Predicting bug report severity with Large Language Model

Andressa Oliveira
Robin Moine

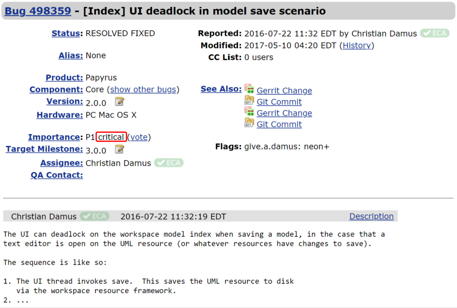
Polytechnique Montréal
DORSAL Laboratory

Outline

1. Contextualization
2. Approach
3. Results
4. Future approaches

Contextualization

- Bug Report Overview
- Severity Levels
- Bug Report Registration



Bug 498359 - **[Index] UI deadlock in model save scenario**

Status: RESOLVED FIXED

Alias: None

Product: Papyrus
Component: Core (show other bugs)
Version: 2.0.0
Hardware: PC Mac OS X

Importance: P1 critical (vote)
Target Milestone: 3.0.0
Assignee: Christian Damus ✓ ECA
QA Contact:

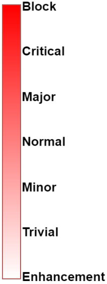
Reported: 2016-07-22 11:32 EDT by Christian Damus ✓ ECA
Modified: 2017-05-10 04:20 EDT (History)
CC List: 0 users

See Also: Gerrit Change
Git Commit
Gerrit Change
Git Commit

Flags: give.a.damus: neon+

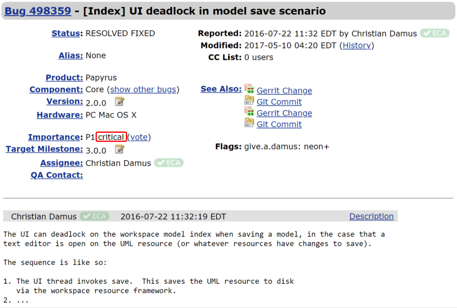
Christian Damus ✓ ECA 2016-07-22 11:32:19 EDT Description

```
The UI can deadlock on the workspace model index when saving a model, in the case that a
text editor is open on the UML resource (or whatever resources have changes to save).

The sequence is like so:

1. The UI thread invokes save.  This saves the UML resource to disk
   via the workspace resource framework.
2. ...
```



Contextualization

- Bug Report Overview
- Severity Levels
- Bug Report Registration



Bug 498359 - **[Index] UI deadlock in model save scenario**

Status: RESOLVED FIXED
Alias: None
Product: Papyrus
Component: Core (show other bugs)
Version: 2.0.0
Hardware: PC Mac OS X
Importance: P1 critical (vote)
Target Milestone: 3.0.0
Assignee: Christian Damus ✓ECA
QA Contact:

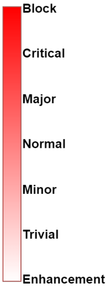
Reported: 2016-07-22 11:32 EDT by Christian Damus ✓ECA
Modified: 2017-05-10 04:20 EDT (History)
CC List: 0 users

See Also: Gerrit Change
Git Commit
Gerrit Change
Git Commit

Flags: give.a.damus: neon+

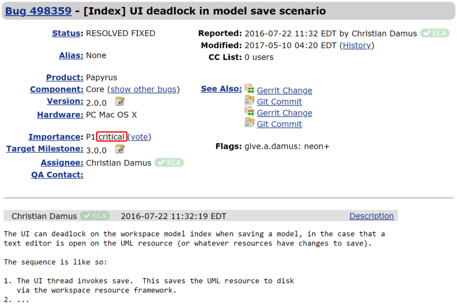
Christian Damus ✓ECA 2016-07-22 11:32:19 EDT Description

```
The UI can deadlock on the workspace model index when saving a model, in the case that a
text editor is open on the UML resource (or whatever resources have changes to save).

The sequence is like so:

1. The UI thread invokes save.  This saves the UML resource to disk
   via the workspace resource framework.
2. ...
```



Contextualization

- Bug Report Overview
- Severity Levels
- Bug Report Registration



Bug 498359 - **[Index] UI deadlock in model save scenario**

Status: RESOLVED FIXED

Alias: None

Product: Papyrus
Component: Core (show other bugs)
Version: 2.0.0
Hardware: PC Mac OS X

Importance: P1 critical (vote)
Target Milestone: 3.0.0
Assignee: Christian Damus ✓ECA
QA Contact:

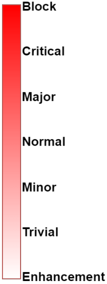
Reported: 2016-07-22 11:32 EDT by Christian Damus ✓ECA
Modified: 2017-05-10 04:20 EDT (History)
CC List: 0 users

See Also: Gerrit Change
Git Commit
Gerrit Change
Git Commit

Flags: give.a.damus: neon+

Christian Damus ✓ECA 2016-07-22 11:32:19 EDT Description

```
The UI can deadlock on the workspace model index when saving a model, in the case that a
text editor is open on the UML resource (or whatever resources have changes to save).

The sequence is like so:

1. The UI thread invokes save.  This saves the UML resource to disk
   via the workspace resource framework.
2. ...
```



Block

Critical

Major

Normal

Minor

Trivial

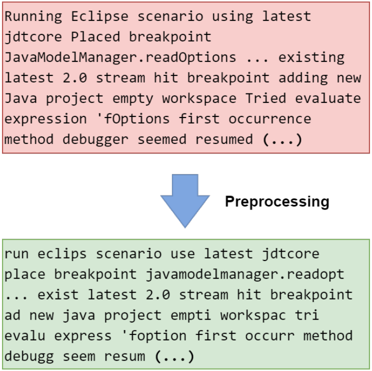
Enhancement

Approach

- Prediction of bug severity with Large Language Model (Llama2)
- Comparison with state of the art algorithms in the literature

Data preprocessing

- Importance of preprocessing
- Techniques:
  - Filter by severities
  - Binary severities
  - Remove URLs
  - Stop-word removal
  - Stemming



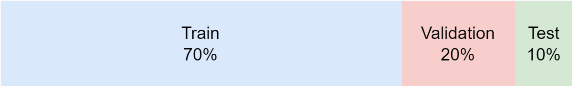
```
Running Eclipse scenario using latest
jdtcore Placed breakpoint
JavaModelManager.readOptions ... existing
latest 2.0 stream hit breakpoint adding new
Java project empty workspace Tried evaluate
expression 'fOptions first occurrence
method debugger seemed resumed (...)
```

Preprocessing

```
run eclips scenario use latest jdtcore
place breakpoint javamodelmanager.readopt
... exist latest 2.0 stream hit breakpoint
ad new java project empti workspac tri
evalu express 'foption first occurr method
debugg seem resum (...)
```

Splitting the dataset

- Fixed division of the dataset



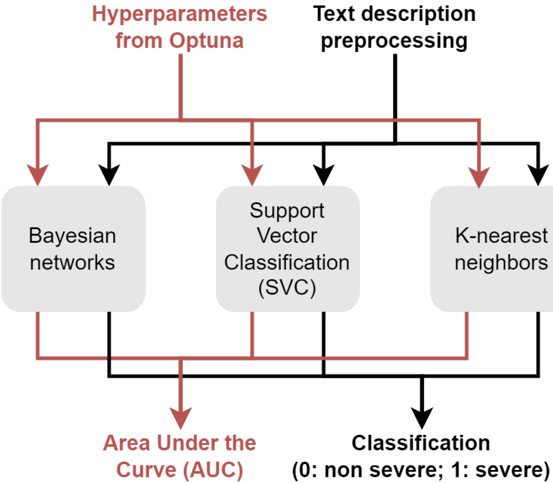
Train
70%

Validation
20%

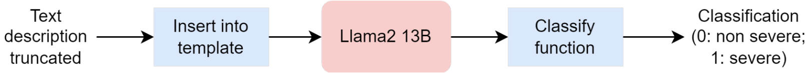
Test
10%

- Maintained the percentage of each class in train, validation and test
  - 0: non-severe, 1: severe

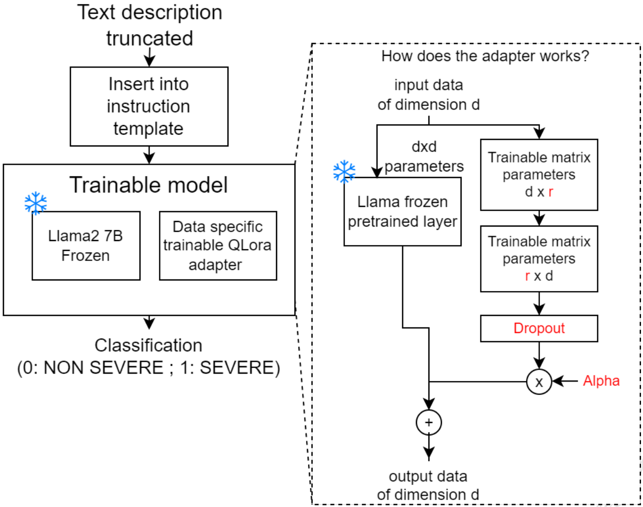
Baselines


Hyperparameters from Optuna
Text description preprocessing
Bayesian networks
Support Vector Classification (SVC)
K-nearest neighbors
Area Under the Curve (AUC)
Classification (0: non severe; 1: severe)

Llama 2 - Inference


Text description truncated
Insert into template
Llama2 13B
Classify function
Classification (0: non severe; 1: severe)

Llama 2 - Fine tuning


Text description truncated
Insert into instruction template
Trainable model
Llama2 7B Frozen
Data specific trainable QLora adapter
Classification
(0: NON SEVERE ; 1: SEVERE)
How does the adapter works?
input data of dimension d
dxd parameters
Llama frozen pretrained layer
Trainable matrix parameters d x r
Trainable matrix parameters r x d
Dropout
x
Alpha
+
output data of dimension d

Llama 2 - Fine tuning

Training data are not balanced (25% non-severe bugs in eclipse, 26% non-severe bugs in mozilla)

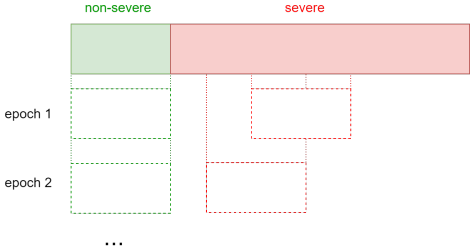


Figure: Random undersampling to use all of the training dataset and provide as many SEVERE as NON SEVERE samples per training loop (epoch)

Results

Model	Accuracy		Area under the curve	
	eclipse	mozilla	eclipse	mozilla
Best baselines (KNN)	71%	73%	0.65	0.56
Llama2-13B inference	14%	21%	-	-
Llama2-7B finetuning	85%	83%	0.89	0.88

Future approaches

- Do a better preprocessing: remove the bugs descriptions that do not provide enough information to classify them as SEVERE or NON-SEVERE
- Classify bugs more precisely: critical, enhancement...
- Use more powerful Large Language Model

Thank You!

Andressa Stéfany Oliveira
astefanysoliveira@gmail.com

Robin Moine
robin.moine456@gmail.com