability issue
- very hard installations

Root cause of issues in DL

① API misuse

② GPU computations

③ Incorrect hyperparameter selection

④ static graph computation

⑤ limited debugging and profiling support

Zhang - Tensorflow bugs

- bug related to tensorflow from SO & GitHub

178 bugs → qualitative

i) Symptoms and root cause

ii) Challenge in detection

iii) challenge in localization

four symptoms, 7 cause

five challenges

five strategies -

Symptoms

① Errors

analogous to exceptions or crashes

② Low Effectiveness

extraordinarily poor accuracy, loss or unexpected outputs.

③ Low Efficiency

Slow running / indefinitely

④ Unknown

detected by code review

Cause

① Incorrect Model parameter or structure

- inappropriate parameter

{ many cycles to catch the failure

{ failure occurs at Training stage

{ Difficult to identify when the symptom is error

② Unaligned Tensors
- when the shape doesn't match

③ Confusion with Tensorflow Computation Model (CCM)

- lack of familiarity of the computational model

④ TensorFlow API Change

- upon a new release of TF

⑤ TensorFlow API Misuse

- Bugs introduced by TF API

⑥ Structure Inefficiency

-

⑦ Others

Empirically study of Bugs in ML

Things

- Bug databases or code rep.
- Manual categorization of bug and their fixes.

① Algorithm - the algorithm implementation 22.6/

② Assignment/init - errors in assigning variables

③ data - wrong data structure

④ External interface - error from other system

⑤ Internal " - " " same "

⑥ logic - incorrect expression in conditional exp.

⑦ non-functional - improper variable or method names, wrong document

⑧ Timing / Optimization - error in performance issues deadlock, memory issues

⑨ Configuration - non code file affecting functionality

⑩ Others - other bugs

Catalino - classifying bug types

- taxonomy of reported bugs
(or bugs broken post amount of time)
- characterization of different bugs.

① Configuration issue

- external libraries that needs to be updated
- wrong directory or xml or manifest file

② Network issue

- having connection or server issue
- network problems

- Server issue
- improper Configuration of Communication protocols.

③ Database-related issue

- Connection problem between app. and a database .

④ GUI-related issues

- stylistic errors (screen layouts, element colors)
- unexpected failures & errors (unexpected error message)

⑤ Performance issue

- report performance issues (memory overuse, energy leaks)

⑥ Permission / Deprecation issue

- presence, modification or removal of deprecated method APIs
- Unused API permissions are included

⑦ Security Issue

- Vulnerability and other security related problems.

⑧ Program Anomaly issue

- introduced by enhancing existing source code

⑨ Test code - related issue

- running, fixing, updating test cases
- intermittent test
- inability to test to find de-localized bugs