Investigating the Validity of Ground Truth in Code Reviewer Recommendation Studies

Emre Doğan, Eray Tüzün, K. Ayberk Tecimer, H. Altay Güvenir

MSc Student
Computer Engineering Department
Bilkent University
Ankara, TURKEY

ESEM 2019
19th September 2019

What is Code Review, Who is a Code Reviewer?

Code Review: A systematic examination of source code in order to highlight bugs and enhance the code quality.

Code Reviewer: The developer performing a code review who ensures the quality of the code.

What is Code Review, Who is a Code Reviewer?

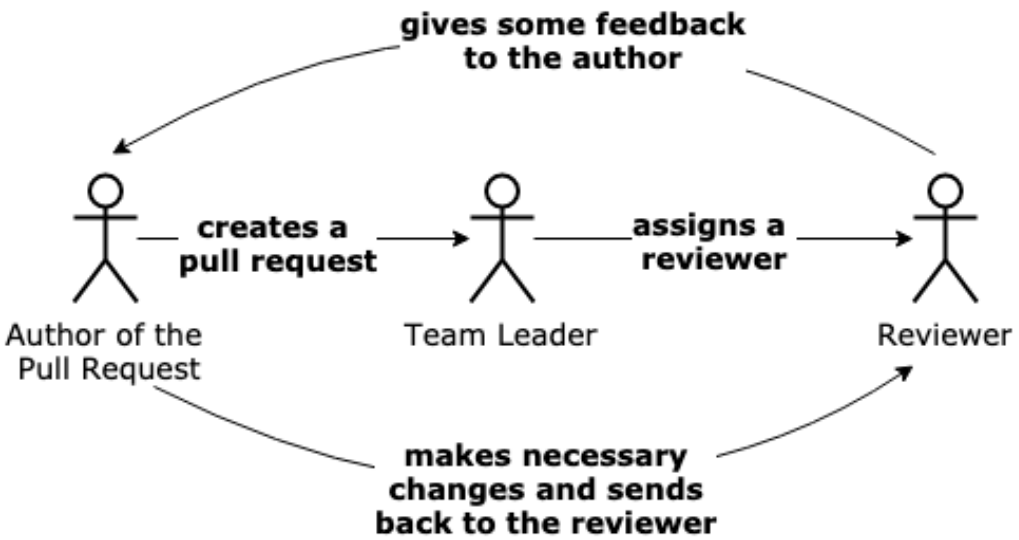
Code Review: A systematic examination of source code in order to highlight bugs and enhance the code quality.

Code Reviewer: The developer performing a code review who ensures the quality of the code.

What is Code Review, Who is a Code Reviewer?

Code Review: A systematic examination of source code in order to highlight bugs and enhance the code quality.

Code Reviewer: The developer performing a code review who ensures the quality of the code.

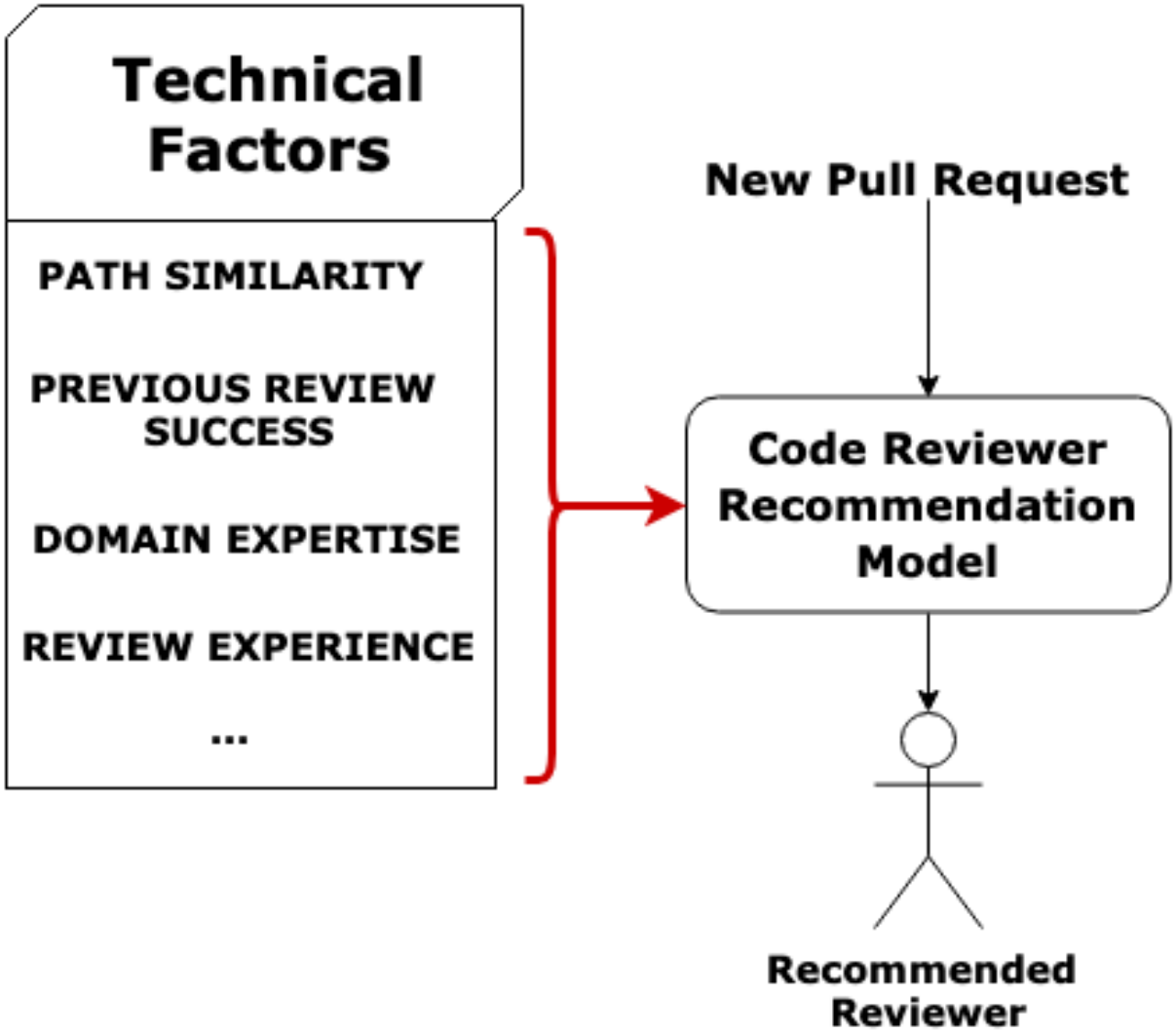


A typical code review scenario

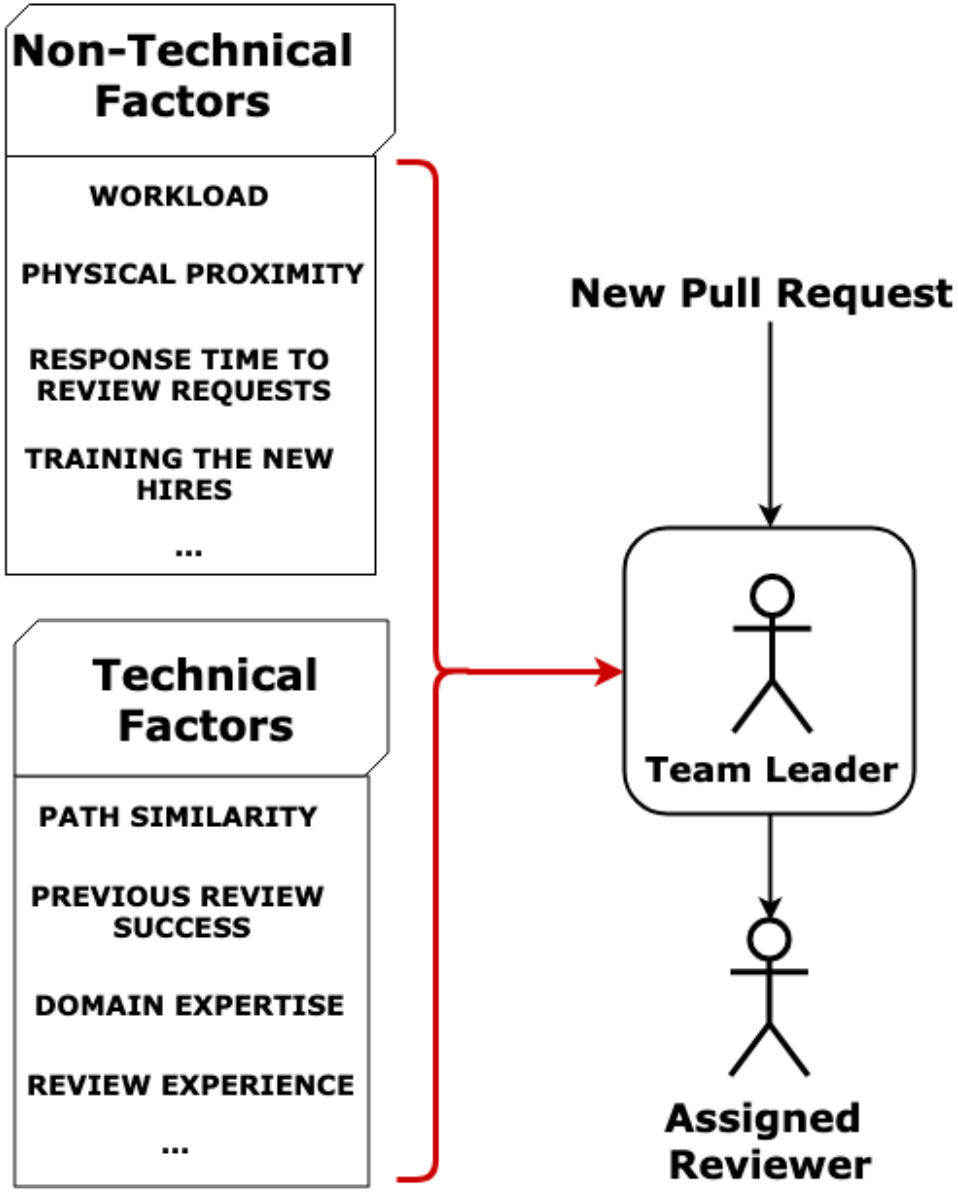
How to find an ideal code reviewer?

- Code reviewer recommendation models/tools help us to choose ideal reviewers.
- These tools help software teams:
  - to find reviewers who can find more(critical) bugs in the source code.
  - to speed up the code review process.

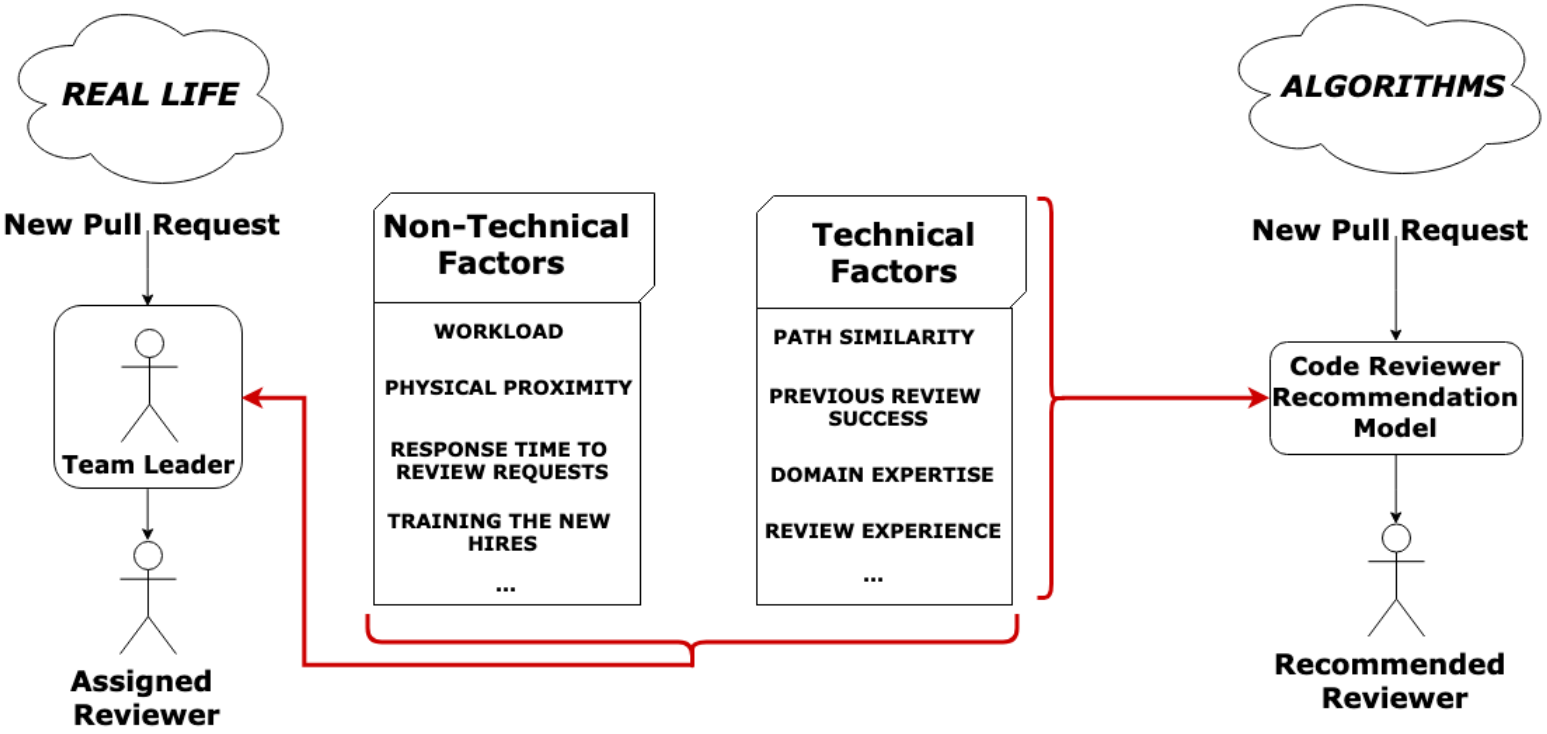
Reviewer Selection in Recommendation Models



Reviewer Selection in Real Life



Comparison of Real Life and Algorithms



Notice:

- There exists a discrepancy between real life and algorithm based reviewer selection process.
- This discrepancy creates a **ground truth problem** in code reviewer recommendation studies and datasets.

Ground Truth

Ground Truth:

- Factual data that has been observed or measured.
- If data stands on some assumptions which is subject to opinion, then it cannot be **ground truth data**.

Ground Truth in Software Engineering:

- The more human aspects involved, the more tendency to the ground truth problems.
- Many fields of empirical software engineering research suffer from the ground truth problem. (i.e. code reviewer recommendation, bug report assignee recommendation, etc.)

Ground Truth

Ground Truth:

- Factual data that has been observed or measured.
- If data stands on some assumptions which is subject to opinion, then it cannot be **ground truth data**.

Ground Truth in Software Engineering:

- The more human aspects involved, the more tendency to the ground truth problems.
- Many fields of empirical software engineering research suffer from the ground truth problem. (i.e. code reviewer recommendation, bug report assignee recommendation, etc.)

Ground Truth

Ground Truth in Code Reviewer Recommendation Studies:

- Recommendation models rely on the real-life assignments.
- These assignments are assumed to be ideal.
- Studies in real-life projects show that code reviewers are not usually assigned with the aim of finding the ideal one.

Who is an ideal reviewer?

Ideal Reviewer:

The theoretical best possible reviewer in the team that would improve or preferably perfect (such as pointing out all the defects) the pull request under review.

Warning:

- Ideal reviewer is selected by considering **only technical factors**.
- I.e., If a developer is considered as the ideal reviewer for a pull request but is not available for a review at that moment, he/she is still the ideal reviewer.

Who is an ideal reviewer?

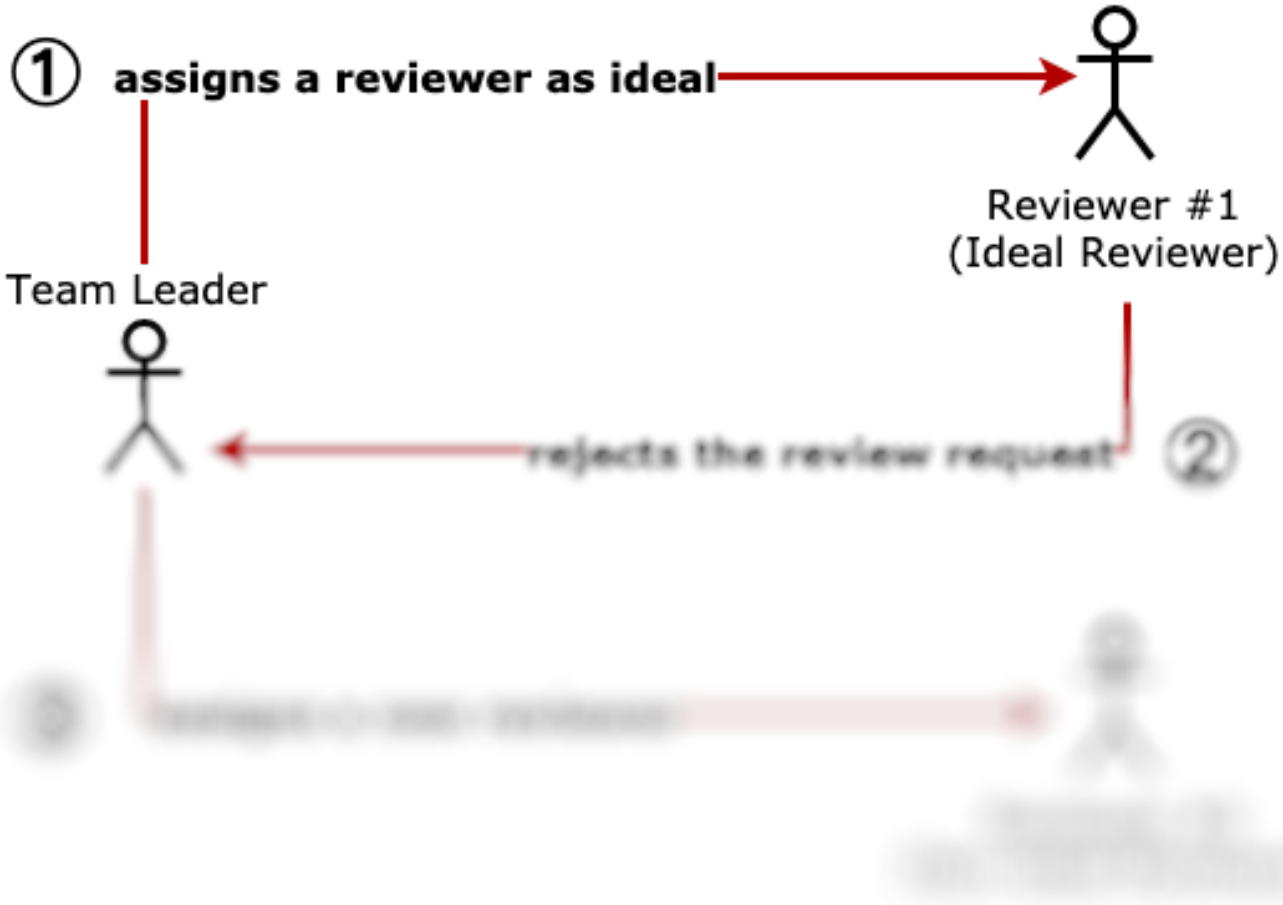
Ideal Reviewer:

The theoretical best possible reviewer in the team that would improve or preferably perfect (such as pointing out all the defects) the pull request under review.

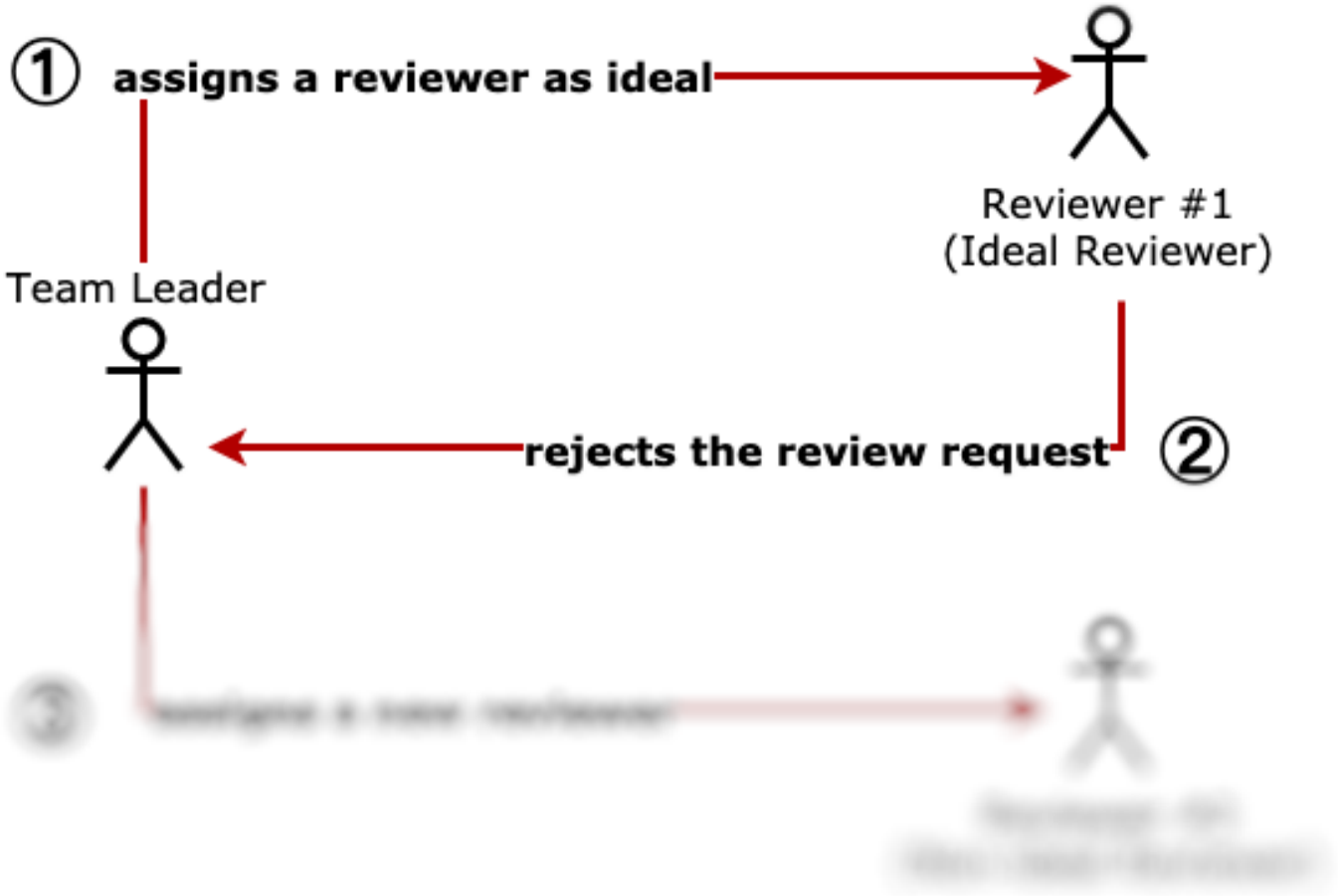
Warning:

- Ideal reviewer is selected by considering **only technical factors**.
- I.e., If a developer is considered as the ideal reviewer for a pull request but is not available for a review at that moment, he/she is still the ideal reviewer.

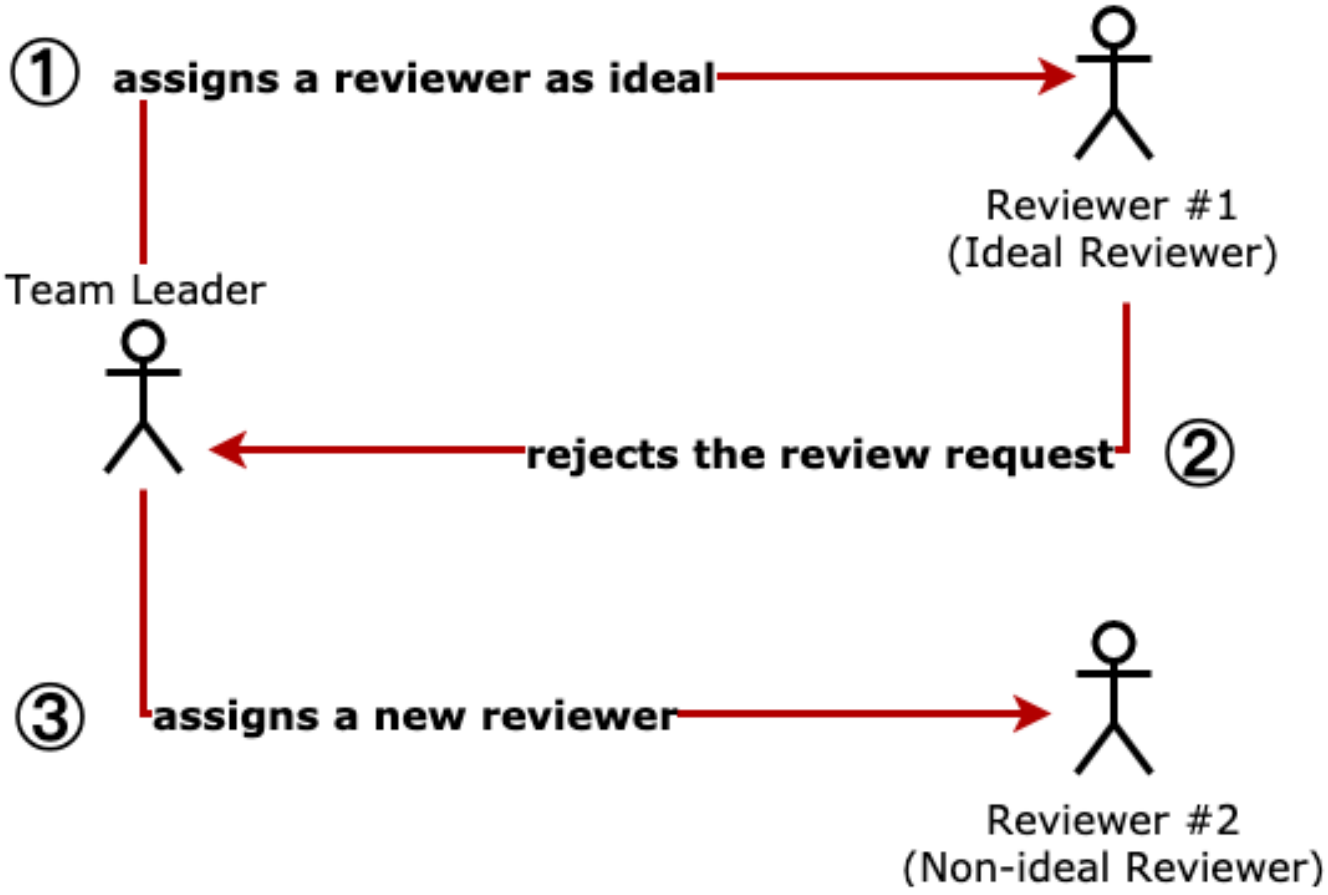
Problematical Reviewer Selection Scenario



Problematical Reviewer Selection Scenario



Problematical Reviewer Selection Scenario

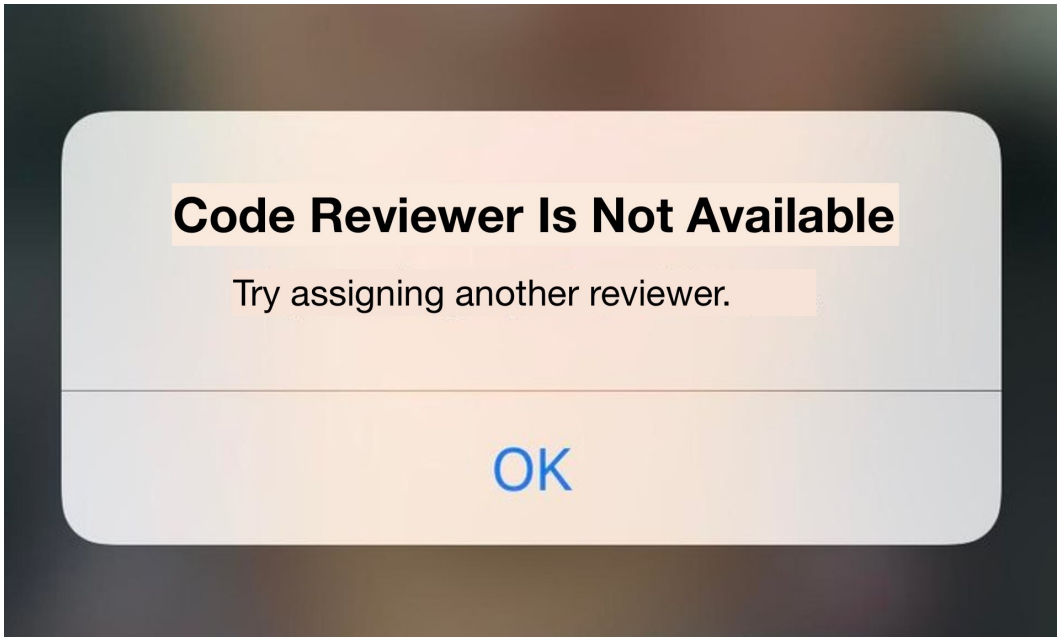


What causes a non-ideal reviewer assignment?

Availability Reasons:

The ideal reviewer might be...

- physically absent from work, so he/she cannot review the pull request.
- busy with some other tasks, so he/she declines to review the pull request.
- busy with some other tasks and is late to reply the review request.



Quantitative Evidence from Real Life

Project Name	Total Number of Pull Requests	Number of PRs with at least one non-responsive reviewer	The ratio of PRs having at least one non-responsive reviewer
Android	36,771	24,367	66%
LibreOffice	18,716	3,039	16%
Open Stack	108,788	24,589	23%
Qt	65,815	30,630	47%
TOTAL	230,090	82,625	36%

Table: An Analysis of Pull Request Reviews from 4 Large OSS Projects[1]

Notice

The results illustrate that 36% of pull requests suffer from the *availability reasons*.

[1] S. Ruangwan, P. Thongtanunam, A. Ihara, and K. Matsumoto, "The impact of human factors on the participation decision of reviewers in modern code review," *Empirical Software Engineering*, vol. 24, no. 2,pp. 973-1016, 2019. pp. 973–1016, 2019.

What causes a non-ideal reviewer assignment?

Cognitive Bias:

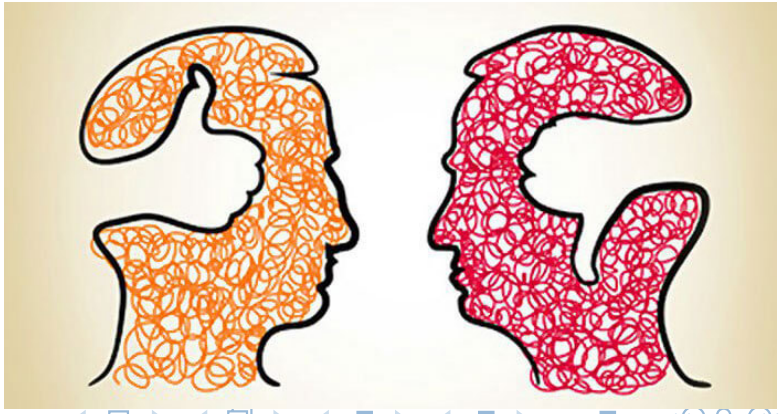
A systematic pattern of deviation from norm or rationality in judgment.

Attribute Substitution:

It occurs when an individual has to make **a computationally complex judgment**, and instead substitutes **a more easily calculated heuristic attribute**.

The team leader prefers to assign...

- a volunteer for the review.
- a reviewer based on their work schedule.
- a new hire as a reviewer for educational purposes.
- a developer based on their relative response time to review requests.



What causes a non-ideal reviewer assignment?

Cognitive Bias:

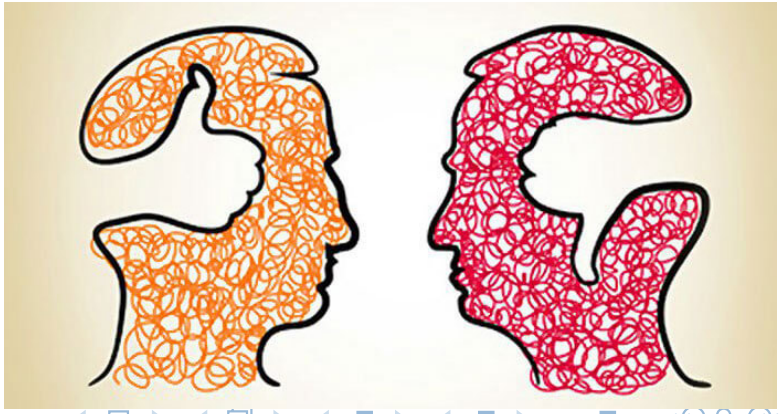
A systematic pattern of deviation from norm or rationality in judgment.

Attribute Substitution:

It occurs when an individual has to make **a computationally complex judgment**, and instead substitutes **a more easily calculated heuristic attribute**.

The team leader prefers to assign...

- a volunteer for the review.
- a reviewer based on their work schedule.
- a new hire as a reviewer for educational purposes.
- a developer based on their relative response time to review requests.



What causes a non-ideal reviewer assignment?

Cognitive Bias:

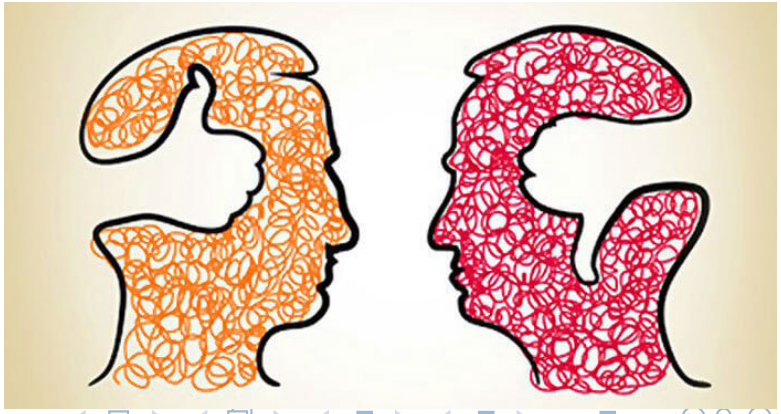
A systematic pattern of deviation from norm or rationality in judgment.

Attribute Substitution:

It occurs when an individual has to make **a computationally complex judgment**, and instead substitutes **a more easily calculated heuristic attribute**.

The team leader prefers to assign...

- a volunteer for the review.
- a reviewer based on their work schedule.
- a new hire as a reviewer for educational purposes.
- a developer based on their relative response time to review requests.



Implications of This Study

- Previous reviewer recommendation studies and datasets should be reviewed in terms of the validity of the ground truth.
- New recommendation models and datasets should be created by considering this validity problem.
- A validated benchmark dataset for reviewer recommendation task should be created.

Implications of This Study

- Previous reviewer recommendation studies and datasets should be reviewed in terms of the validity of the ground truth.
- New recommendation models and datasets should be created by considering this validity problem.
- A validated benchmark dataset for reviewer recommendation task should be created.

Implications of This Study

- Previous reviewer recommendation studies and datasets should be reviewed in terms of the validity of the ground truth.
- New recommendation models and datasets should be created by considering this validity problem.
- A validated benchmark dataset for reviewer recommendation task should be created.

Conclusion

- The validation of real-life collected reviewer datasets are problematic.
- The validation problem of these datasets affect the validity of recommendation models.

Conclusion

- The validation of real-life collected reviewer datasets are problematic.
- The validation problem of these datasets affect the validity of recommendation models.

Future Work

- Introducing quantitative evidence for cognitive bias.
- Establishing ground truth data by alternative solutions.
- Check our paper for solution proposals:
  - Setting up an Experiment in Real Life
  - Forward-Looking Mining the Issue Repository

Future Work

- Introducing quantitative evidence for cognitive bias.
- Establishing ground truth data by alternative solutions.
- Check our paper for solution proposals:
  - Setting up an Experiment in Real Life
  - Forward-Looking Mining the Issue Repository

Future Work

- Introducing quantitative evidence for cognitive bias.
- Establishing ground truth data by alternative solutions.
- Check our paper for solution proposals:
  - Setting up an Experiment in Real Life
  - Forward-Looking Mining the Issue Repository

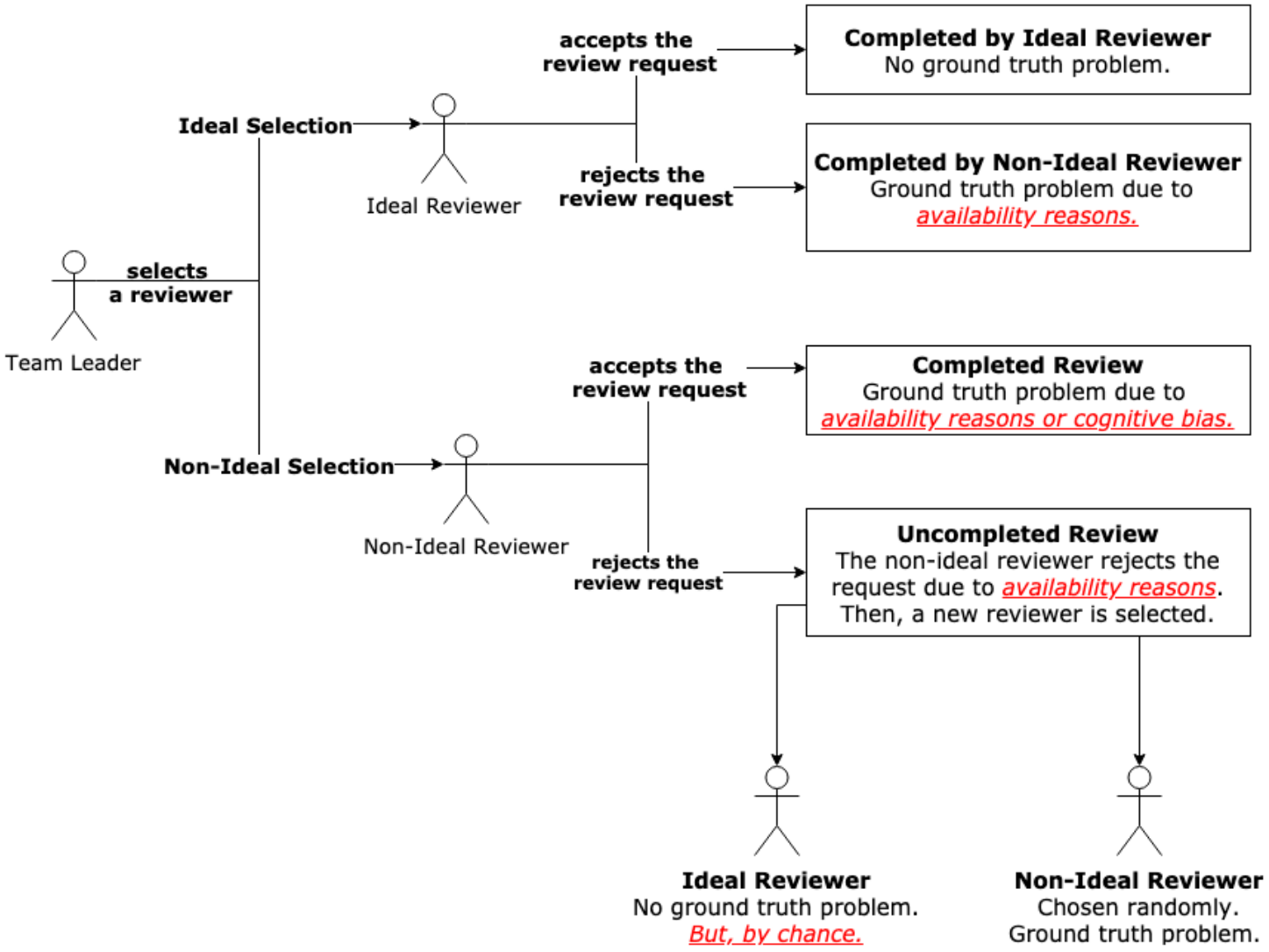
Thank you

Emre Doğan
MSc Student
Computer Engineering Department
Bilkent University
Ankara, Turkey
emre.dogan@bilkent.edu.tr

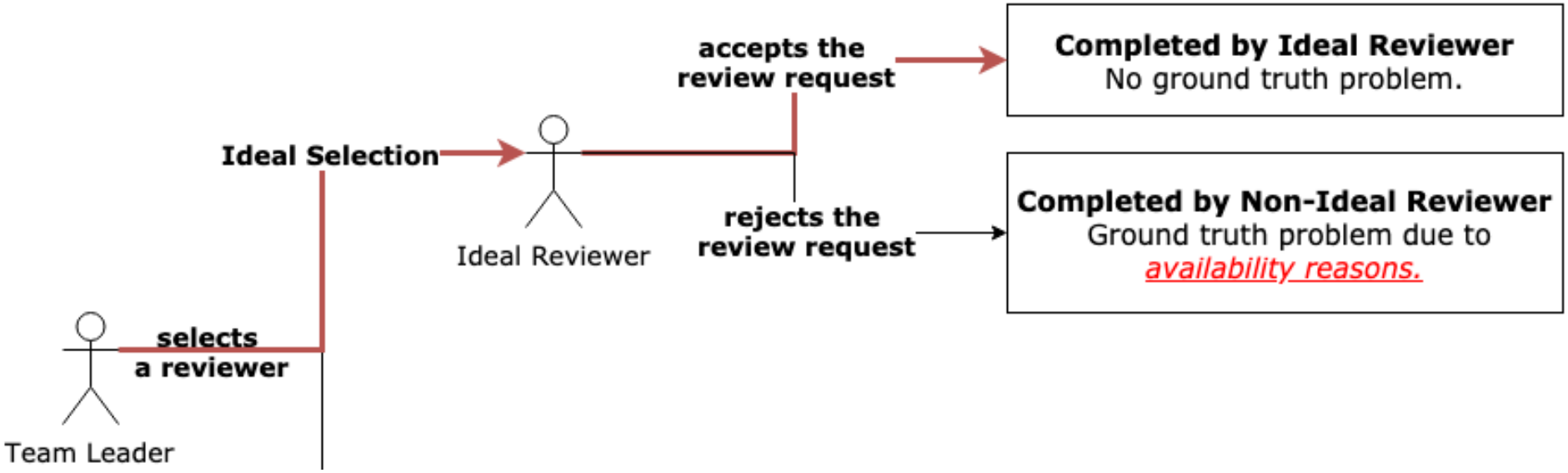
https://bit.ly/2koV4Jk

Backup Slides

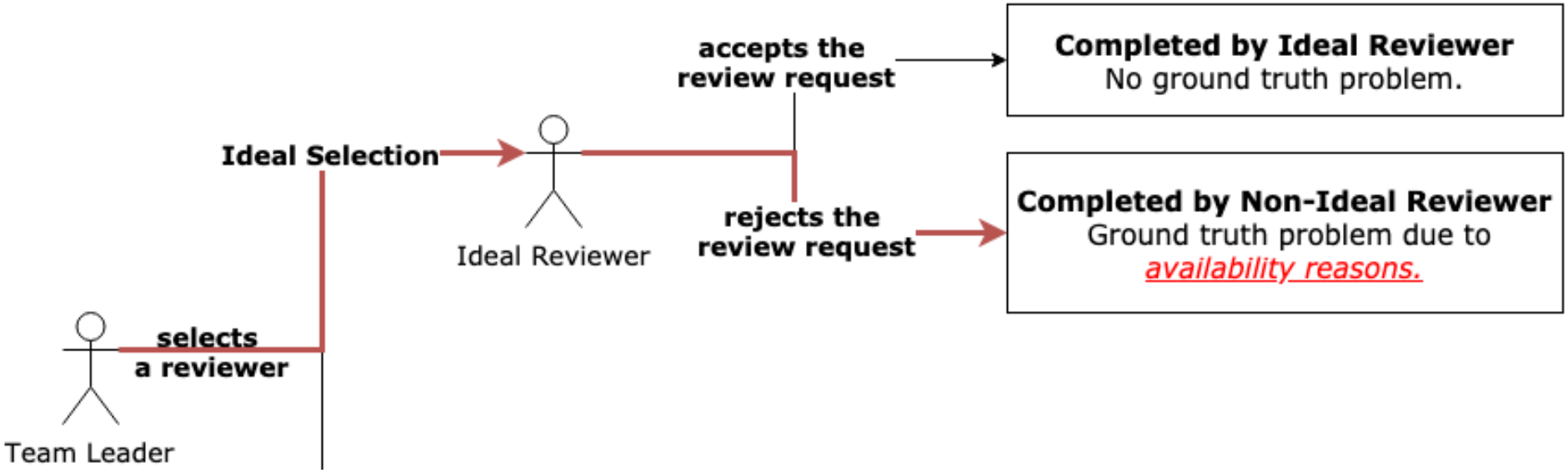
Possible Reviewer Assignment Scenarios



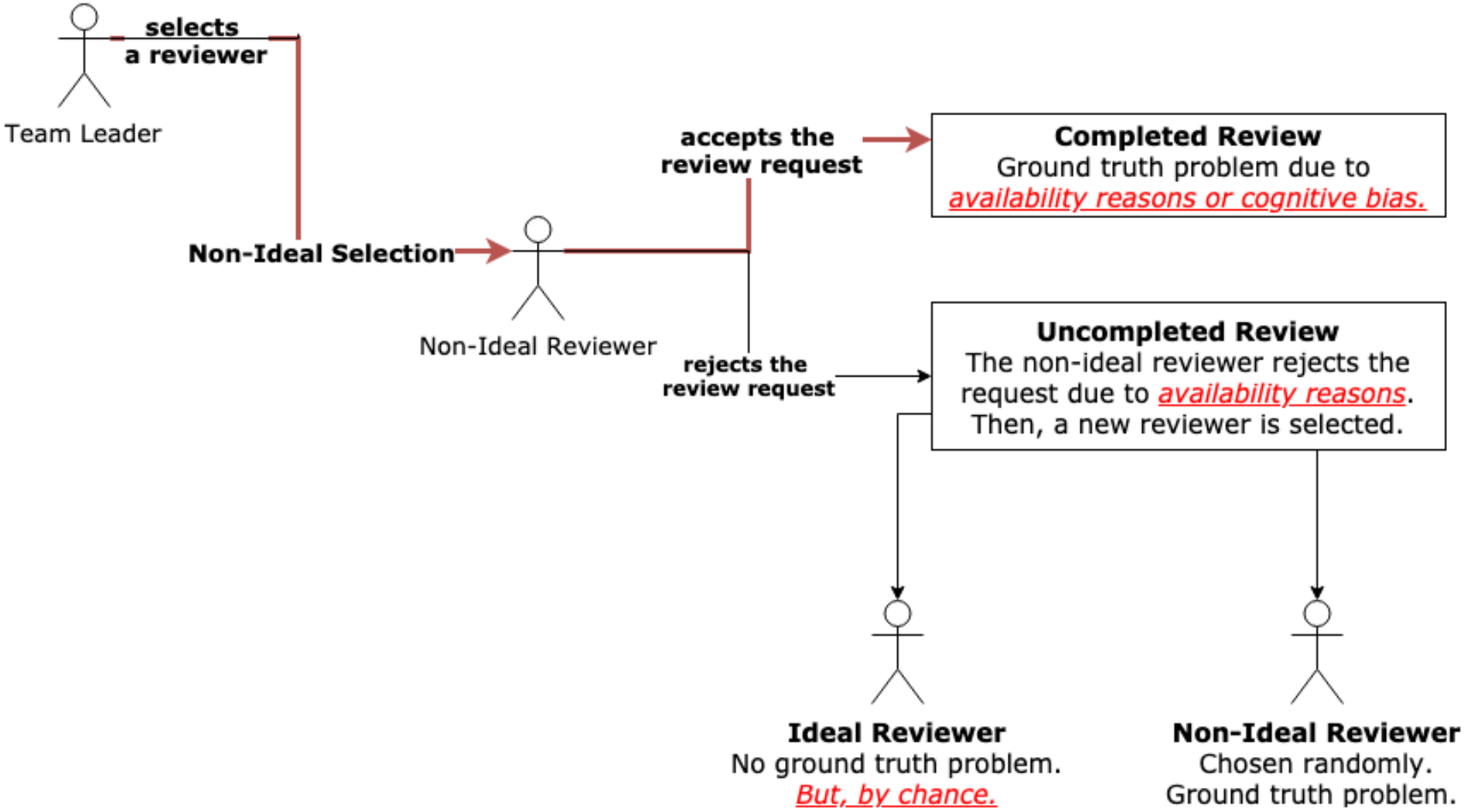
Scenario 1


accepts the
review request
Completed by Ideal Reviewer
No ground truth problem.
Ideal Selection
Ideal Reviewer
rejects the
review request
Completed by Non-Ideal Reviewer
Ground truth problem due to
availability reasons.
selects
a reviewer
Team Leader

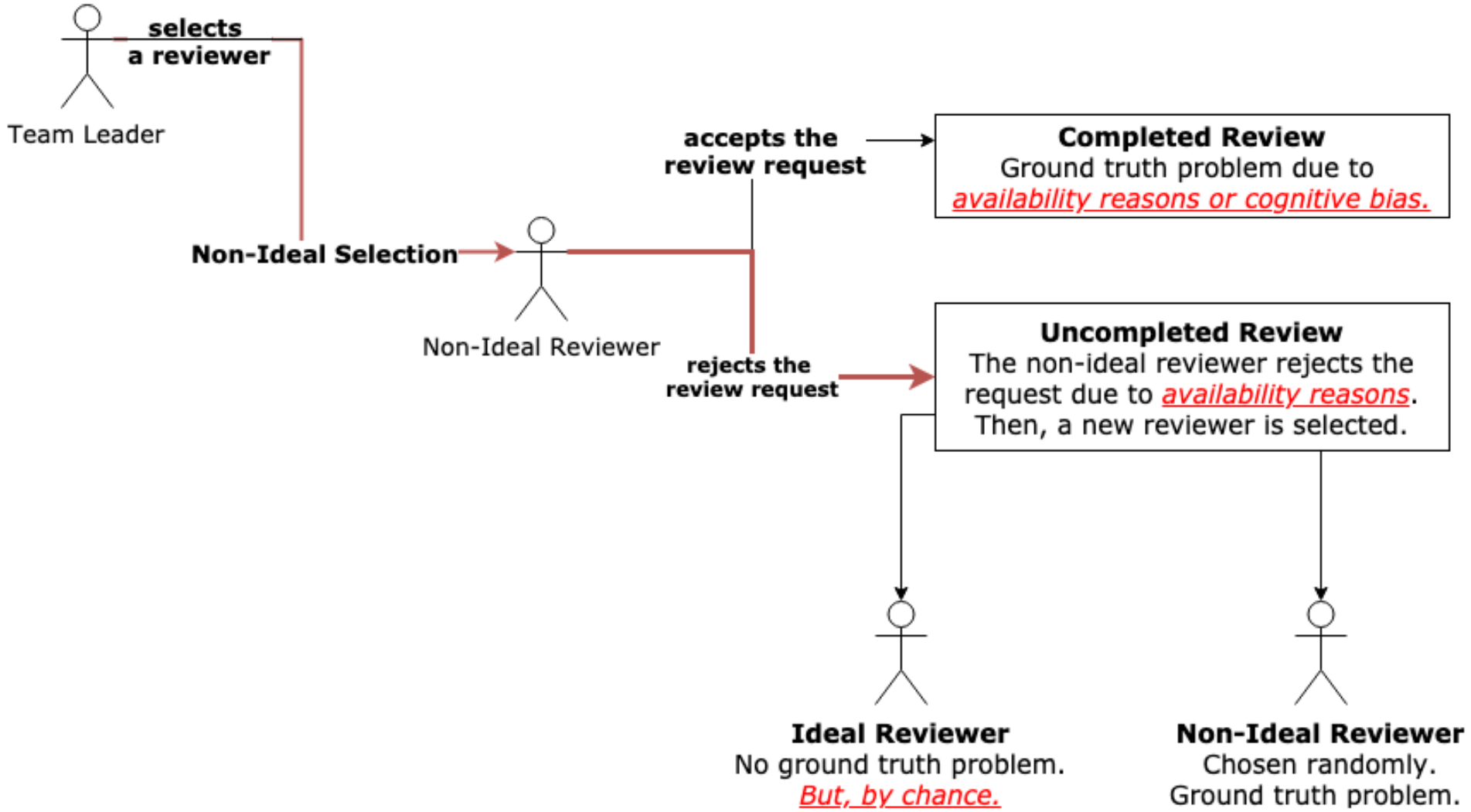
Scenario 2


accepts the
review request
Completed by Ideal Reviewer
No ground truth problem.
Ideal Selection
Ideal Reviewer
rejects the
review request
Completed by Non-Ideal Reviewer
Ground truth problem due to
availability reasons.
selects
a reviewer
Team Leader

Scenario 3


selects
a reviewer
Team Leader
Non-Ideal Selection
Non-Ideal Reviewer
accepts the
review request
Completed Review
Ground truth problem due to
availability reasons or cognitive bias.
rejects the
review request
Uncompleted Review
The non-ideal reviewer rejects the
request due to availability reasons.
Then, a new reviewer is selected.
Ideal Reviewer
No ground truth problem.
But, by chance.
Non-Ideal Reviewer
Chosen randomly.
Ground truth problem.

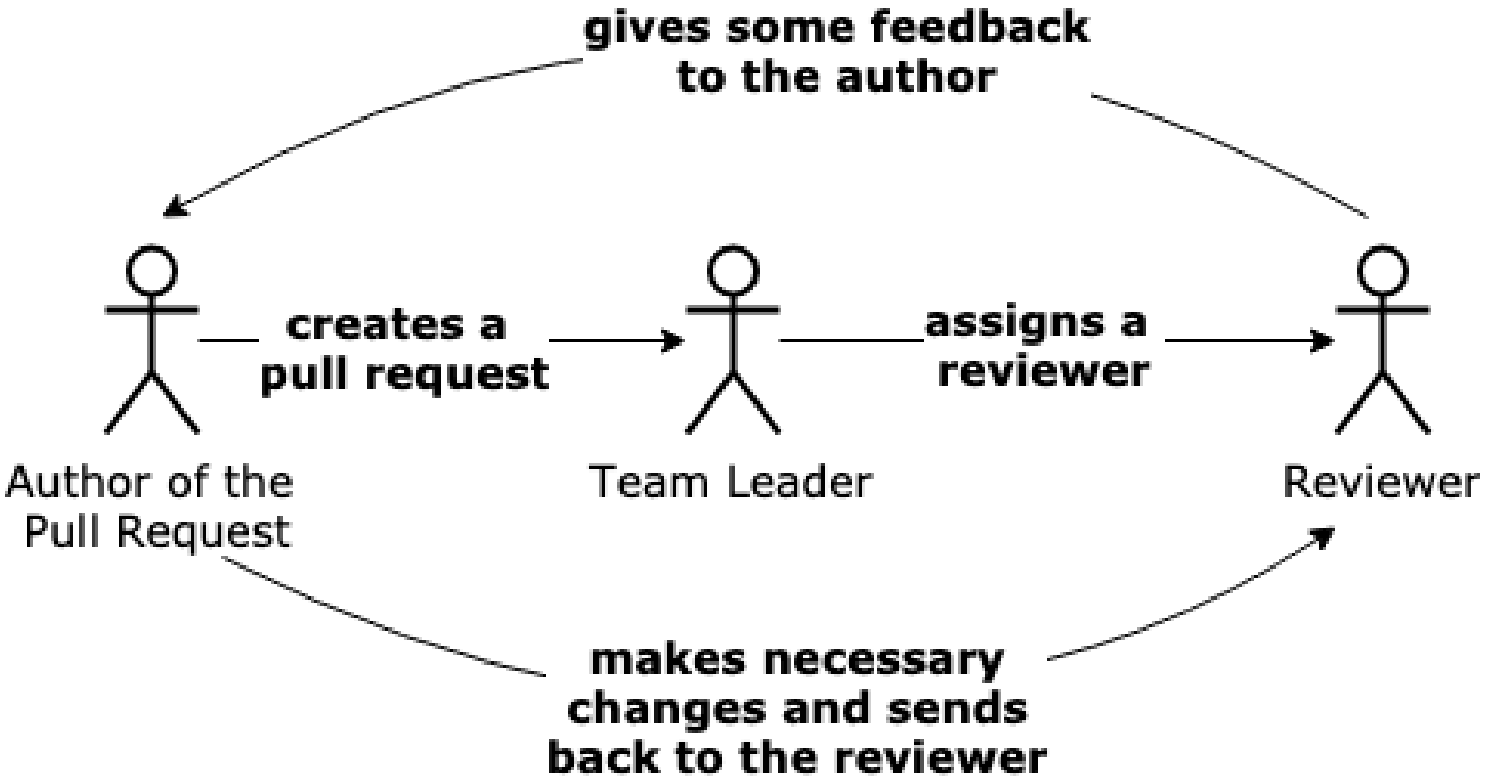
Scenario 4


selects
a reviewer
Team Leader
Non-Ideal Selection
Non-Ideal Reviewer
accepts the
review request
Completed Review
Ground truth problem due to
availability reasons or cognitive bias.
rejects the
review request
Uncompleted Review
The non-ideal reviewer rejects the
request due to availability reasons.
Then, a new reviewer is selected.
Ideal Reviewer
No ground truth problem.
But, by chance.
Non-Ideal Reviewer
Chosen randomly.
Ground truth problem.

Solution Alternatives

I - Setting up an Experiment in Real Life
II - Forward-Looking Mining

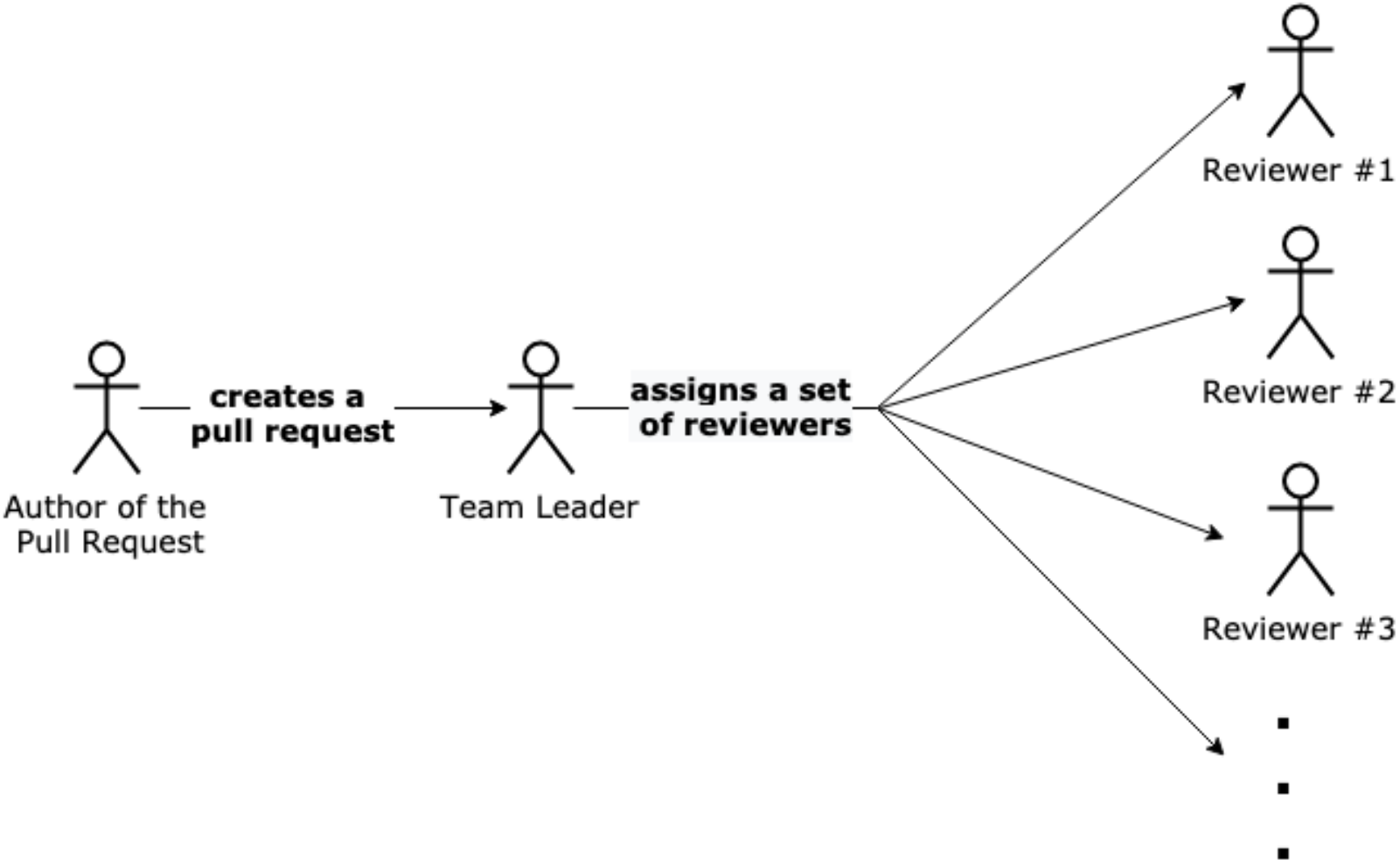
Solution I- Setting up an Experiment in Real Life



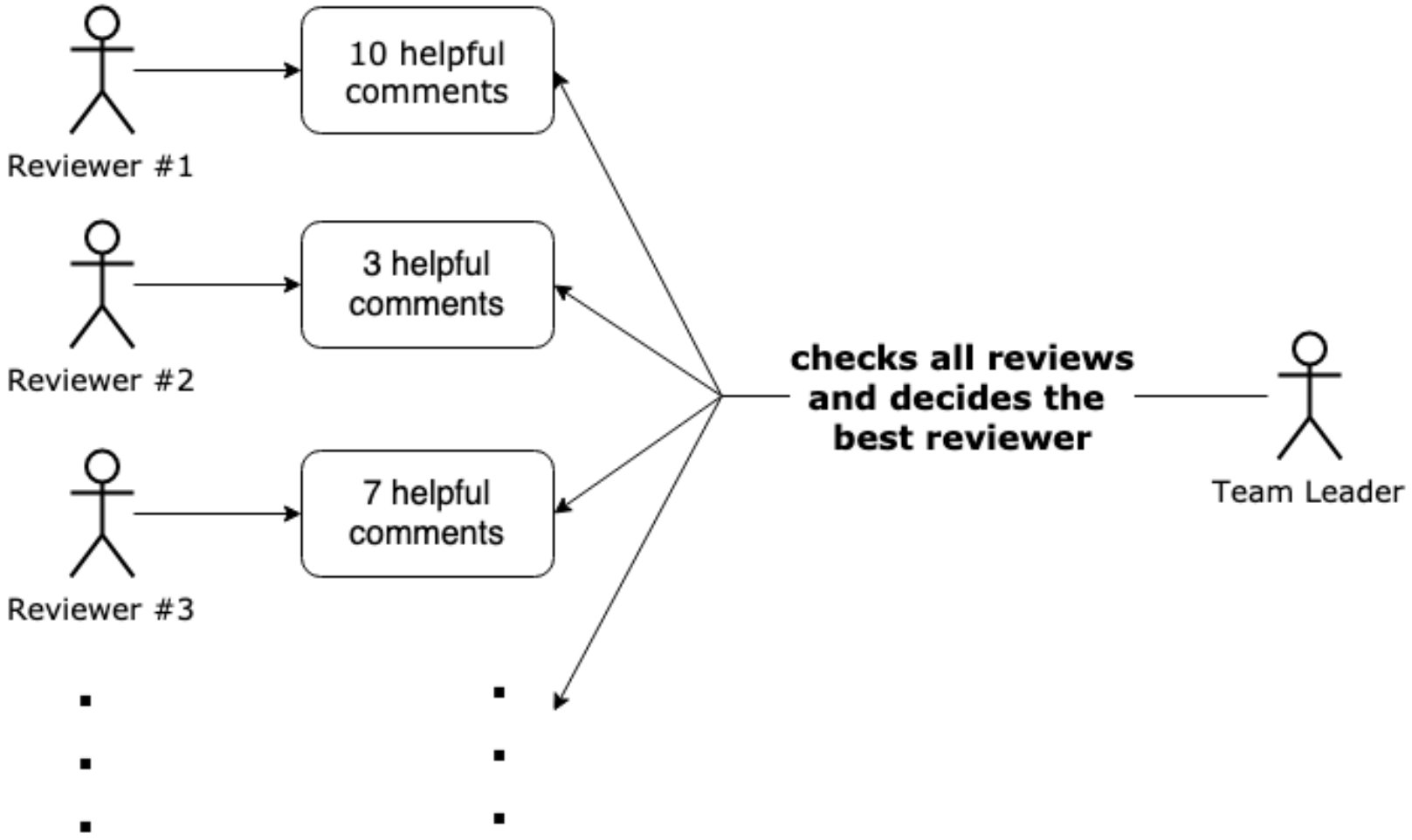
A Typical Reviewer Assignment Scenario

What if...

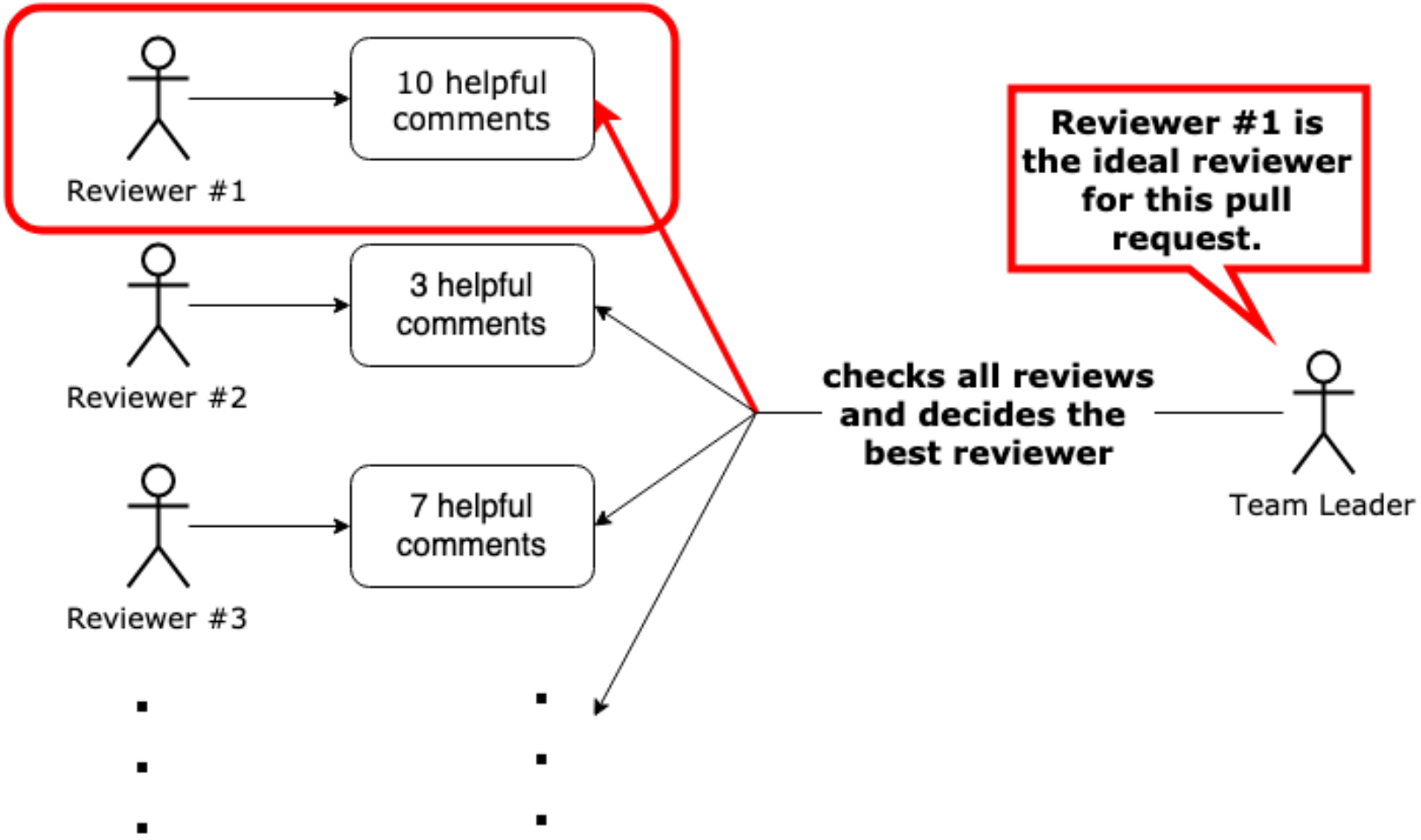
... we assign the same review to multiple reviewers simultaneously?



Then, choose the best one



Then, choose the best one



What is wrong with this method?

Expensive!

Making multiple developers spend time on a single review task is impractical and expensive.

Hard to evaluate!

It is not a straightforward task for the team leader to check all reviews and choose the best one.

What is wrong with this method?

Expensive!

Making multiple developers spend time on a single review task is impractical and expensive.

Hard to evaluate!

It is not a straightforward task for the team leader to check all reviews and choose the best one.

Solution II - Forward-Looking Mining

Idea:

Reopened bugs might indicate a bad code review.

How?

Consider the following scenario.

Solution II - Forward-Looking Mining

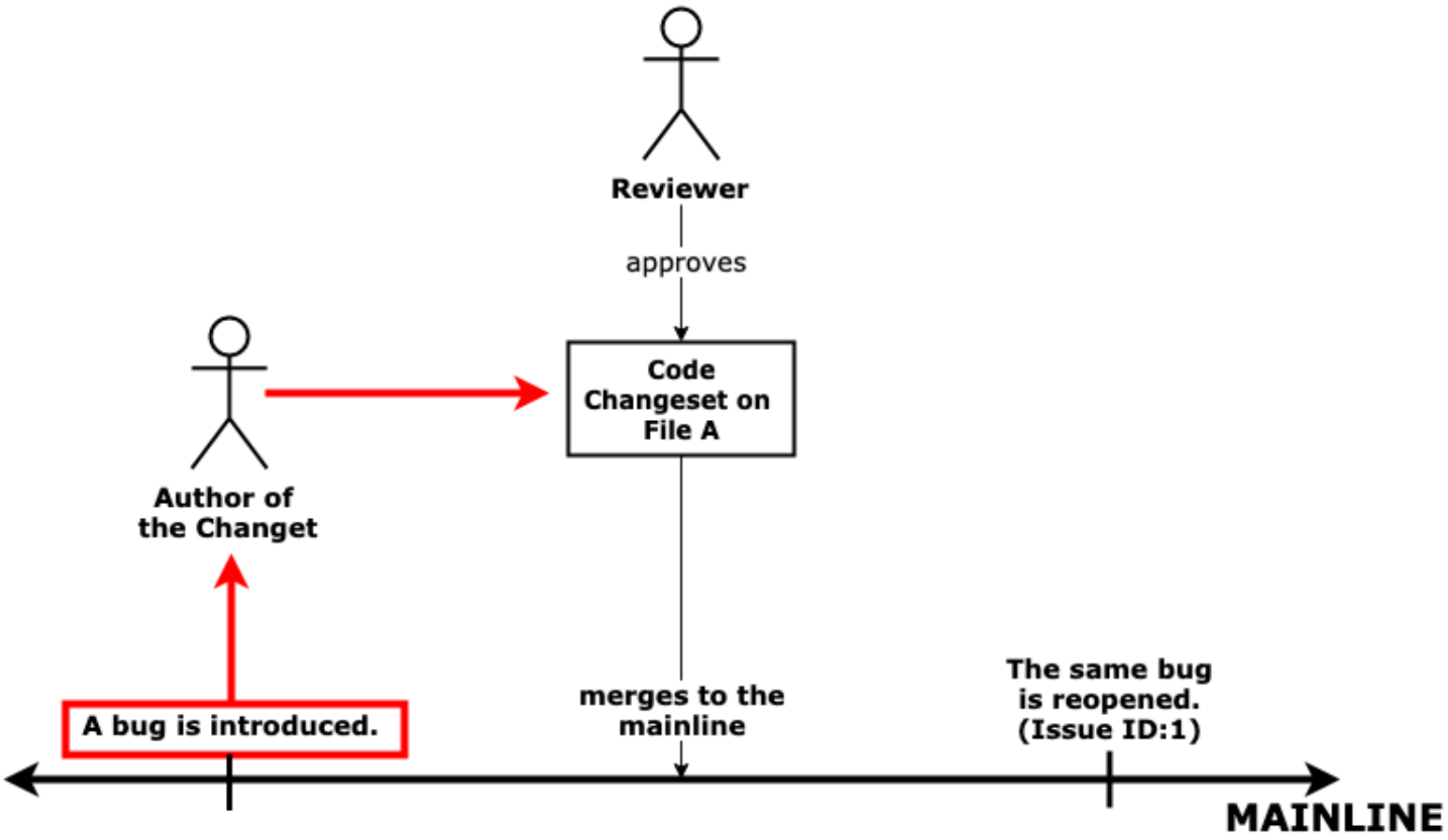
Idea:

Reopened bugs might indicate a bad code review.

How?

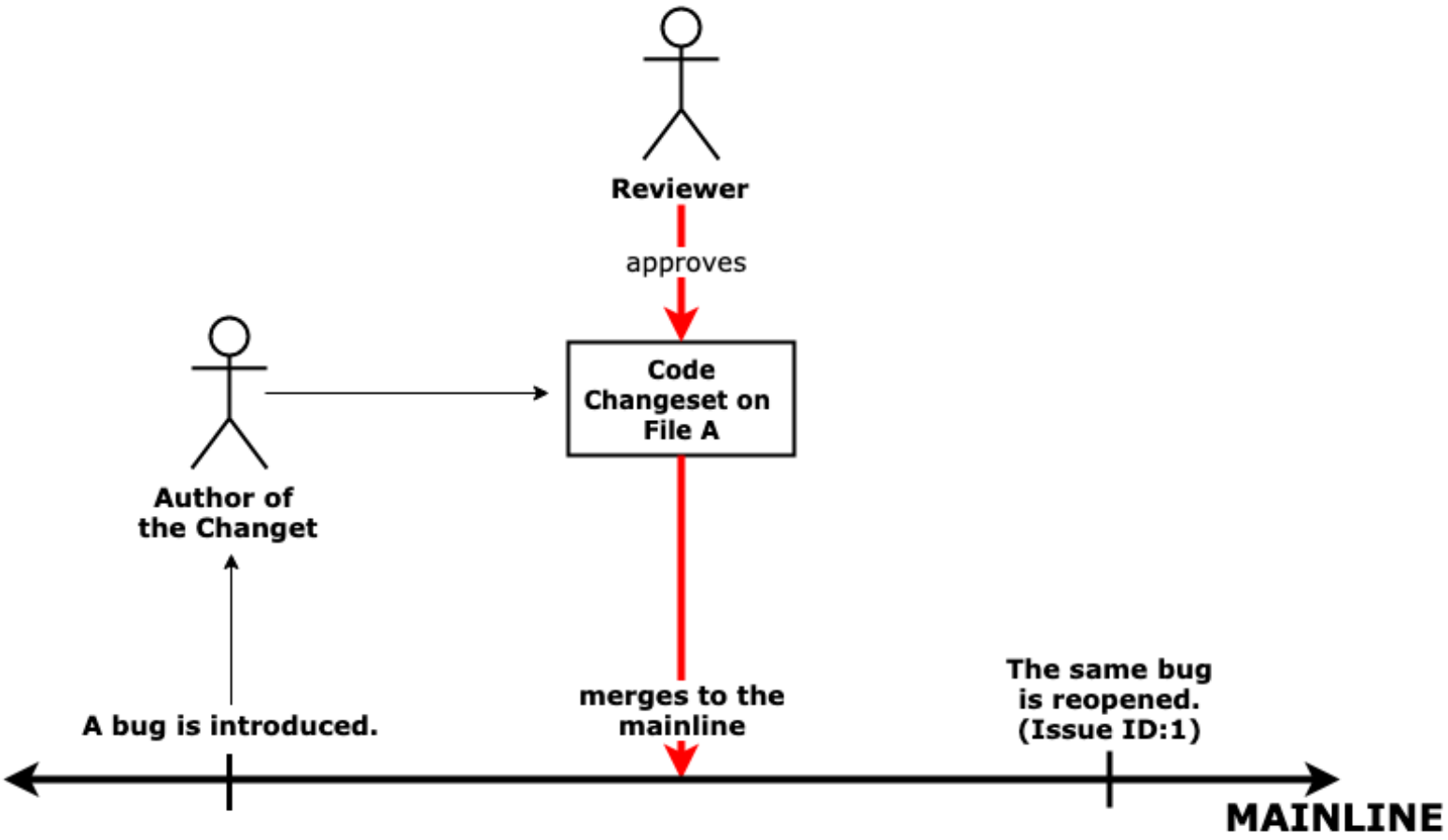
Consider the following scenario.

Consider the following scenario:



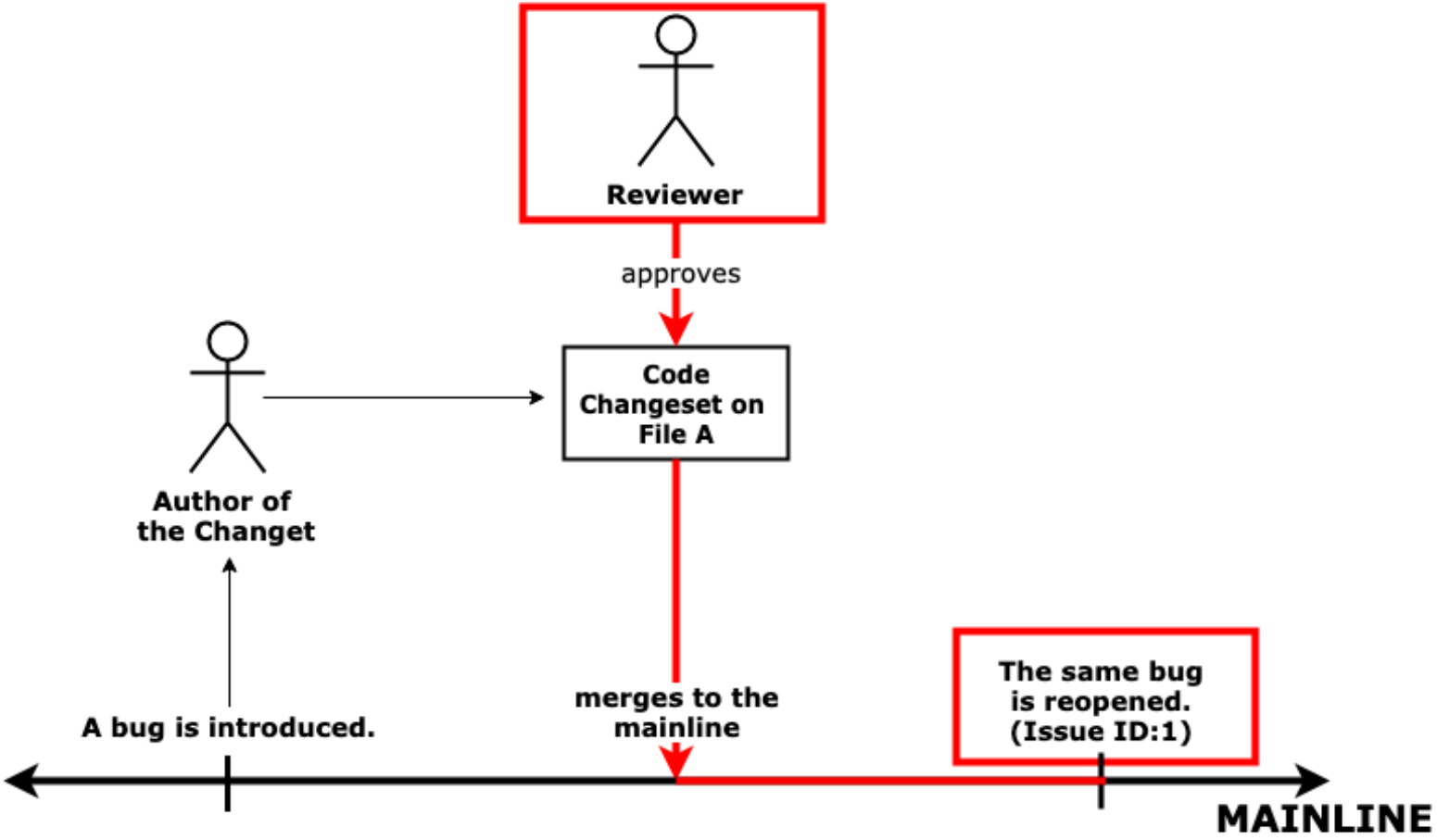
To fix a bug, a developer creates a pull request.

Consider the following scenario:



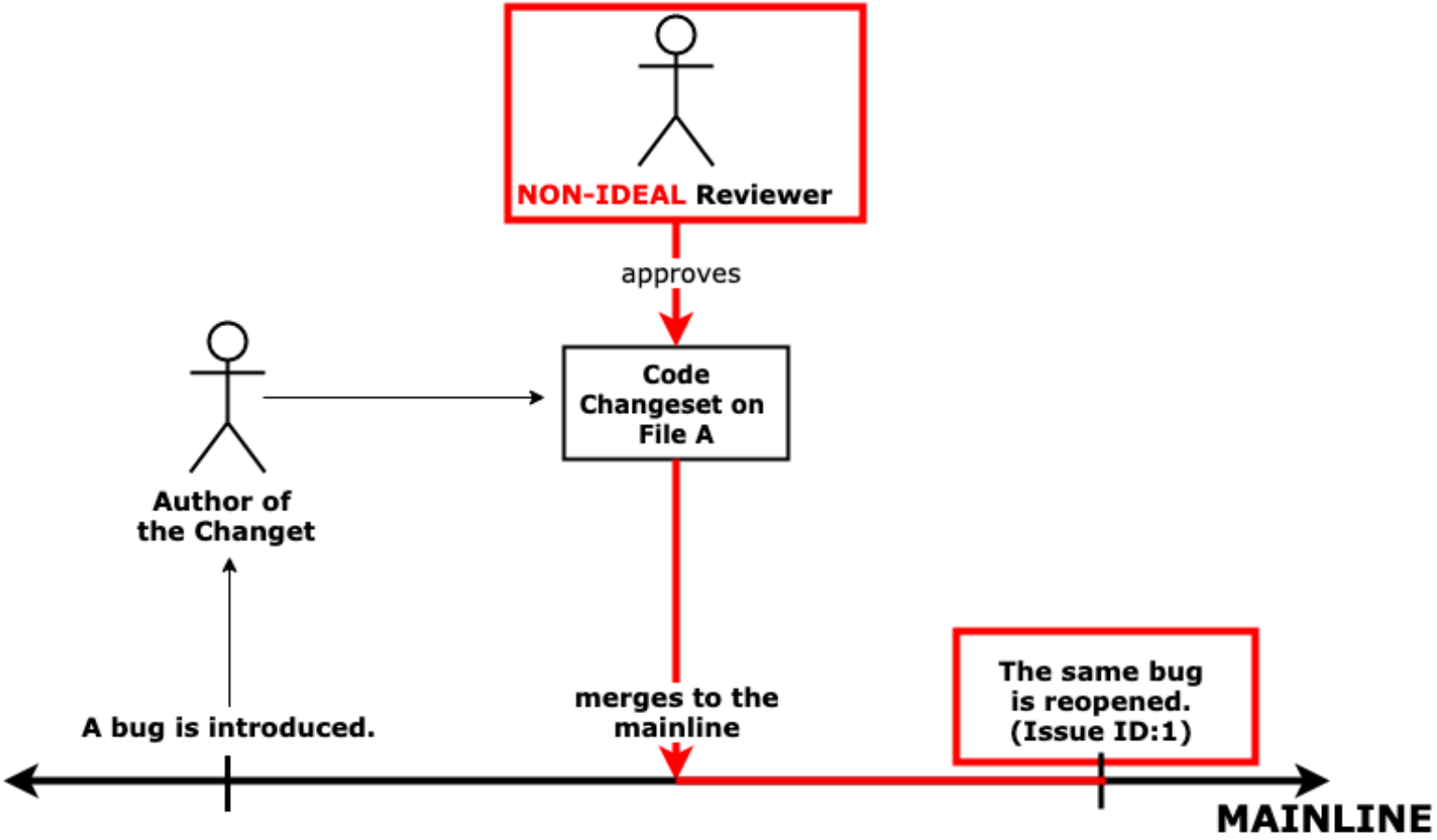
The assigned reviewer approves the pull request and the bug is closed.

Consider the following scenario:



If the same bug is reopened later, it is a potential indicator that the pull request is not conducted properly and the reviewer is not ideal.

Consider the following scenario:



Removing these instances from the dataset will increase the validity of ground truth.