



# Demystifying “Bad” Error Messages in Data Science Libraries

---

Yida Tao<sup>1</sup>, Zihui Chen<sup>1</sup>, Yepang Liu<sup>2</sup>, Jifeng Xuan<sup>3</sup>, Zhiwu Xu<sup>1</sup>, Shengchao Qin<sup>1,4</sup>

<sup>1</sup> Shenzhen University, China

<sup>2</sup> Southern University of Science and Technology, China

<sup>3</sup> Wuhan University, China

<sup>4</sup> Teesside University, UK

FSE. August 23–28, 2021.



# Error messages have a bad reputation ..... but why?



confusing  
uninformative  
misleading  
poor  
unhelpful  
obscure  
cryptic  
hard  
unclear



Error messages are (short) piece of natural-language text  
(e.g., “Table xxx not found.”)





# Research Questions



## RQ1

Why do certain error messages fail to be helpful debugging aids?



## RQ2

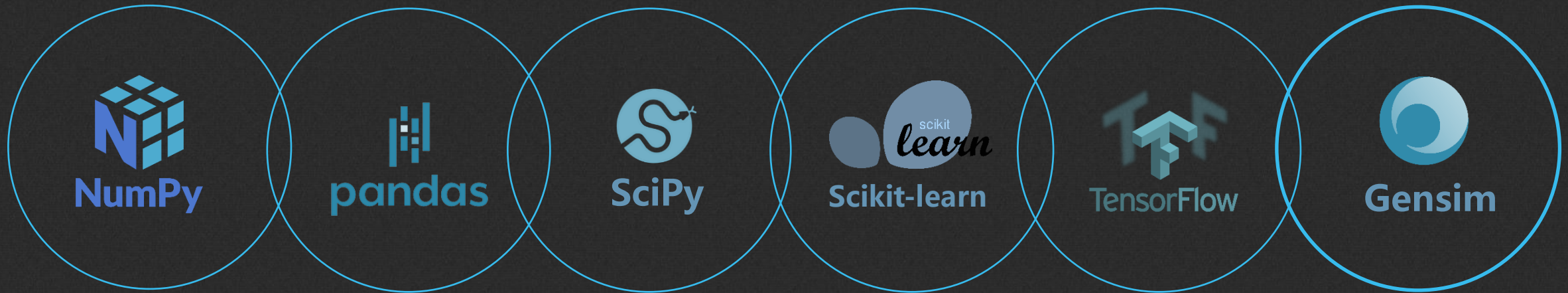
How do developers fix bad error messages in practice?



Guide practitioners to design and use error messages more effectively



# Subjects



- These 6 Python libraries are widely used in data science development
- Debugging data science program is crucial in modern software development
- Bad error messages could have significant negative impact
- Error message quality has never been studied in the data science domain







# 1 Methodology



# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?

## DEBUGGING CONTEXT

ERROR MESSAGE

?

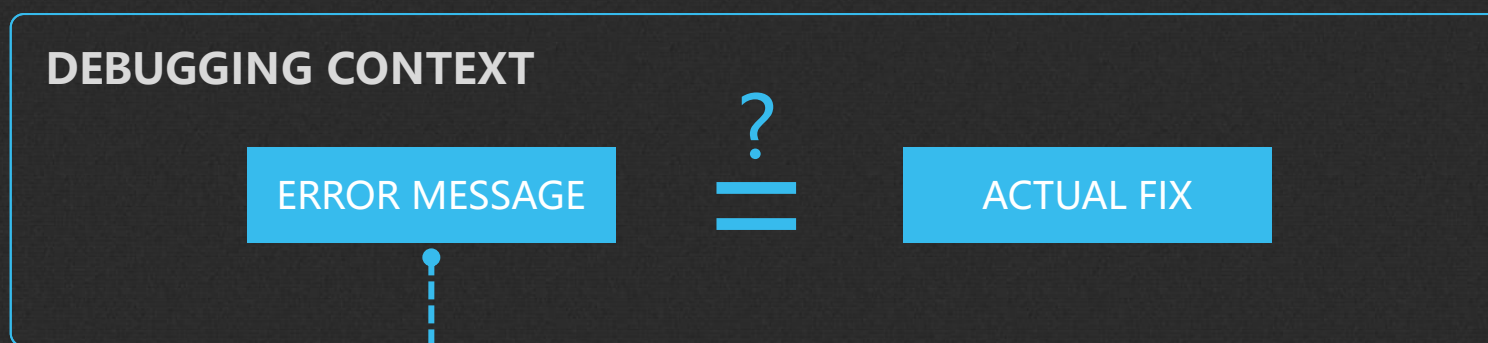
ACTUAL FIX





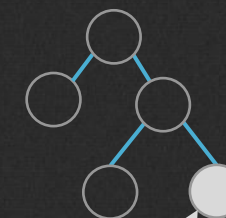
# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?



- 1 Extract all potential error messages by traversing the AST of subject libraries and look for `ast.Raise` nodes (4,744)

```
if not type_x in type_dict:  
    raise ValueError('Not supported set of arguments.')
```

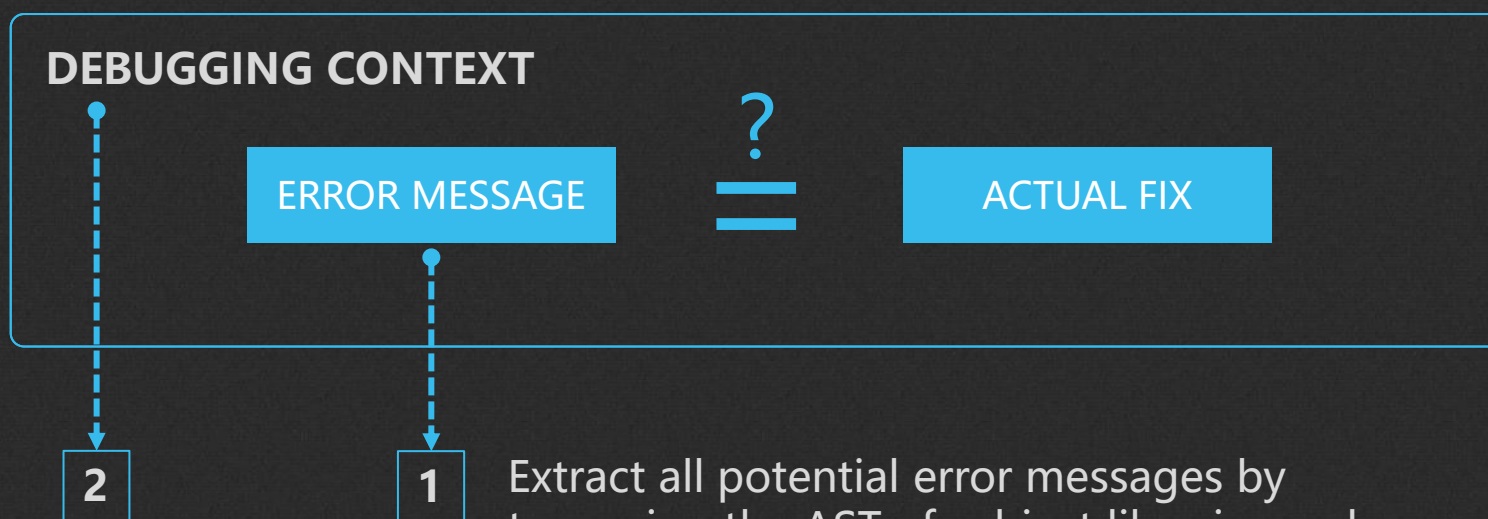


Node type: `ast.Raise`



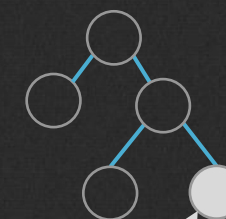
# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?



Search for Stack Overflow questions that contain the extracted error messages with the library tags, and with accepted answers (201)

```
if not type_x in type_dict:  
    raise ValueError('Not supported set of arguments.')
```



Node type: `ast.Raise`





# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?

**Stack Overflow Question**

ERROR MESSAGE

?

=

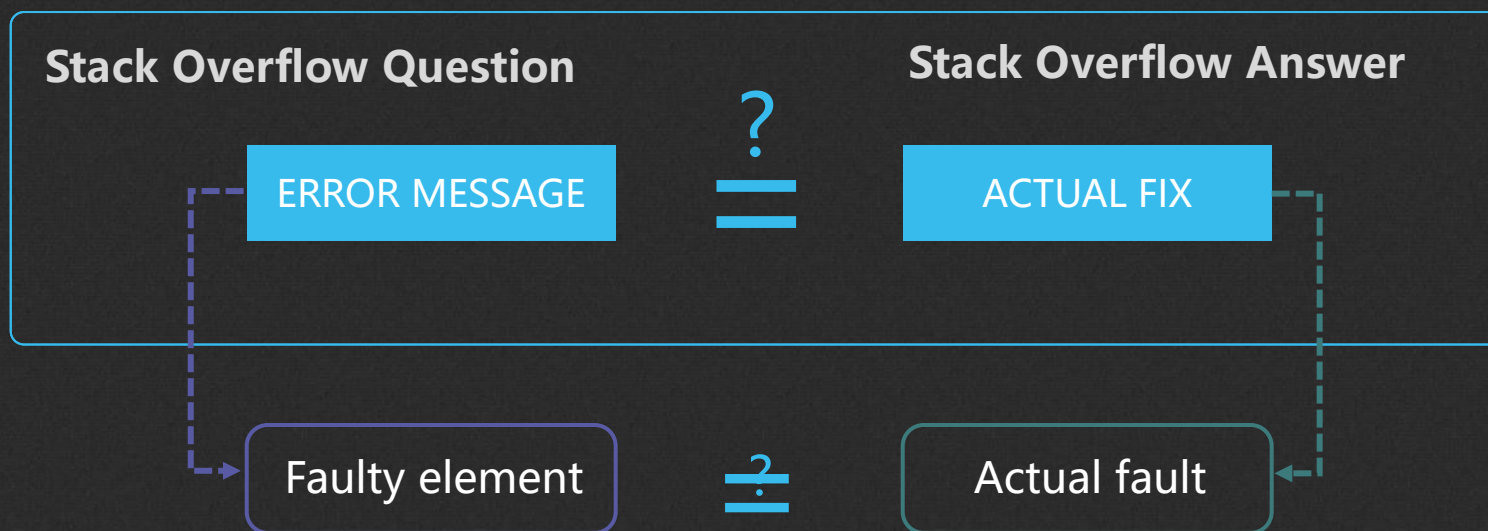
**Stack Overflow Answer**

ACTUAL FIX



# RQ1 Methodology

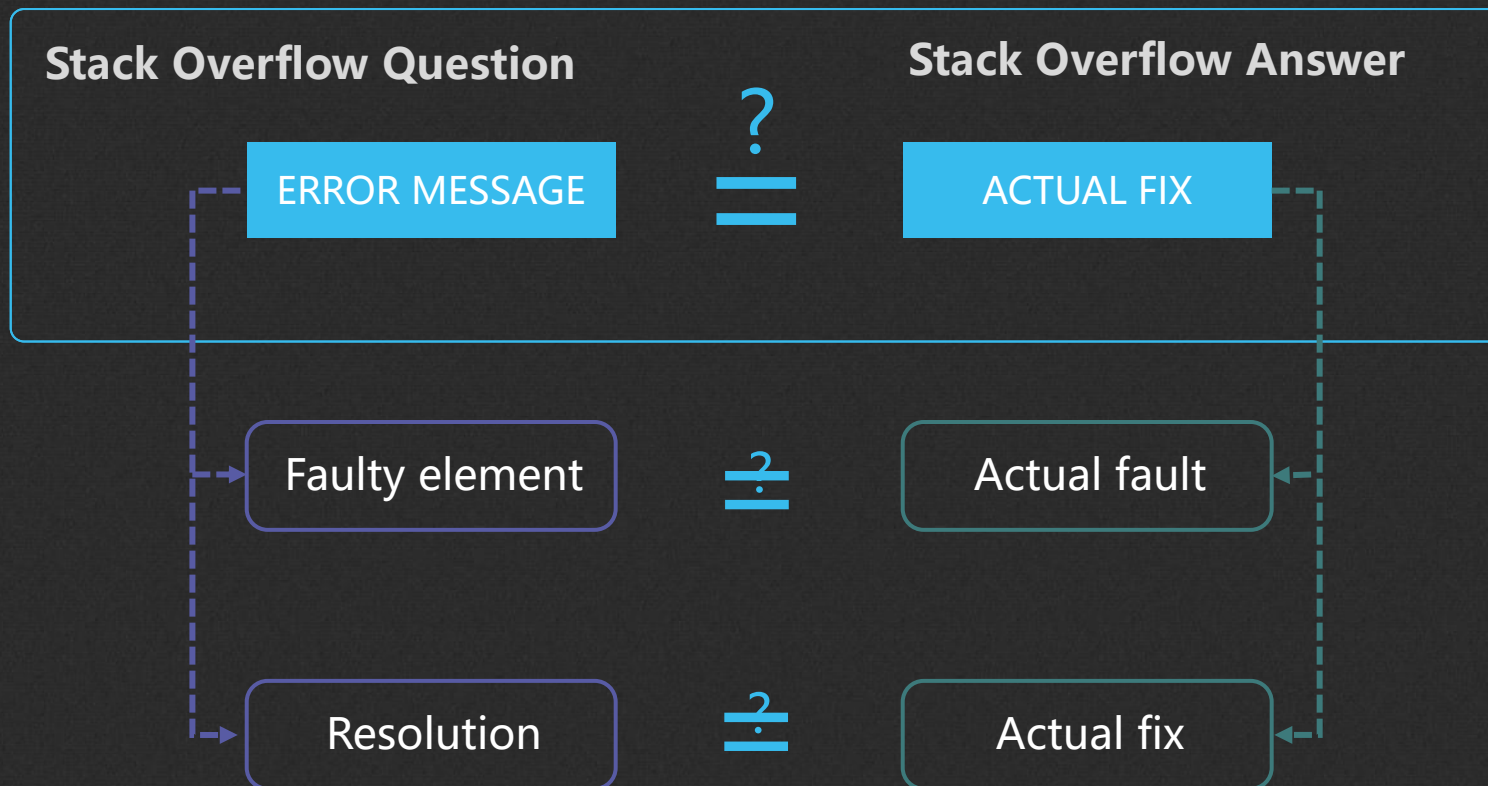
RQ1: Why do certain error messages fail to be helpful debugging aids?





# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?

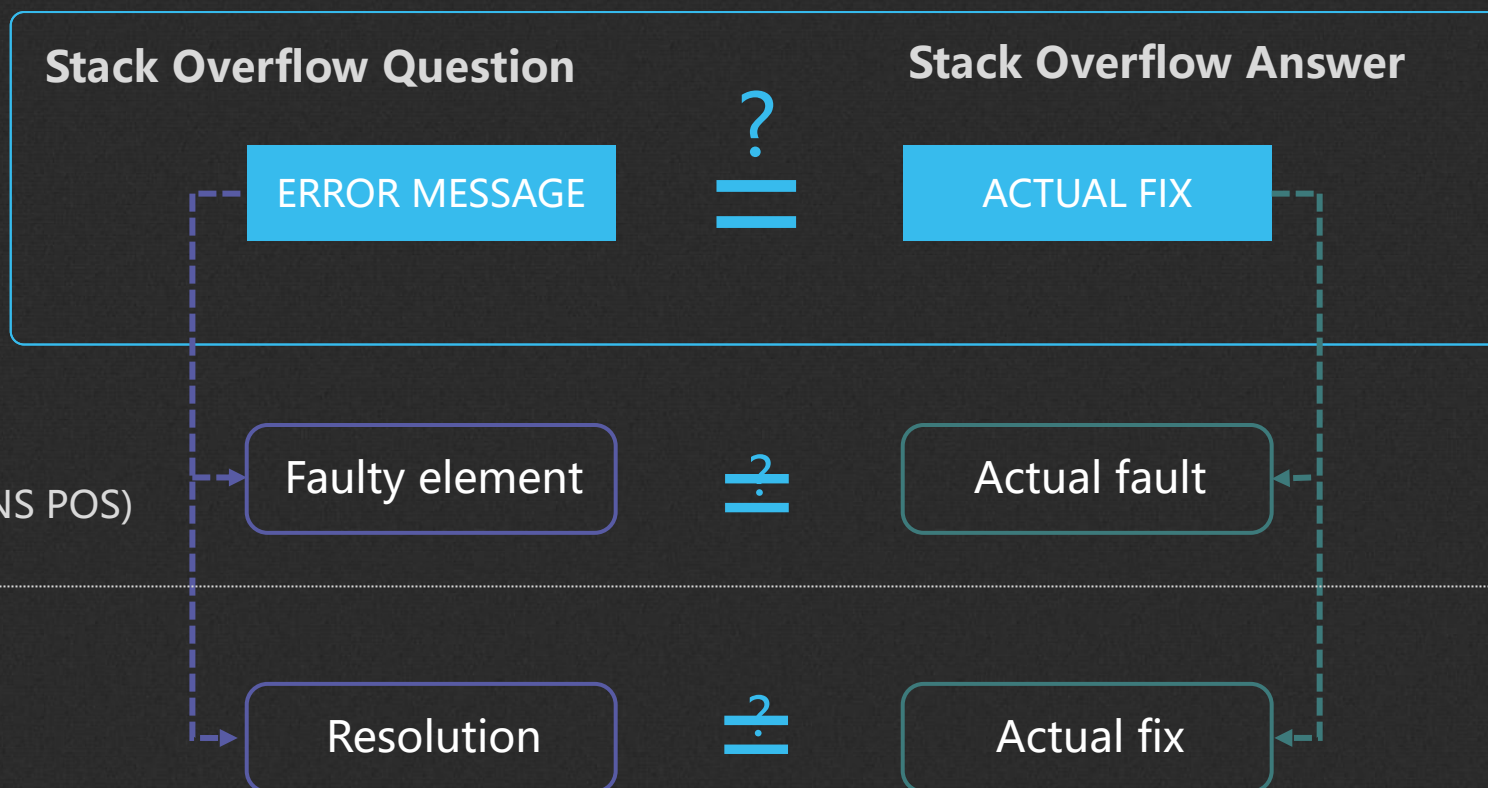


# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?

## Heuristics

**Faulty element:**  
Nouns (NN, NNP, NNS POS)



## Manual Classification

**Faulty element:** 2 labels

Same

Different





# RQ1 Methodology

RQ1: Why do certain error messages fail to be helpful debugging aids?

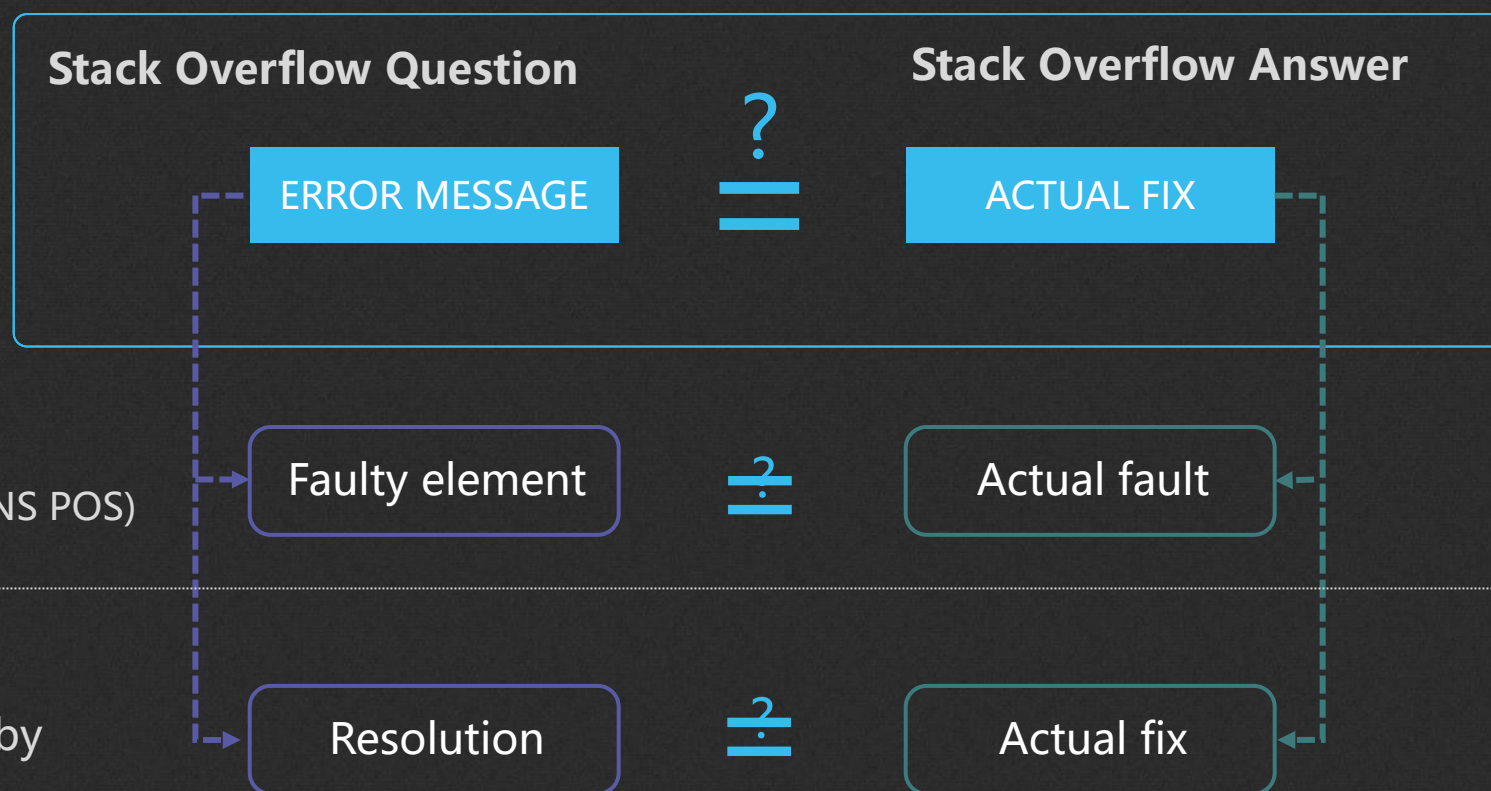
## Heuristics

### Faulty element:

Nouns (NN, NNP, NNS POS)

### Resolution:

Contents entailed by “must”, “should”, “expected”, “have to”, etc.



## Manual Classification

Faulty element: 2 labels

Same

Different

Resolution: 3 labels

Same

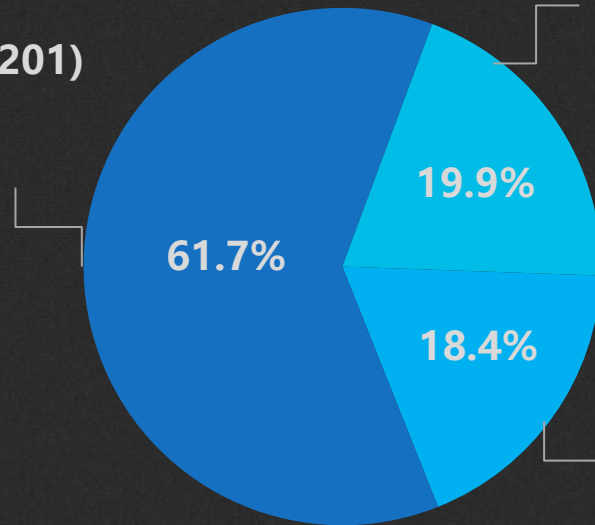
Different

Indirect



## RQ1 Error Message Characteristics

**UNINFORMATIVE** Error Messages (124/201)  
- Provide indirect resolution



**MISLEADING** Error Messages (40/201)  
- Resolution is different from the actual fix

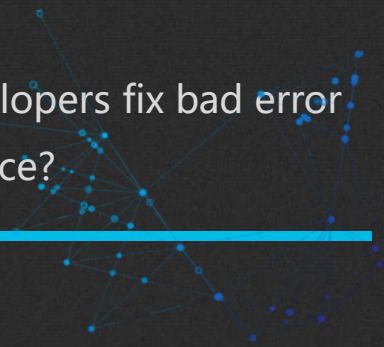
**SHOULD-BE-CLEAR** Error Messages (37/201)  
- Pinpoint the same faulty element and resolution as the actual fix





## RQ2 Methodology

RQ2: How do developers fix bad error messages in practice?



### Use GitHub issues to study fixing strategies

- ❑ Search for closed issues that contain the 201 error messages or the string “error message”
- ❑ Filter irrelevant or duplicate issues
- ❑ Manually study the issue threads with merged commits (280) and without merged commits (55)



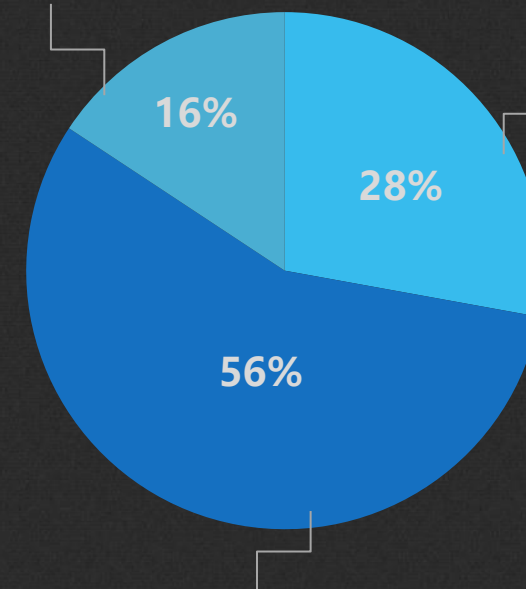
## RQ2 Methodology

RQ2: How do developers fix bad error messages in practice?

### Use GitHub issues to study fixing strategies

- ❑ Search for closed issues that contain the 201 error messages or the string “error message”
- ❑ Filter irrelevant or duplicate issues
- ❑ Manually study the issue threads with merged commits (280) and without merged commits (55)

Code + message change  
(44/280)



Message content change  
(78/280)

Source code change (158/280)



**Most error-message issues are fixed by updating source code logics, instead of by updating the phrasing of message content.**







# 2

## Qualitative Results

Deficiency causes & fixing strategies



# Misleading Error Messages



---

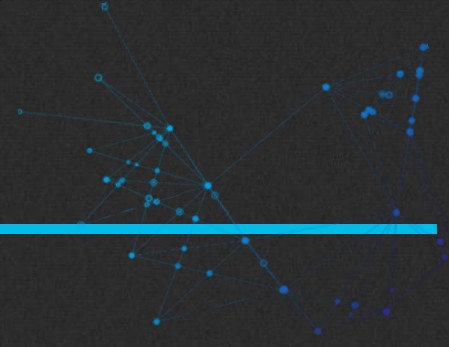
## CAUSE

- ❑ Library is blind to user intentions or errors (12%)
- ❑ Library is buggy or outdated (10%)
- ❑ Open design issues (2%)





# Misleading Error Messages



## CAUSE

- ❑ Library is blind to user intentions or errors (12%)
- ❑ Library is buggy or outdated (10%)
- ❑ Open design issues (2%)

## EXAMPLE

Code `numpy.loadtxt('data.txt', delimiter='\\t')`

Error msg Wrong number of columns at line 2.

C#	6.78
D	5.32
W	5.32

data.txt



# Misleading Error Messages



## CAUSE

- ❑ Library is blind to user intentions or errors (12%)
- ❑ Library is buggy or outdated (10%)
- ❑ Open design issues (2%)

## EXAMPLE

Code

```
numpy.loadtxt('data.txt',delimiter='\t')
```

Actual  
fix

```
numpy.loadtxt('data.txt',delimiter='\t', comments=None)
```

Set comments=None to disable default '#' and imply no comments

Error  
msg

Wrong number of columns at line 2.

C# 6.78

D 5.32

W 5.32

data.txt





# Misleading Error Messages

## CAUSE

- ❑ Library is blind to user intentions or errors (12%)
- ❑ Library is buggy or outdated (10%)
- ❑ Open design issues (2%)

## EXAMPLE

**Code** `numpy.loadtxt('data.txt', delimiter='\\t')`

**Actual fix** `numpy.loadtxt('data.txt', delimiter='\\t', comments=None)`  
Set comments=None to disable default '#' and imply no comments

**Error msg** Wrong number of columns at line 2.



```
C# 6.78
D  5.32
W  5.32
```

data.txt

Ambiguous user intention:  
Is '#' a comment or a normal token?



# Misleading Error Messages

---

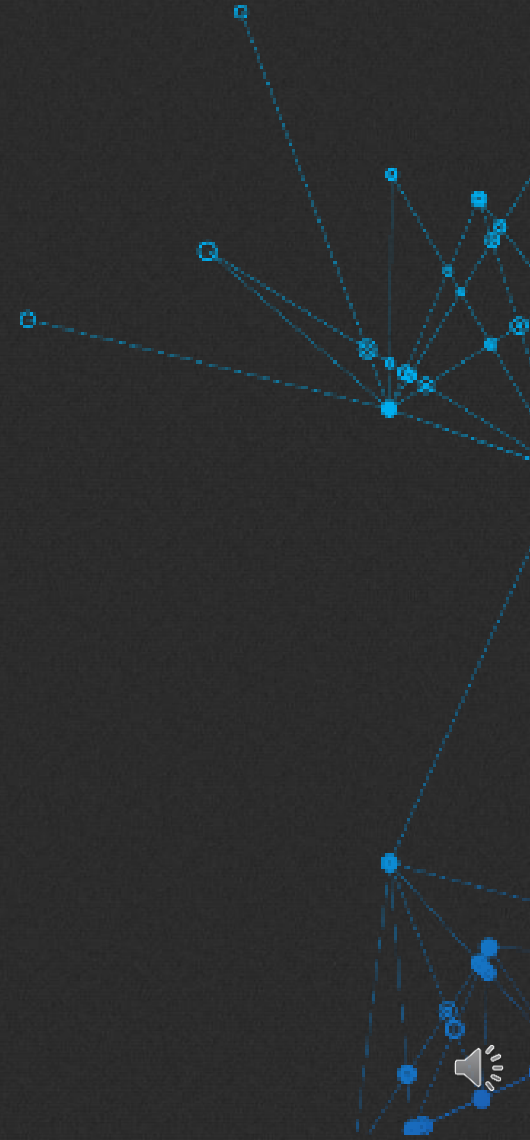
## CAUSE

- ❑ Library is blind to user intentions or errors (12%)
- ❑ Library is buggy or outdated (10%)
- ❑ Open design issues (2%)



## FIXING STRATEGIES

Can't or Won't fix





# Misleading Error Messages

## CAUSE

- ❑ Library is blind to user intentions or errors (12%)
- ❑ Library is buggy or outdated (10%)
- ❑ Open design issues (2%)



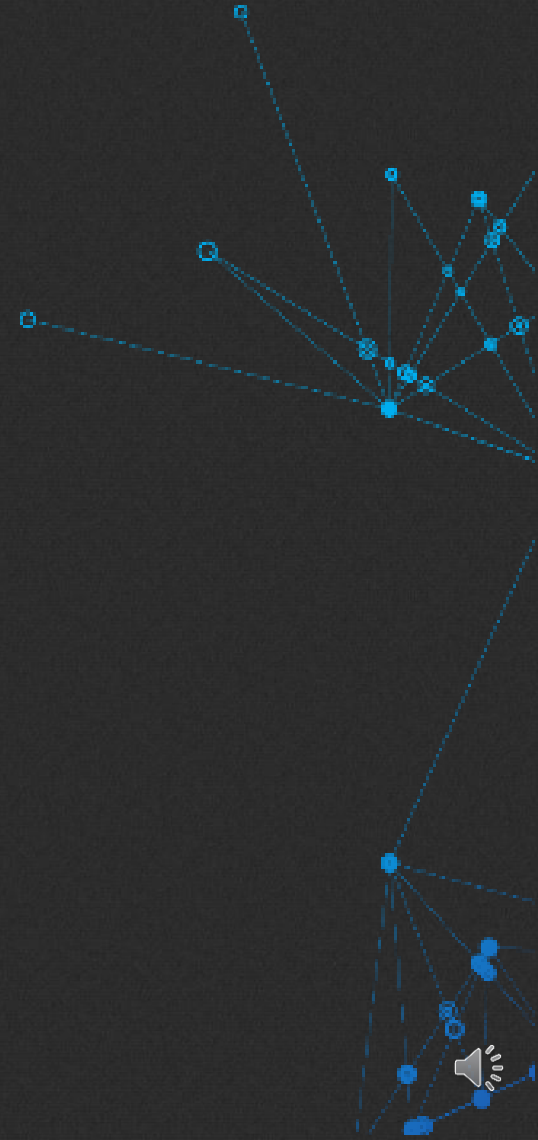
## FIXING STRATEGIES

Can't or Won't fix



Source code change

Logic fixes and design changes (23.9%)



# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)





# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

## EXAMPLE

Error  
msg

Shapes (118, 1) and (118, 3) are incompatible

Actual  
fix

Set label\_dimension=3 when invoking the DNNRegressor API

This is the end of the error message I get:

```
File "/usr/lib/python2.7/site-packages/tensorflow/contrib/learn/python/learn/estimators/c
return head.head_ops(features, labels, mode, _train_op_fn, logits)
File "/usr/lib/python2.7/site-packages/tensorflow/contrib/learn/python/learn/estimators/t
head_name=self.head_name)
File "/usr/lib/python2.7/site-packages/tensorflow/contrib/learn/python/learn/estimators/t
loss_fn(logits, labels),
File "/usr/lib/python2.7/site-packages/tensorflow/contrib/learn/python/learn/estimators/t
logits.get_shape().assert_is_compatible_with(labels.get_shape())
File "/usr/lib/python2.7/site-packages/tensorflow/python/framework/tensor_shape.py", line
raise ValueError("Shapes %s and %s are incompatible" % (self, other))
ValueError: Shapes (118, 1) and (118, 3) are incompatible
```



# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

## EXAMPLE

Error  
msg

Shapes (118, 1) and (118, 3) are incompatible

Actual  
fix

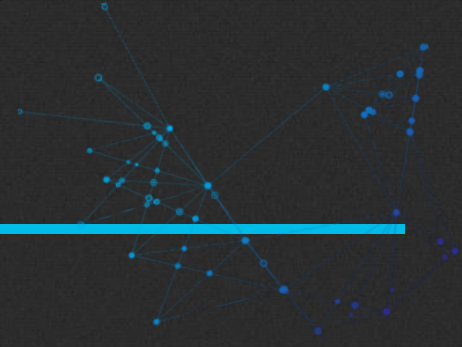
Set `label_dimension=3` when invoking the `DNNRegressor` API

Error messages pinpoint to the final manifestation of the fault instead of the faulty element itself. Although it points to the right direction, it might be considered as less straightforward





# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

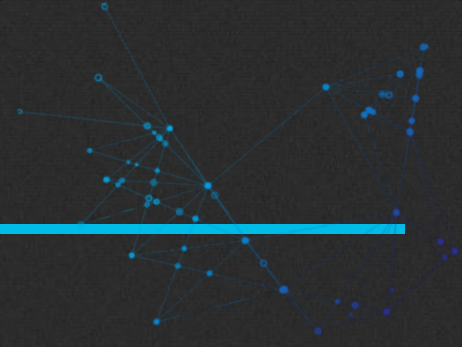
## FIXING STRATEGIES

### 1. Fail early, fail fast (31%)

```
def __init__(self, objs, axis=0, join='outer', join_axes=None, keys=None, levels=None,
              names=None, ignore_index=False, verify_integrity=False):
+   if not isinstance(objs, (tuple, list, dict)):
+       raise AssertionError('first argument must be a list of pandas objects, you passed'
+                             'an object of type "{}".format(type(objs).__name__)')
```



# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

## FIXING STRATEGIES

### 1. Fail early, fail fast (31%)

```
def __init__(self, objs, axis=0, join='outer', join_axes=None, keys=None, levels=None,
              names=None, ignore_index=False, verify_integrity=False):
+   if not isinstance(objs, (tuple, list, dict)):
+       raise AssertionError('first argument must be a list of pandas objects, you passed'
+                             'an object of type "{}".format(type(objs).__name__))
```

### 2. Ask for forgiveness than permission (7%)

```
+   try:
+       s = Series(A.data, MultiIndex.from_arrays((A.row, A.col)))
+   except AttributeError:
+       raise TypeError('Expected coo_matrix. Got {} instead.
+                       .format(type(A).__name__))
```





# Uninformative Error Messages

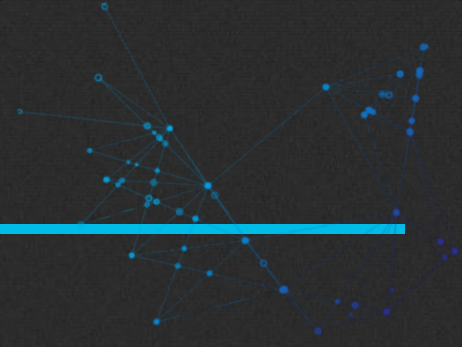


## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)



# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

## EXAMPLE

Code

```
pandas.to_numeric(df)
```

Error  
msg

Arg must be a list, tuple, 1-d array, or Series.



Actual  
fix

```
df.apply(lambda x: pandas.to_numeric(x), axis=0))  
(among other viable resolutions)
```





# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

## FIXING STRATEGIES

### 1. Add actionable instructions (message content fix)

“X needs to contain only non-negative integers. Please set categories='auto' explicitly to be able to use arbitrary integer values as category identifiers.”



# Uninformative Error Messages



## CAUSE

- ❑ Faulty element is too low-level (15%)
- ❑ Faulty element is too general (2%)
- ❑ Requirements are cognitively distant from the fix implementation (56%)
- ❑ Requirements are implicit (2%)

## FIXING STRATEGIES

### 1. Add actionable instructions (message content fix)

“X needs to contain only non-negative integers. Please set categories='auto' explicitly to be able to use arbitrary integer values as category identifiers.”

### 2. Can't or Won't fix

“ I'm not really fond of being prescriptive with solution as library is blind to the user's intentions.”





# Should-be-clear Error Messages



These error messages pinpoint the same faulty element and resolution as the actual fix ..... but they still appear in Stack Overflow questions. Why?

User are unfamiliar with:

- Domain-specific concepts
- Library/API-specific concepts
- Programming languages

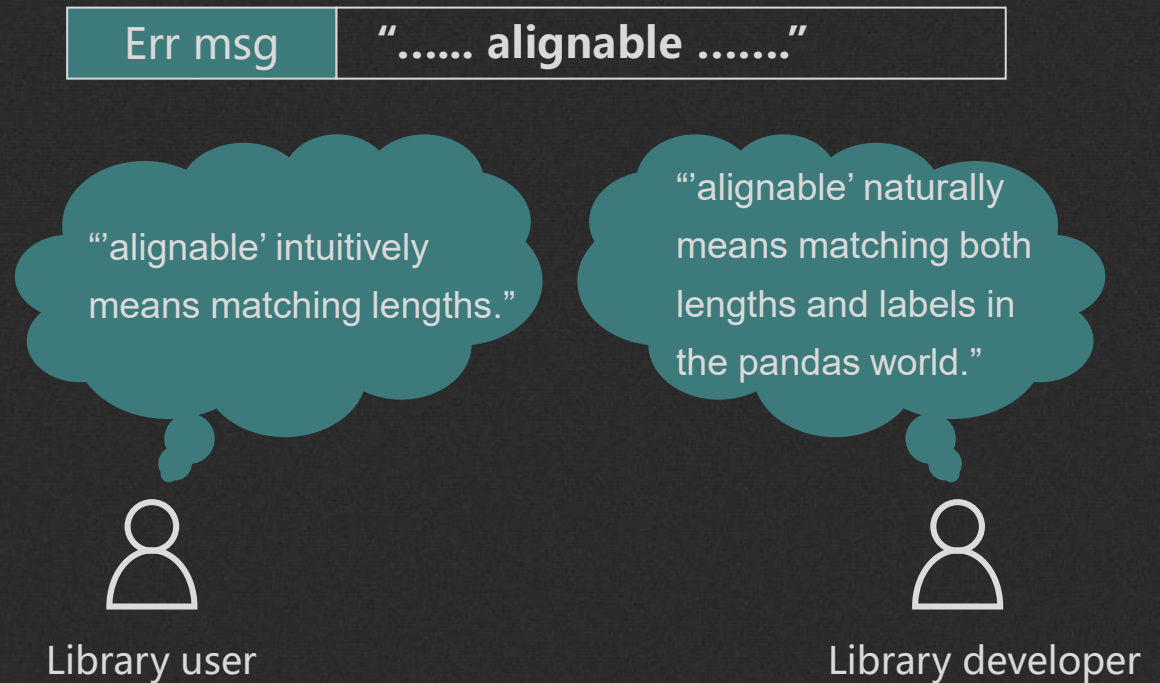


# Should-be-clear Error Messages

These error messages pinpoint the same faulty element and resolution as the actual fix ..... but they still appear in Stack Overflow questions. Why?

User are unfamiliar with:

- Domain-specific concepts
- Library/API-specific concepts
- Programming languages







3

# Summary & Insights



# Summary & Insights

---

## **Bad error message ↗ Fixable by libraries**

The “misleadingness” and “uninformativeness” of error messages are often inevitable and hard to resolve, since libraries are inherently blind to user intentions or mistakes

## **Bad error message ≠ Bad phrasing of message content**

Source code and program-logic changes are often required to fix error-message-related complaints. Exclusive message content updates are less frequent.

## **Bad error message (users' perspective) ≠ Bad error message (devs' perspective)**

Library users and developers may have different interpretations and perceptions of the same error message.





# Summary & Insights

## **Bad error message ↗ Fixable by libraries**

The “misleadingness” and “uninformativeness” of error messages are often inevitable and hard to resolve, since libraries are inherently blind to user intentions or mistakes

## **Bad error message ≠ Bad phrasing of message content**

Source code and program-logic changes are often required to fix error-message-related complaints. Exclusive message content updates are less frequent.

## **Bad error message (users' perspective) ≠ Bad error message (devs' perspective)**

Library users and developers may have different interpretations and perceptions of the same error message.

## **Library users**

Raise awareness of the inherent limitations of libraries  
Avoid overreliance on error messages in debugging

## **Library developers**

Raise awareness of the conceptual mismatch in users  
Improving error handling practices

## **Researchers**

Differentiate bad error messages more carefully based on specific deficiency causes  
Focus on how error handling practice and logic errors affect the perception of error messages





# Thank You.

---

