Review Dynamics in Open-Source Software: A Case Study of OpenStack

Siqi (David) Liu

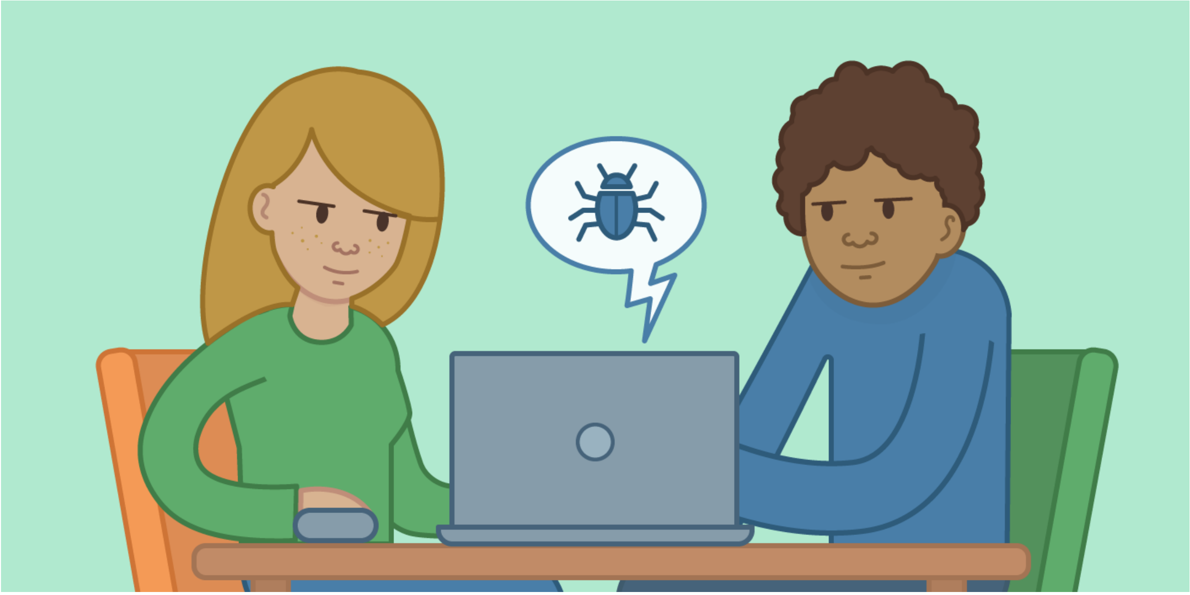
M.Math. CS, University of Waterloo

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Background

Code review is crucial in identifying bugs and maintaining standards.

In open-source software (OSS), code review is done collaboratively with activities visible to the public.



UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

What if …

Junior reviewers tend to agree with more experienced reviewers?

Reviewers tend to agree with prior reviewers?

Developers who collaborate more often tend to agree with each other?

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

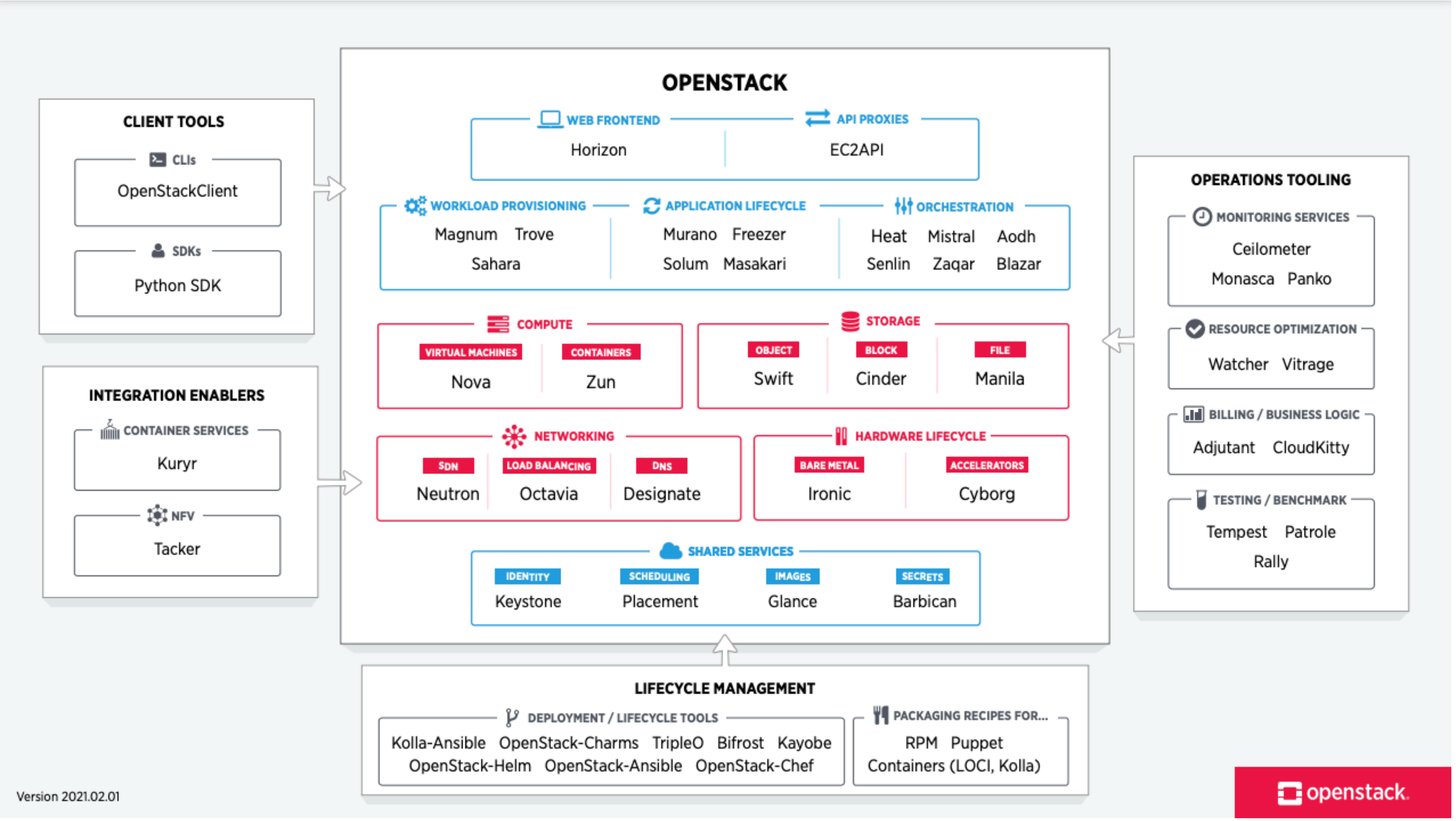
How does visible information in a code review affect the …

Evaluation Decision?

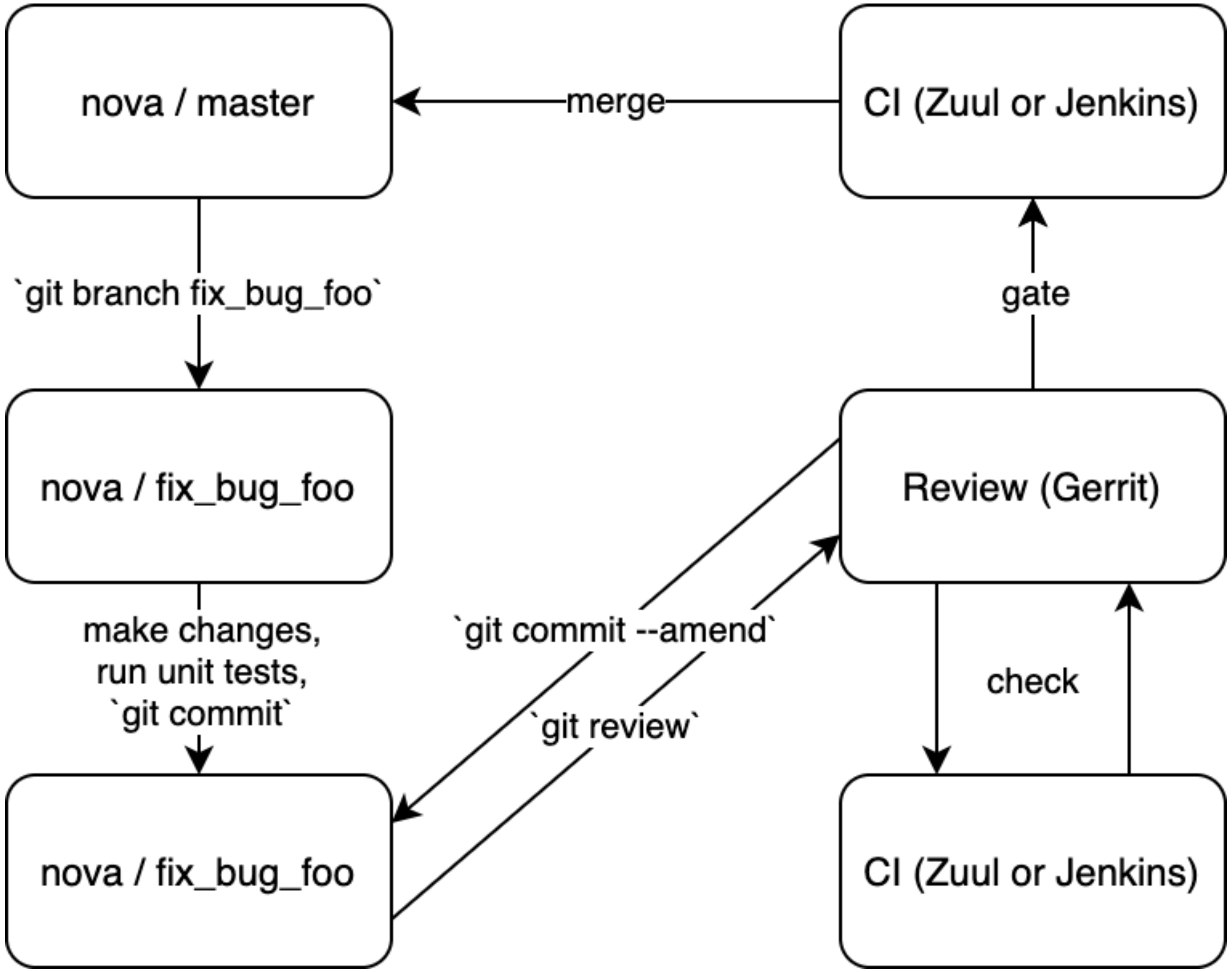
Defect Proneness?

OpenStack is one of the most active open-source projects

With ~30 components, OpenStack is the most widely deployed open-source cloud infrastructure software in the world.

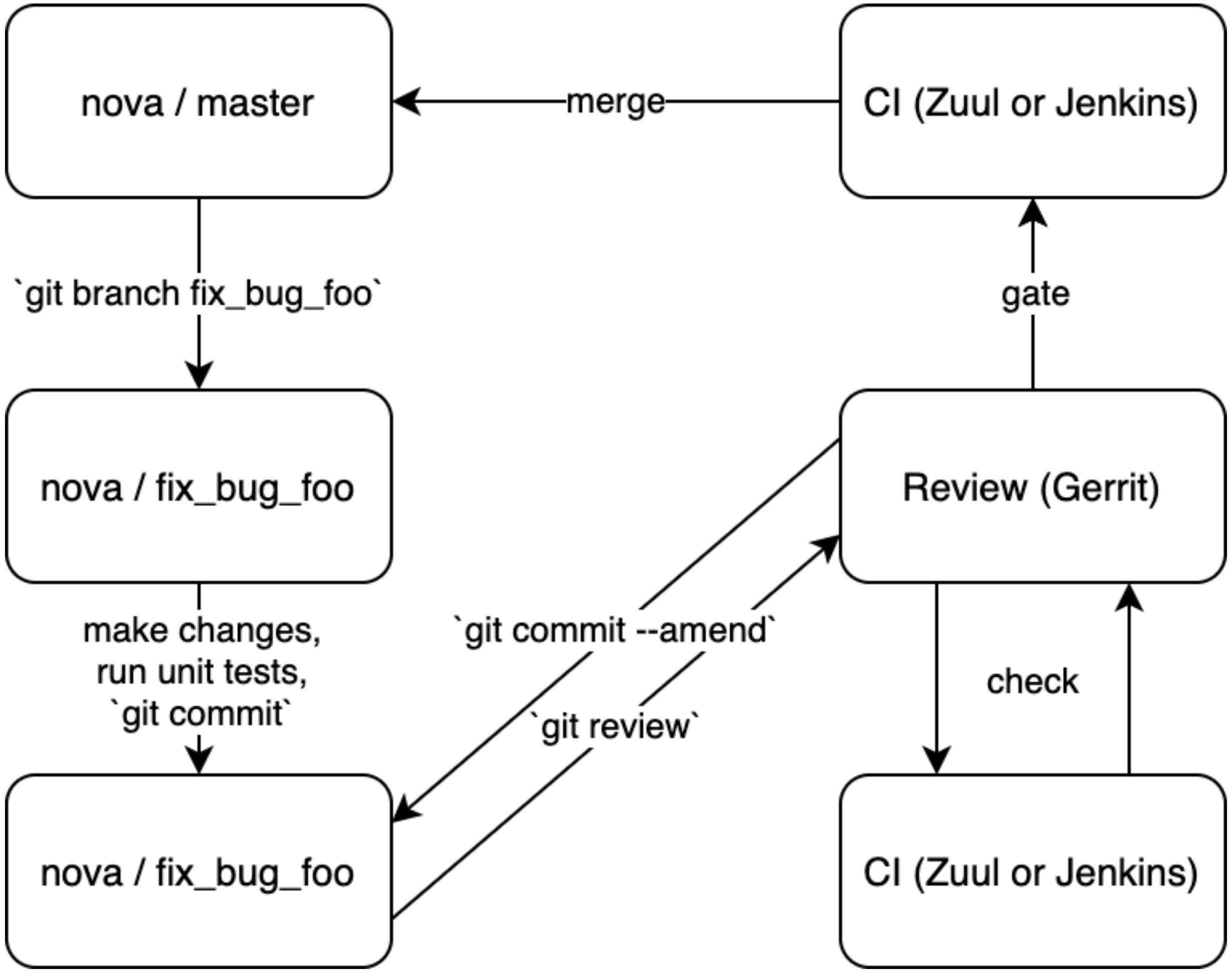


OpenStack Workflow



OpenStack Workflow

1. Create a work branch

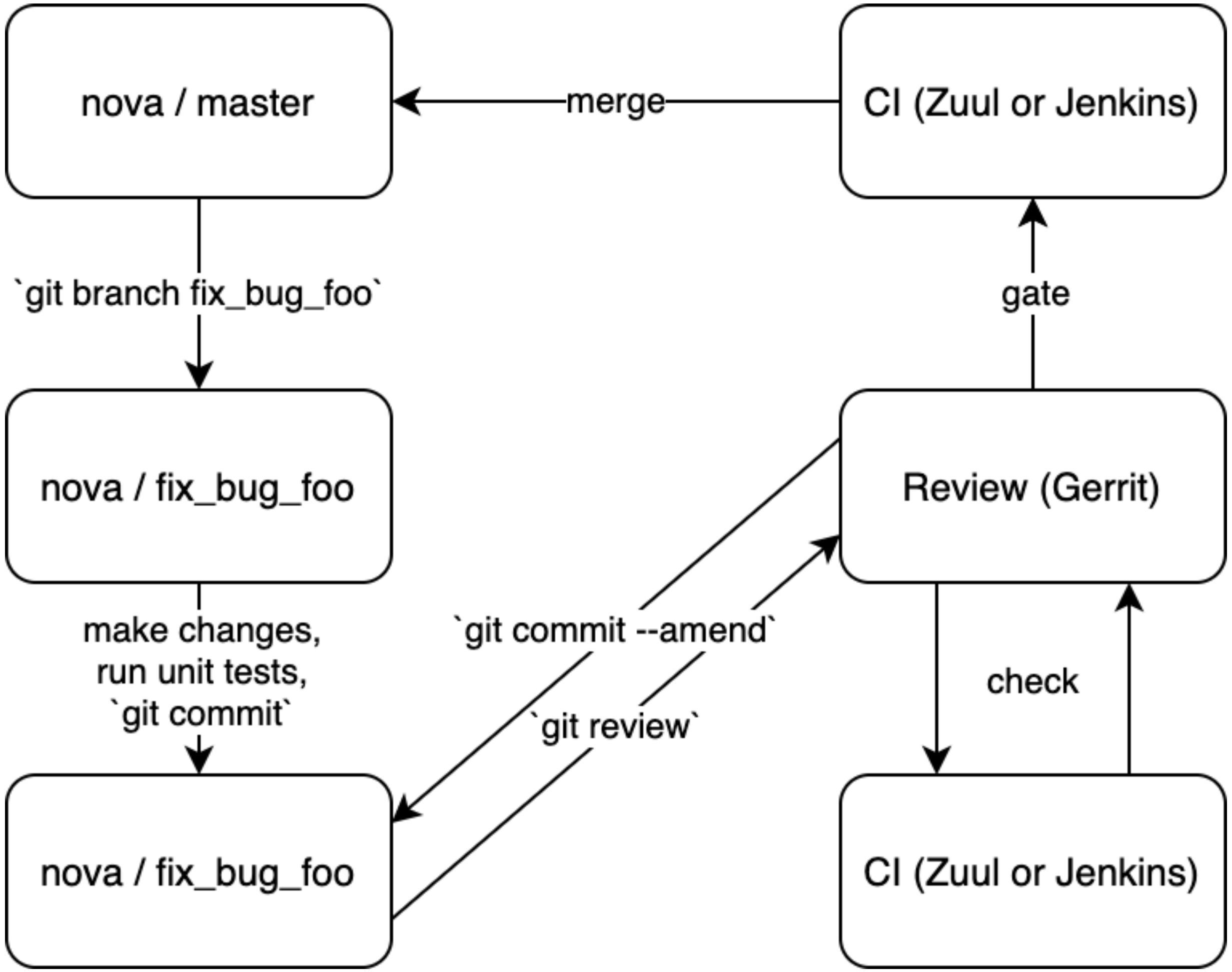


UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

OpenStack Workflow

1. Create a work branch

2. Propose to Gerrit for review

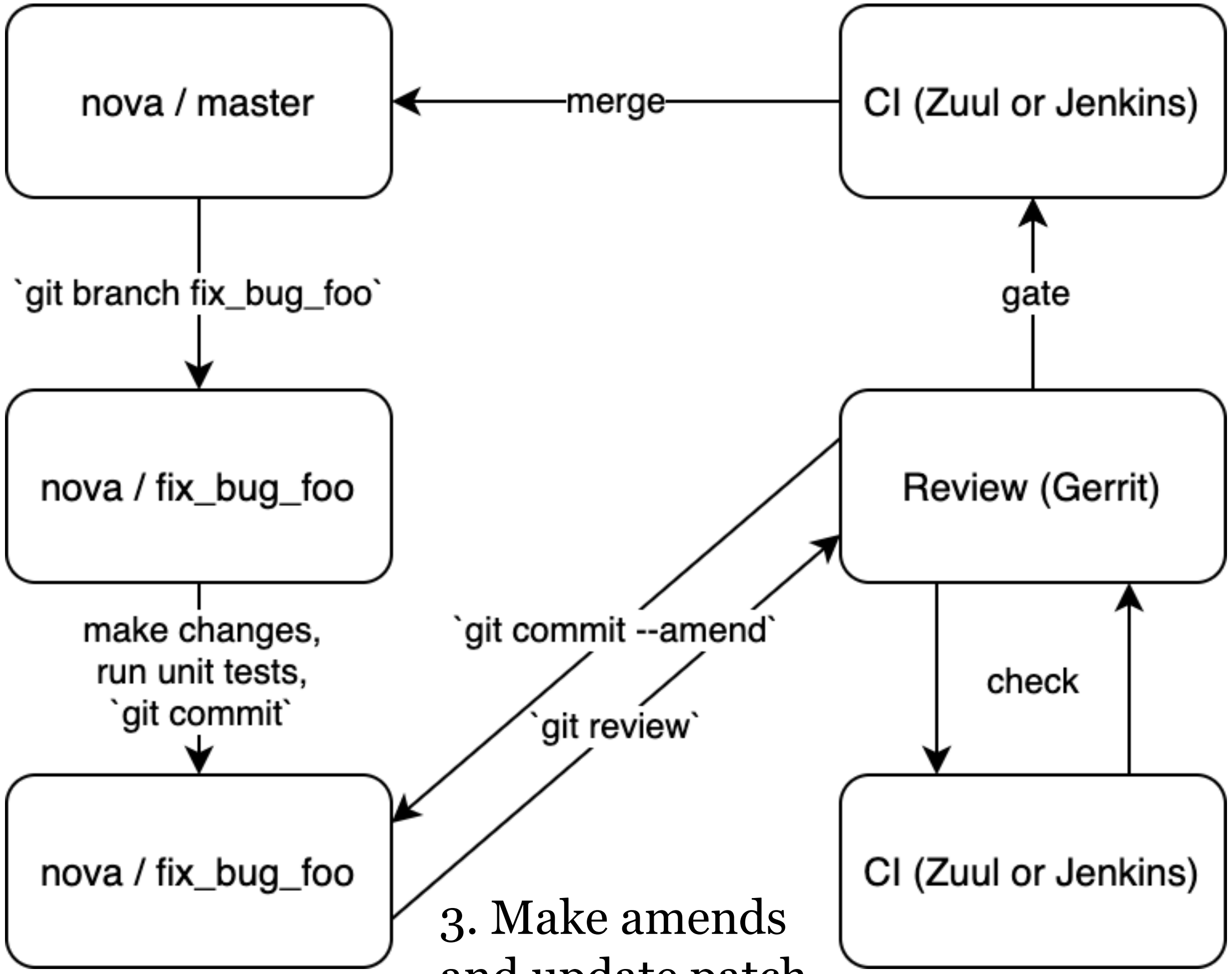


UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

OpenStack Workflow

1. Create a work branch

2. Propose to Gerrit for review

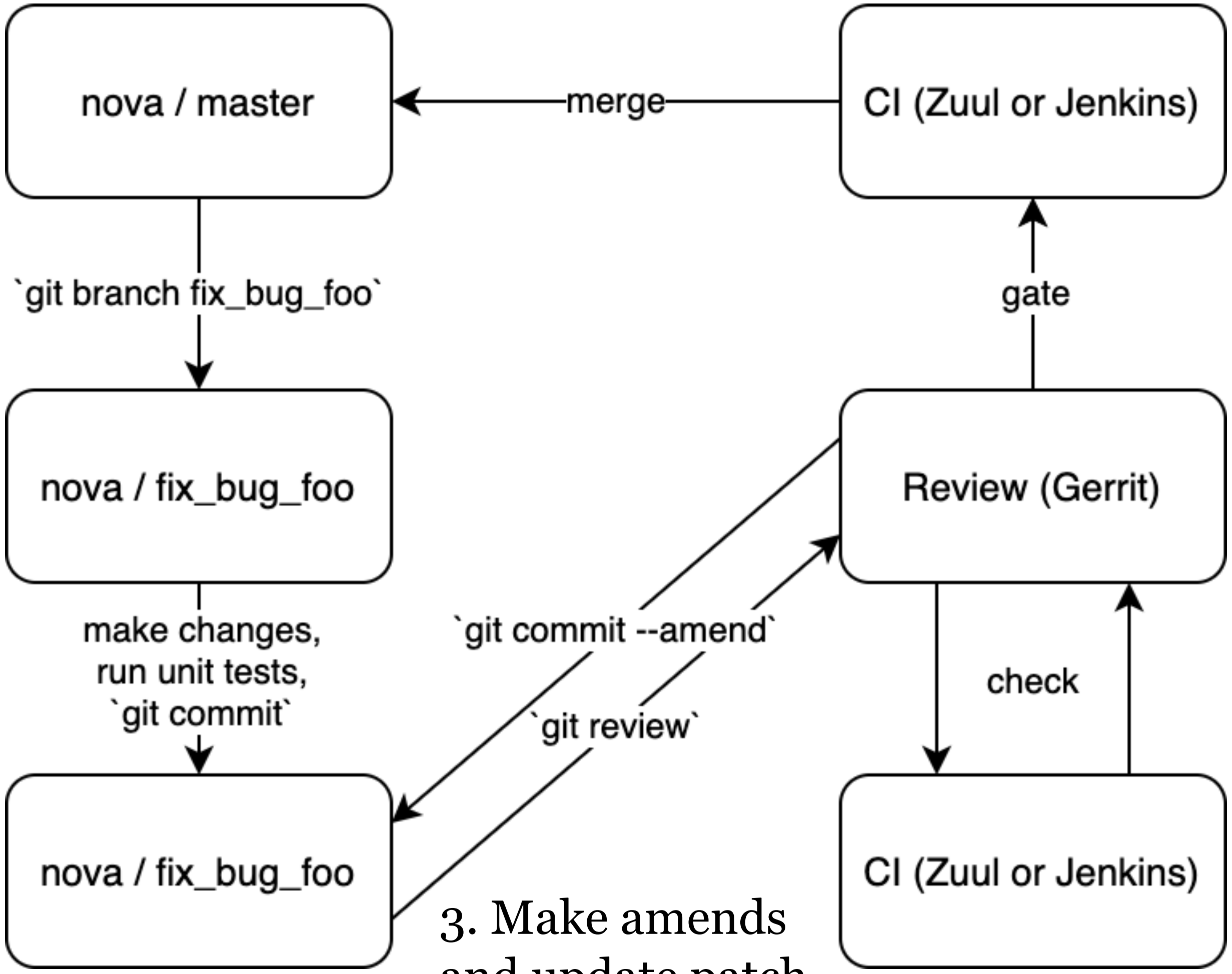


3. Make amends and update patch

3. Gets reviewed/voted on by reviewers

3. Runs check tests

OpenStack Workflow



1. Create a work branch

2. Propose to Gerrit for review

3. Make amends and update patch

3. Runs check tests

3. Gets reviewed/voted on by reviewers

4. Runs gate tests and merge

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Four projects from OpenStack are selected for the study

		Glance (Training)	Cinder (Training)	Neutron (Training)	Sahara (Validation)
	# Patches	2,936	8,518	10,575	3,164
	# Reviewers	626	1,246	1,332	245
	Avg # Reviewers per Patch	3.8	4.5	5.4	4.4
	% Patches w/ >1 Reviewer	94%	96%	100%	88%
RQ1 →	% Reviews w/ Positive Votes	92%	88%	90%	94%
RQ2 →	% Patches Fix Inducing	57%	52%	60%	50%

RQ1 – Review Dynamics

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Study Design

Data Extraction

Data Cleaning

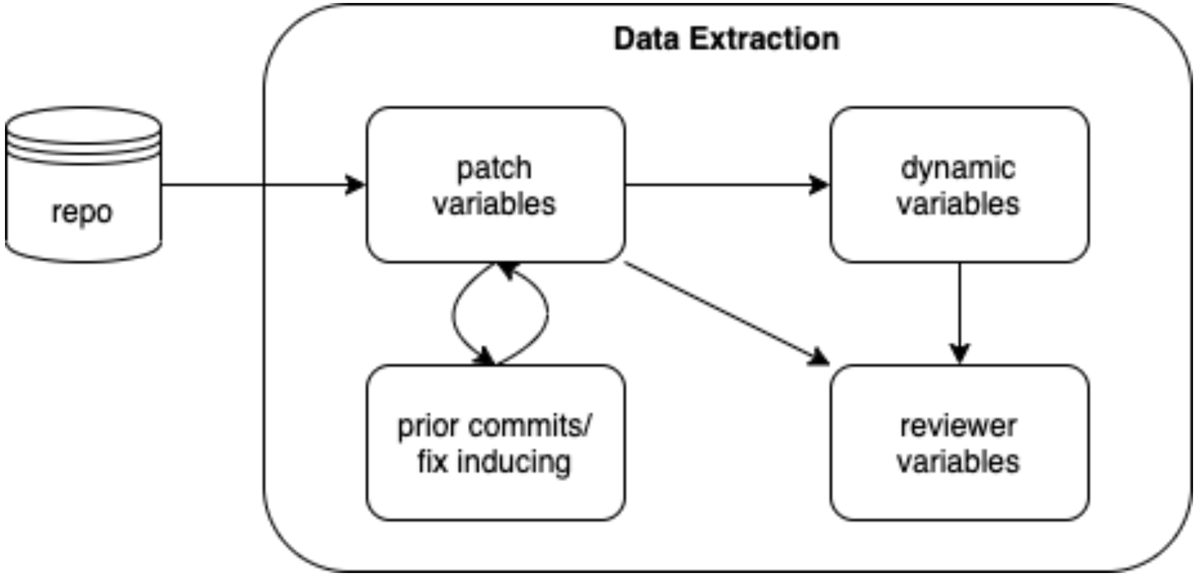
Model Training

Model Evaluation

Data Extraction

PyDriller is used to mine repositories.

Gerrit API is used to extract code review comments & votes.



Data is divided into three dimensions

Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	% Prior Negative Votes From Core Developers	% Prior Comments By Reviewer
Average Cyclomatic Complexity	# Prior Comments	Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Data is divided into three dimensions

Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	[illegible]	% Prior Comments By Reviewer
Average Cyclomatic Complexity		Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

From Hassan 2009, measures the dispersion in lines changed, normalized by number of files changed

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Data is divided into three dimensions

Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	% [illegible]e	% Prior Comments By Reviewer
Average Cyclomatic Complexity		Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

Regular expression search of keywords such as “fix”, “bug” and “defect”

Data is divided into three dimensions

Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	% [illegible] e	% Prior Comments By Reviewer
Average Cyclomatic Complexity		Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

Commits that last modified the same lines that the current commit modified

Data is divided into three dimensions

Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	% [illegible] e	% Prior Comments By Reviewer
Average Cyclomatic Complexity	[illegible]	Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

If current commit is bug fixing, then its prior commits are fix inducing

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Data is divided into three dimensions

Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	% [illegible] e	% Prior Comments By Reviewer
Average Cyclomatic Complexity		Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

Reviewer has authored/reviewed the prior commits

Data is divided into three dimensions

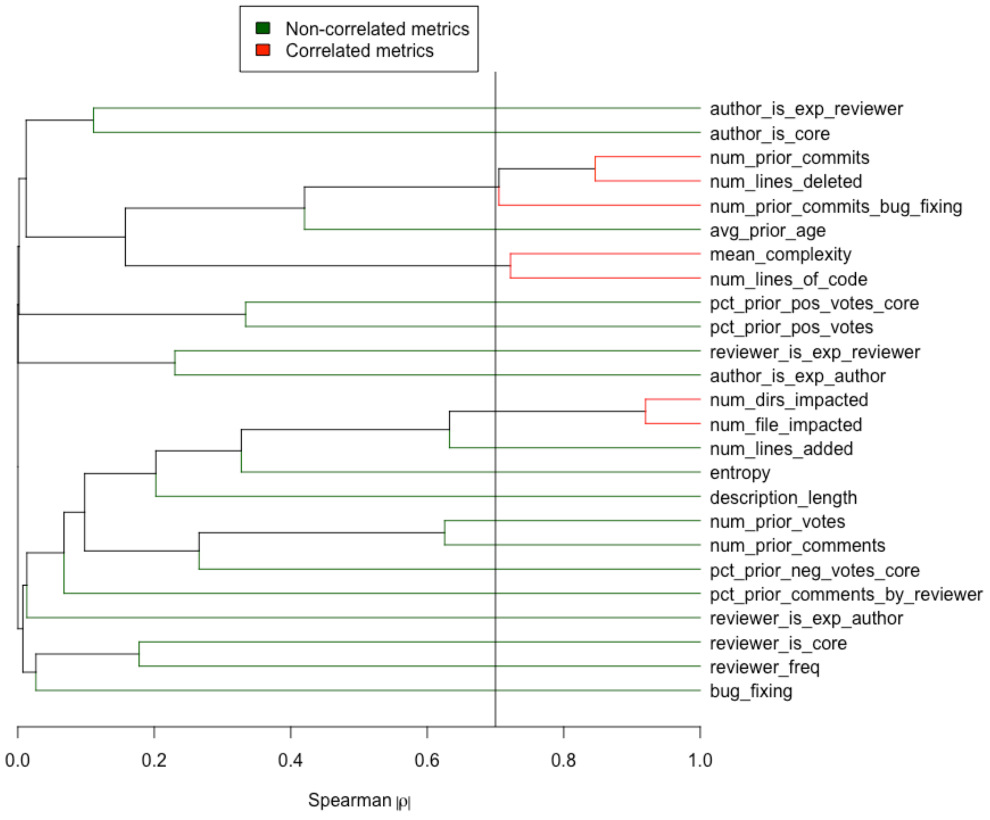
Patch	Dynamic	Reviewer
# Lines Added	# Prior Votes	Reviewer Is Core
# Files Impacted	% Prior Votes Positive	Reviewer Is Experienced Author
Entropy	% Prior Positive Votes From Core Developers	Reviewer Is Experienced Reviewer
Description Length	% [illegible] e	% Prior Comments By Reviewer
Average Cyclomatic Complexity		Reviewer Interaction Frequency w/ Author
Is Bug Fixing		
# Prior Commits		
Author Is Core		
… 7 more		

% of patches the author has written that were also reviewed by the reviewer

Variables that are highly correlated are removed

Spearman's rank correlation coefficient (ρ) with threshold of 0.7 is used.

Var A	Var B	|ρ|	Survived Var
# Directories Impacted	# Files Impacted	0.93	#Directories Impacted
# Prior Commits	# Lines Deleted	0.86	# Lines Deleted
Average Complexity	# Lines of Codes	0.73	Average Complexity
# Prior Commits Bug Fixing	# Lines Deleted	0.71	# Lines Deleted

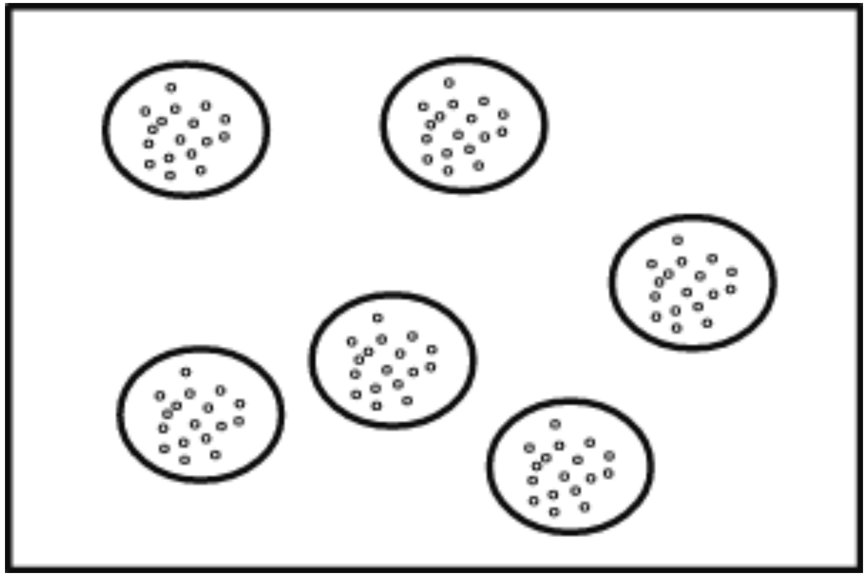


Linear Mixed Model (LMM) is used since our data is hierarchical

Linear Mixed Model (LMM) is an extension of simple linear model to allow both **fixed** and **random** effects.

LMMs are used when there is a **hierarchical structure** to the data. The variability in the outcome can be either within group, or between groups.

In our case, the random effects/groups are the **reviewers**.



Groups: reviewers
Dots: reviews

We train a full model, a null model, and three separate models excluding each data dimension

Since we want to understand how evaluation decision is affected, we use **positive vote** as our target (binary) variable.

Model	Formula	# Fixed Vars
Full	Positive Vote ~ Patch Vars + Dynamic Vars + Reviewer Vars + *(1 | Reviewer ID)*	21
Ex-Patch	Positive Vote ~ Dynamic Vars + Reviewer Vars + *(1 | Reviewer ID)*	10
Ex-Dynamic	Positive Vote ~ Patch Vars + Reviewer Vars + *(1 | Reviewer ID)*	16
Ex-Reviewer	Positive Vote ~ Patch Vars + Dynamic Vars + *(1 | Reviewer ID)*	16
Null	Positive Vote ~ 1 + *(1 | Reviewer ID)*	0

We evaluate the model performance using AUC

Larger the AUC (closer to 1), better the discriminant ability.

Model	AUC	X of Null AUC
Null	0.72	
Ex-Patch	0.81	1.12
Ex-Dynamic	0.77	1.06
Ex-Reviewer	0.82	1.14
Full	0.82	1.14

We estimate the explanatory power of each data dimension by performing log-likelihood ratio tests

Log-likelihood ratio tests assess the goodness of fit of two competing models based on the ratio of their likelihoods. Large LR means the two models are different.

Model A (Less Complex)	Model B (More Complex)	Δ D.F.	LR	% of Full LR
Null	Full	21	8,768	
Ex-Patch	Full	11	1,512	17%
Ex-Dynamic	Full	5	5,934	68%
Ex-Reviewer	Full	5	326	4%

Since reviewer characteristics do not offer significant performance increase, we use **Ex-Reviewer** model as our final model.

We estimate the explanatory power of each individual variable in the final model by calculating its Wald statistics

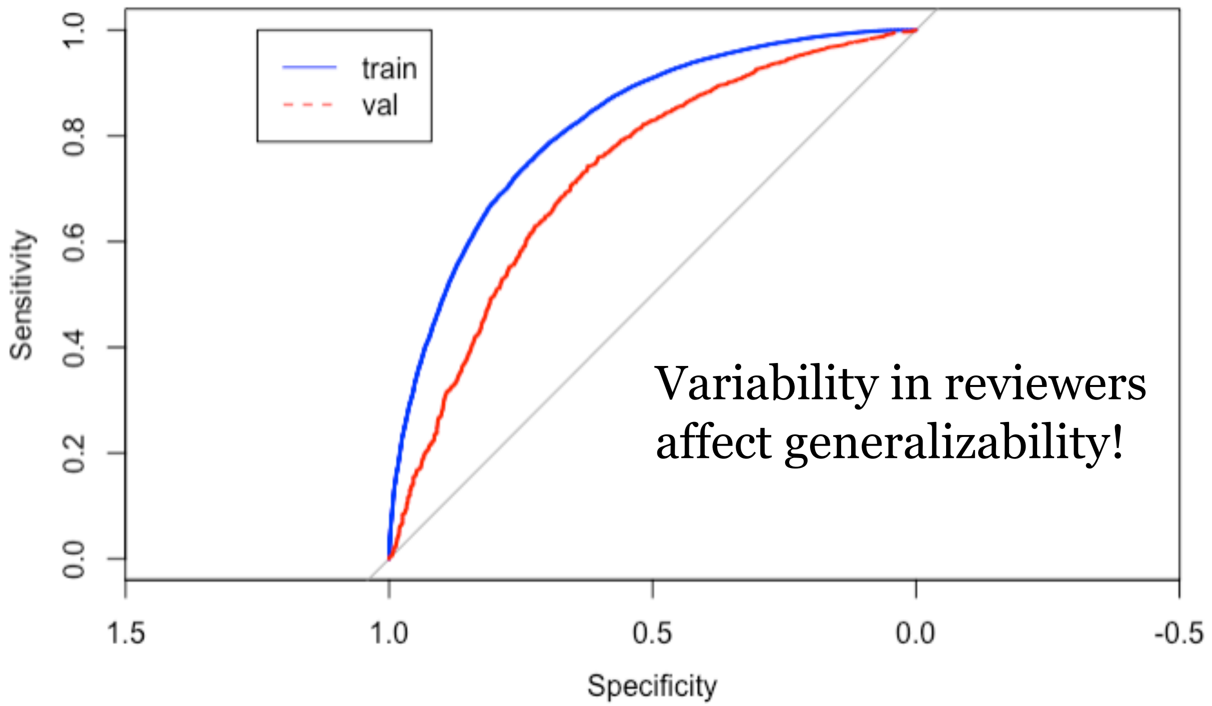
Patch Var	χ^2	Sign
# Lines Deleted	2,109	+
# Lines Added	547	-
Entropy	469	-
Description Length	239	-
Author Is Core	102	+
# Directories Impacted	86	-
Author Is Experienced Reviewer/Author	51/26	+/+

Dynamic Var	χ^2	Sign
% Prior Votes Positive	4,013	+
# Prior Votes	39	-
% Prior Negative/Positive Votes From Core Developers	33/23	-/+

Variables with p-value < 0.001 are shown

We validate the final model against the unseen project

Model	Training AUC	Validation AUC
Ex-Review	0.82	0.73



RQ1 Conclusion

Review dynamics, particularly the **proportion of prior positive votes** have a significant impact on the evaluation decision of a reviewer.

Unlike in the original paper, we do not observe significant association between the relationship (interaction frequency) with the patch author and the evaluation decision of a reviewer.

The final model also does not perform well on the validation dataset. This could be due to the random effect of the reviewers.

RQ2 – Defect Proneness

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Study Design

Metrics Formulation

Data Cleaning

Model Training

Model Evaluation

Based on results from RQ1, we formulate six “social” metrics for measuring review dynamics

Dynamic Variable	Sign	Social Metric
% Prior Votes Positive	+	% Positive Voters Consistent w/ Prior Positive Votes
% Prior Votes Negative	-	% Positive Voters Inconsistent w/ Prior Negative Votes
% Prior Positive Votes From Core Developers	+	% Positive Voters Consistent w/ Prior Core Positive Votes
% Prior Negative Votes From Core Developers	-	% Positive Voters Inconsistent w/ Prior Core Negative Votes
# Prior Votes	-	Average # Prior Votes for Positive Voters
# Prior Comments	-	Average # Prior Comments for Positive Voters

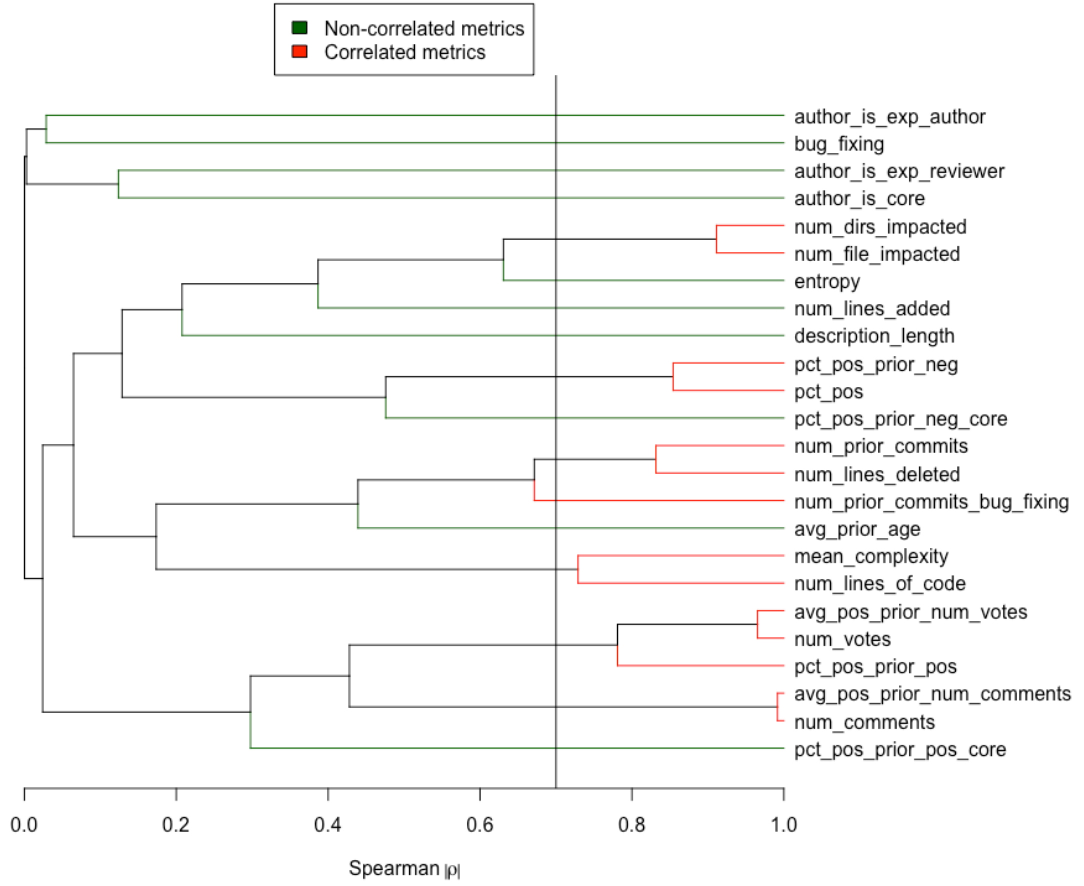
We also add patch characteristics from RQ1, and some aggregated review characteristics and combine each patch into one data point

Patch	Review	Social
# Lines Added	# Votes	% Positive Voters Consistent w/ Prior Positive Votes
# Files Impacted	# Comments	% Positive Voters Inconsistent w/ Prior Negative Votes
Entropy	% Positive Votes	% Positive Voters Consistent w/ Prior Core Positive Votes
Description Length		% Positive Voters Inconsistent w/ Prior Core Negative Votes
Average Cyclomatic Complexity		Average # Prior Votes for Positive Voters
Is Bug Fixing		Average # Prior Comments for Positive Voters
# Prior Commits		
Author Is Core		
… 7 more		

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Variables that are highly correlated are removed

Dimension	Removed Vars
Patch	# Files Impacted
Patch	# Prior Commits
Patch	# Lines of Code
Review	# Votes
Social	Average # Prior Votes for Positive Voters
Social	Average # Prior Comments for Positive Voters
Social	% Positive Voters Inconsistent w/ Prior Negative Votes



UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

We train a full model, a null model, and three separate models excluding each data dimension

We use **fix inducing** as our target (binary) variable. Note that we are using generic GLM instead of mixed-effect linear model.

Model	Formula	# Vars
Full	Fix Inducing ~ Patch Vars + Review Vars + Social Vars	17
Ex-Patch	Positive Vote ~ Review Vars + Social Vars	5
Ex-Review	Fix Inducing ~ Patch Vars + Social Vars	15
Ex-Social	Fix Inducing ~ Patch Vars + Review Vars	14
Null	Fix Inducing ~ 1	0

Model performance

Model	AUC	X of Null AUC
Null	0.500	
Ex-Patch	0.665	1.33
Ex-Review	0.783	1.57
Ex-Social	0.777	1.55
Full	0.784	1.57

Log-likelihood ratio tests

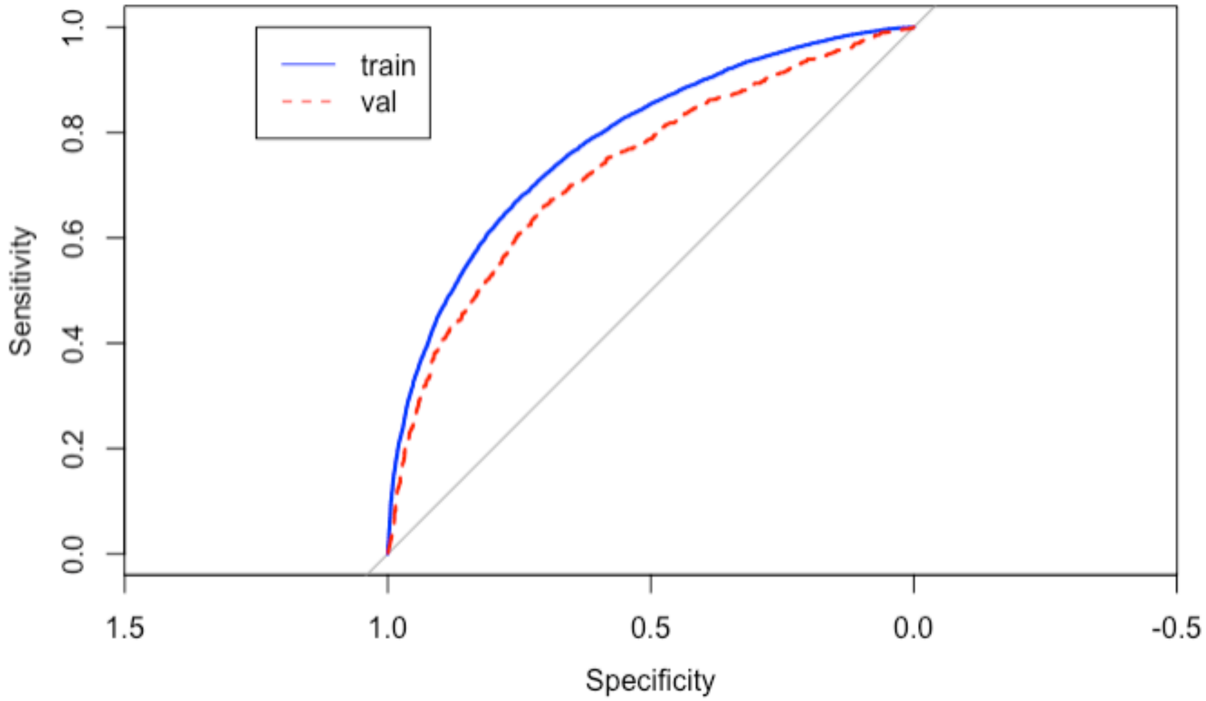
Model A (Less Complex)	Model B (More Complex)	Δ D.F.	LR	% of Full LR
Null	Full	17	5,402	
Ex-Patch	Full	12	3,373	62%
Ex-Review	Full	2	98	2%
Ex-Social	Full	3	359	7%

Since review characteristics do not offer significant performance increase, we use **Ex-Review** model as our final model.

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

We validate the final model against the unseen project

Model	Training AUC	Validation AUC
Ex-Review	0.78	0.73



Final model Wald statistics

Patch Var	χ^2	Sign
Average Prior Commits Age	494	-
Entropy	410	+
# Lines Added	298	+
Is Bug Fixing	127	+
# Prior Commits Bug Fixing	92	+
Description Length	70	-
# Directories Impacted	69	+
# Lines Deleted	31	-

Social Metrics	χ^2	Sign
% Positive Voters Consistent w/ Prior Core Positive Votes	232	+
% Positive Voters Consistent w/ Prior Positive Votes	174	-
% Positive Voters Inconsistent w/ Prior Core Negative Votes	45	-

Variables with p-value < 0.001 are shown

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

RQ2 Conclusion

Review dynamic metrics and the likelihood of inducing fixes do not have as strong of an association as those of patch characteristics.

UNIVERSITY OF WATERLOO | FACULTY OF MATHEMATICS

Discussion

Our study shows several implications in an open code review

Dynamics	Patch	Reviewers
Reviewers tend to adhere to opinions of the community. However, this has little impact on the patch qualities.	Reviewers tend to prefer patches with low entropy and small modifications. High entropy is also associated with high likelihood of defects.	There is no evidence of strong association between the reviewer's own characteristics (including past relationship with the author) and the vote outcome.

Some personal lessons learned

Data	Model
Always use virtual environments to ensure package consistencies and reproducibility.	LMM is useful in situations with random effects.
Conduct thorough spot-checks on data points, covering all possible scenarios, to make sure that the data pipeline is correct.	Always conduct validation against unseen datasets to check for generalizability.