

Does AI Replace Software Developers?

Michael Pradel (University of Stuttgart)

Based on joint work with Aryaz Eghbali, Islem Bouzenia,
Luca Di Grazia, and Wai Chow

Will AI Replace Software Engineers?



Mohammed Hassan · [Follow](#)

3 min read · Feb 25

Will AI Replace Software Engineers?



Mohammed Hassan

3 min read · Feb 2



The Future Of Programming: Will AI Replace Programmers?



Strivemindz

Your Digital Transformation Partner

Published Jun 5, 2023

+ Follow

Will AI Replace Software Engineers?

ChatGPT may be coming for our jobs. Here are the 10 roles that AI is most likely to replace.

Aaron Mok and Jacob Zinkula Updated Sep 4, 2023, 4:24 PM GMT+2



g: ...

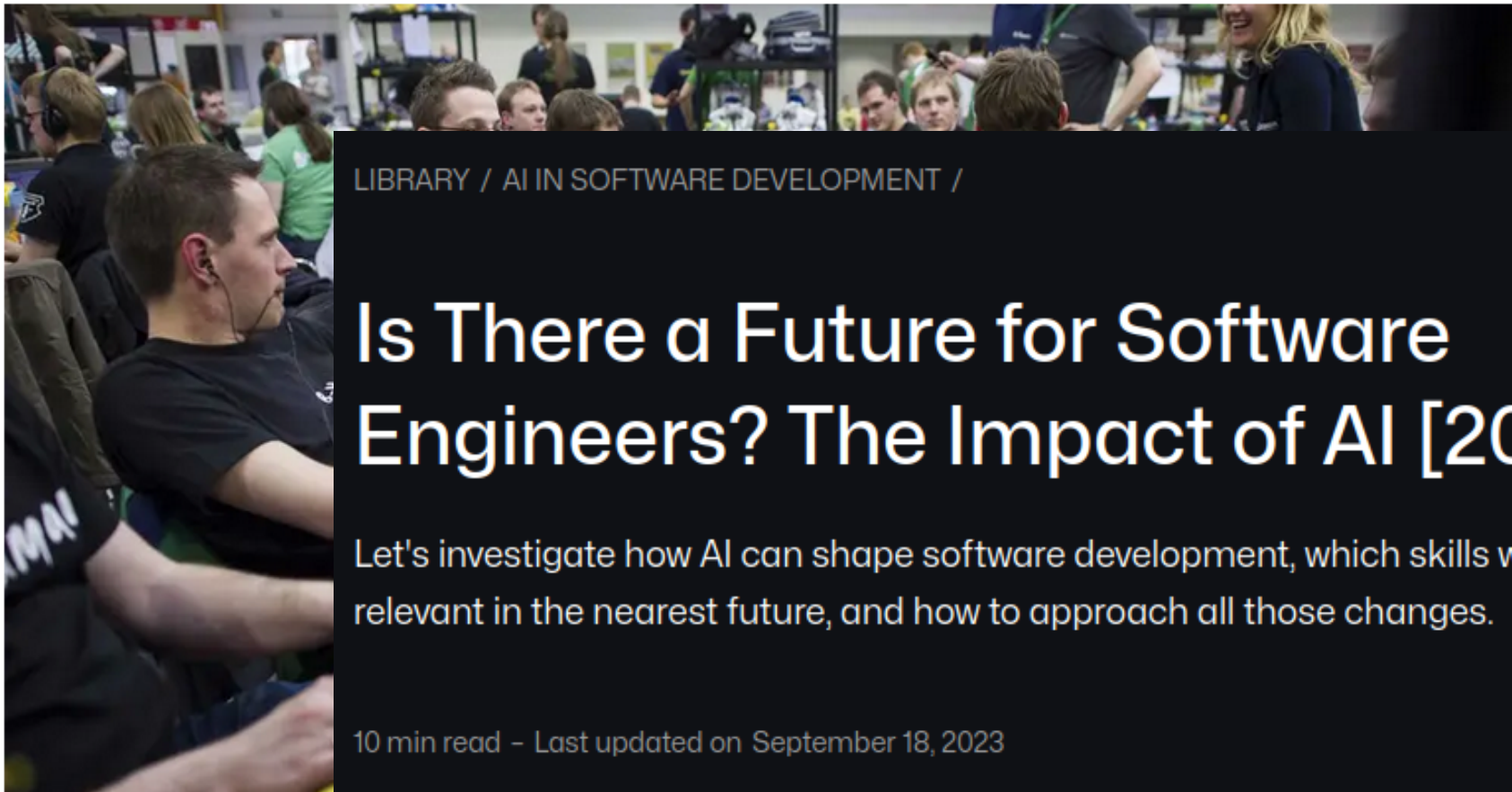
+ Follow

Insider compiled a list of the 10 jobs that could be disrupted by AI tools like ChatGPT, according to experts. Jens Schlueter/Getty Images

Will AI Replace Software Engineers?

ChatGPT may be coming for our jobs. Here are the 10 roles that AI is most likely to replace.

Aaron Mok and Jacob Zinkula Updated Sep 4, 2023, 4:24 PM GMT+2



LIBRARY / AI IN SOFTWARE DEVELOPMENT /

Is There a Future for Software Engineers? The Impact of AI [2023]

Let's investigate how AI can shape software development, which skills will be relevant in the nearest future, and how to approach all those changes.

10 min read – Last updated on September 18, 2023

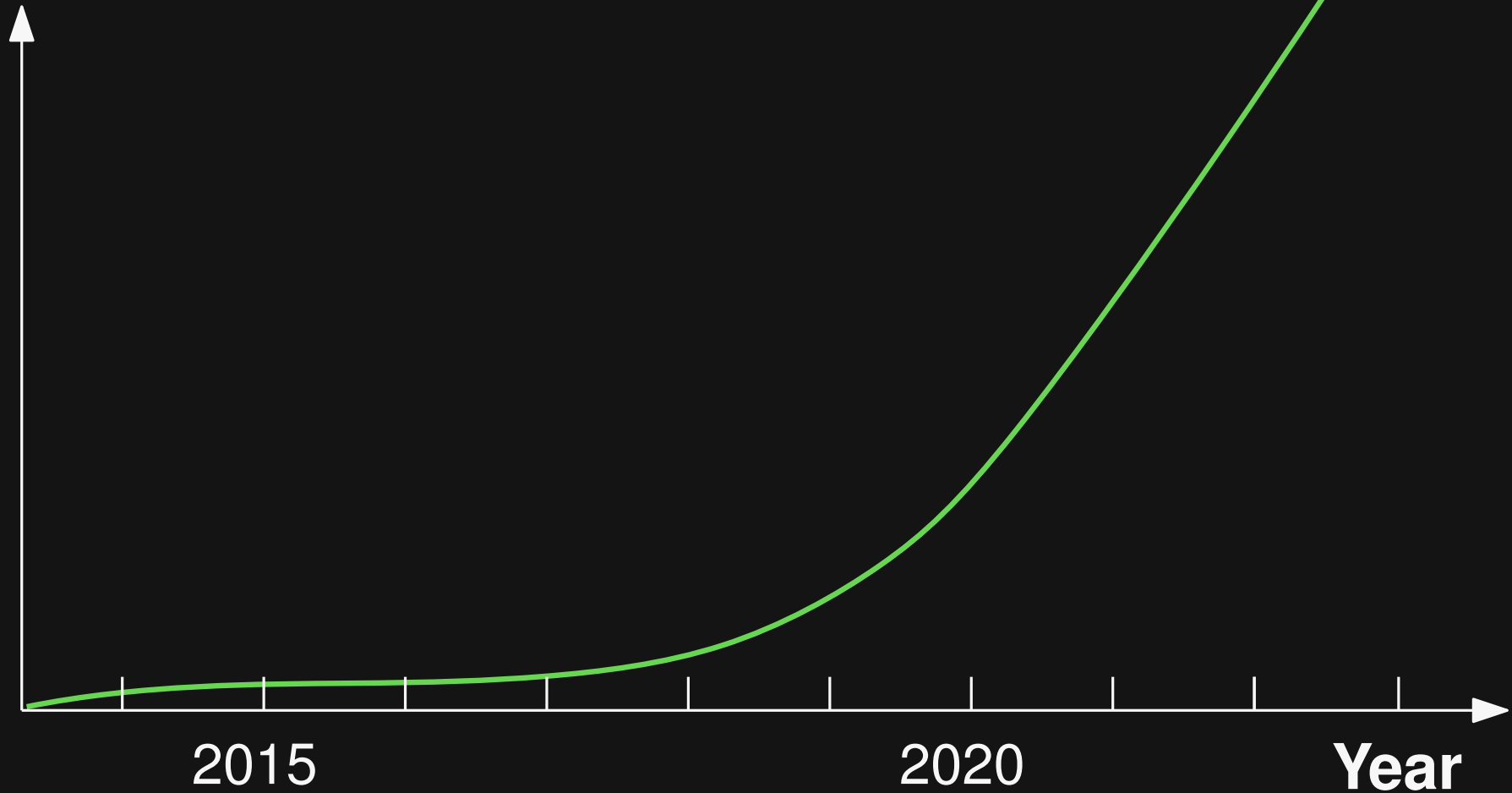
Insider compiled a list of the 10 jobs that could be disrupted by AI tools like ChatGPT, according to experts. Jens Schlueter/Getty Images

This Talk

- 1) Overview of state-of-the-art
- 2) My answer to the question
- 3) Peek into the future

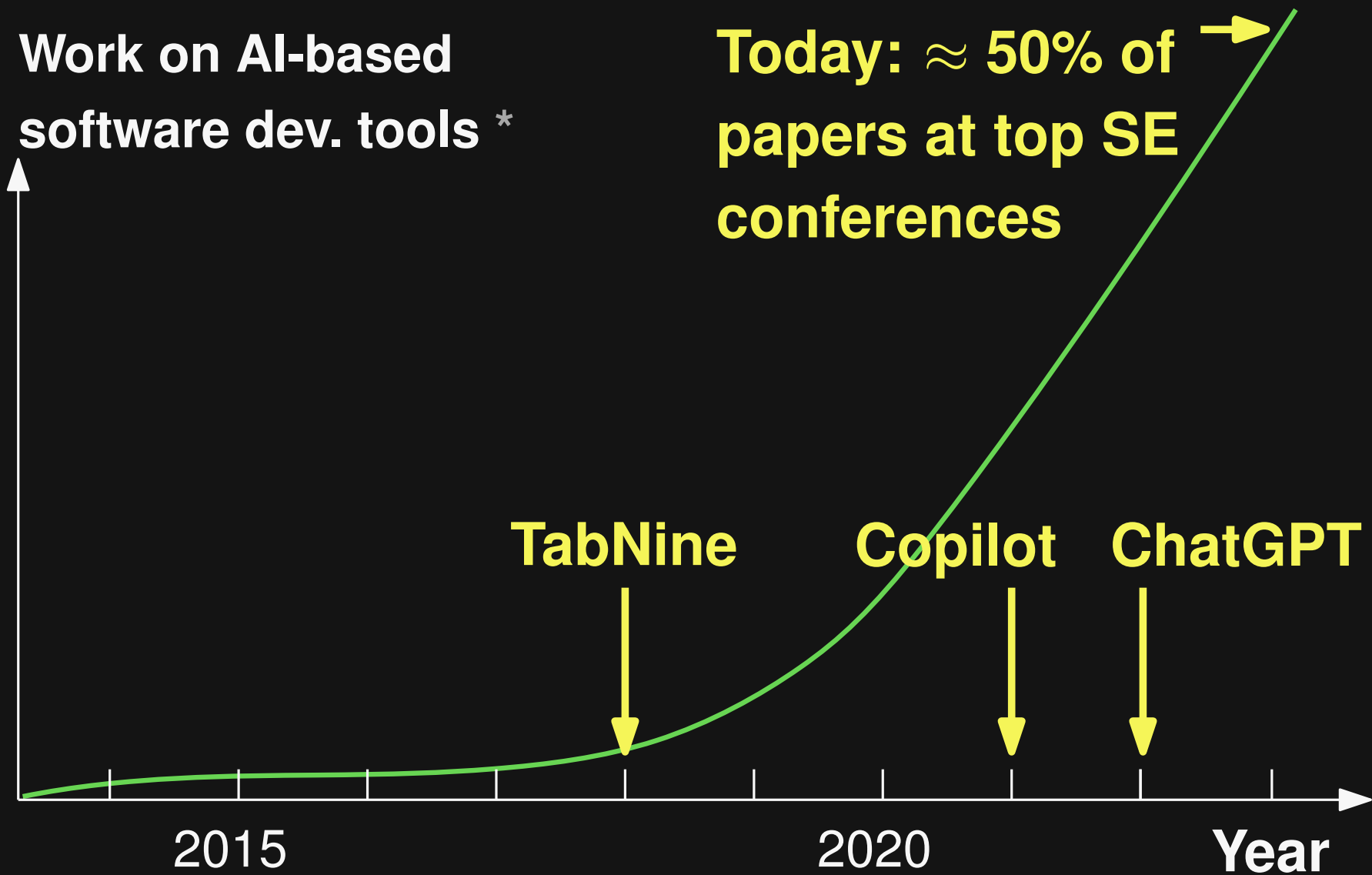
Timeline

Work on AI-based
software dev. tools *



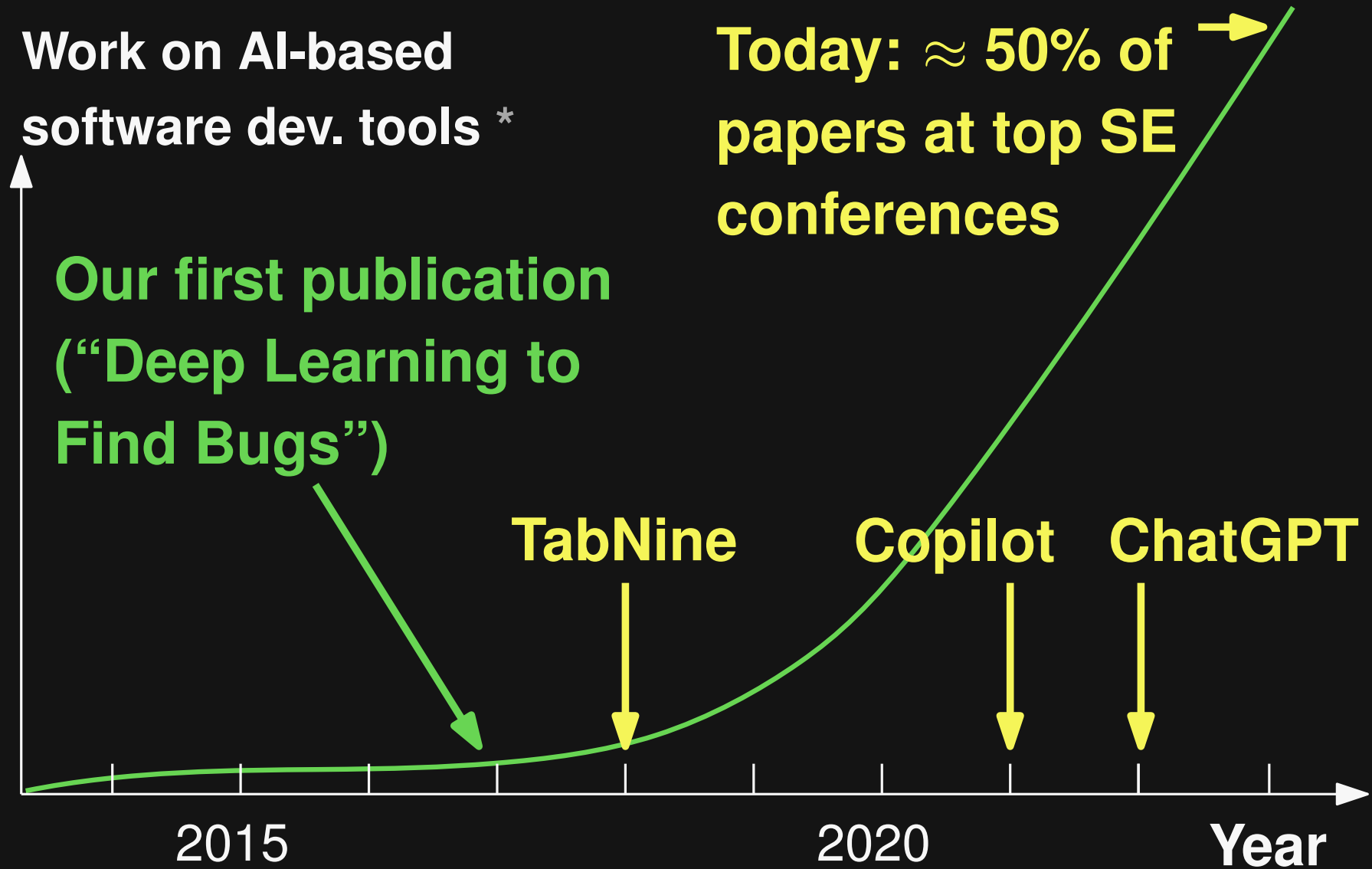
* Estimate based on *Neural Software Analysis*, Pradel & Chandra, CACM'22

Timeline



* Estimate based on *Neural Software Analysis*, Pradel & Chandra, CACM'22

Timeline



* Estimate based on *Neural Software Analysis*, Pradel & Chandra, CACM'22

Three Examples

Software development activities
(traditionally done by hand):

- Write code
- Search for bugs
- Fix bugs

Three Examples

Software development activities
(traditionally done by hand):

- Write code → ■ Code completion
- Search for bugs → ■ Neural bug detection
- Fix bugs → ■ Automated program repair

Code Completion

- [Copilot demo]

Problem Solved?

Useful, but many unsolved questions

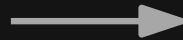
- Challenge 1: **Project-specific APIs**
- Challenge 2: **Prioritizing context**

Problem Solved?

Useful, but many unsolved questions

- Challenge 1: **Project-specific APIs**
- Challenge 2: **Prioritizing context**

Code used
for training



AI

Problem Solved?

Useful, but many unsolved questions

- Challenge 1: **Project-specific APIs**
- Challenge 2: **Prioritizing context**



Problem Solved?

Useful, but many unsolved questions

- Challenge 1: **Project-specific APIs**
- Challenge 2: **Prioritizing context**

Prompt size
of models:



Size of real-world projects:



An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
```

An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    return sorted(docs, key=lambda doc: doc.relevance, reverse=True)[:top_k]
```



Prediction by CodeGen model

An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    docs_scores = [(doc, compute_relevance_score(doc, keyword) ) for doc in docs]
    sorted_docs_scores = sorted(docs_scores, key=lambda x: x[1], reverse=True)
    return [doc_score[0] for doc_score in sorted_docs_scores[:top_k]]
```



Prediction by ChatGPT's model

An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

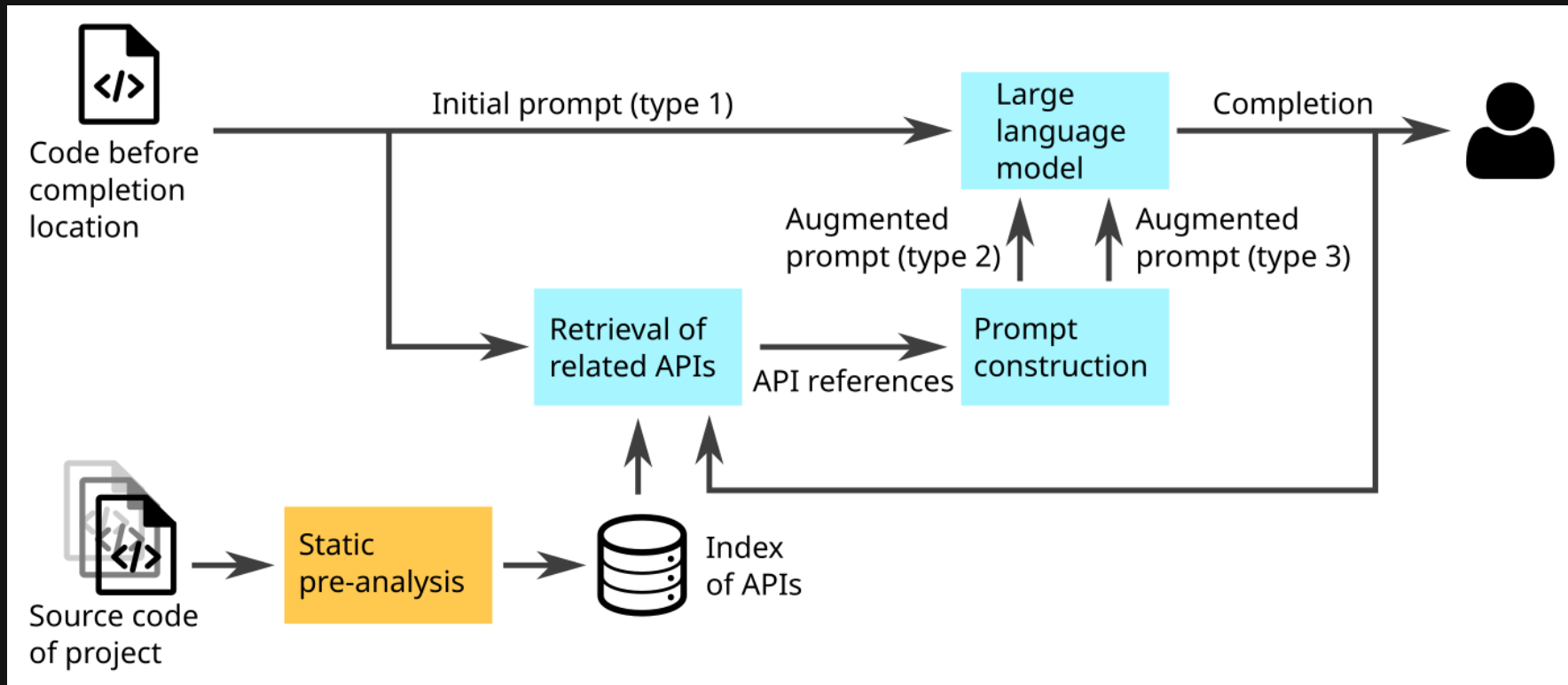
```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    docs_scores = [(doc, compute_relevance_score(doc, keyword) ) for doc in docs]
    sorted_docs_scores = sorted(docs_scores, key=lambda x: x[1], reverse=True)
    return [doc_score[0] for doc_score in sorted_docs_scores[:top_k]]
```



Prediction by ChatGPT's model

Problem: Hallucination

De-Hallucinator



An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
```


An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    return sorted(docs, key=lambda doc: doc.relevance, reverse=True)[:top_k]
```

An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    return sorted(docs, key=lambda doc: doc.relevance, reverse=True)[:top_k]
```

Augmented prompt:

```
# API Reference:
# relevance(document: str, keyword: str) -> float # Returns the relevance of the document to the
# keyword.
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
```

An API somewhere in our project:

```
def relevance(document: str, keyword: str) -> float:
    """Returns the relevance of the document to the keyword."""
    return document.count(keyword) / len(document)
```

Code we want to complete:

```
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    return sorted(docs, key=lambda doc: doc.relevance, reverse=True)[:top_k]
```

Augmented prompt:

```
# API Reference:
# relevance(document: str, keyword: str) -> float # Returns the relevance of the document to the
# keyword.
def search(ds: DataStore, keyword: str, top_k: int) -> List[str]:
    """Returns the top_k most relevant documents that contain the keyword sorted by relevance."""
    docs = ds.find_by_keyword(keyword)
    return sorted(docs, key=lambda doc: relevance(keyword, doc), reverse=True)[:top_k]
```


Three Examples

Software development activities
(traditionally done by hand):

- Write code
- Search for bugs
- Fix bugs

Three Examples

Software development activities
(traditionally done by hand):

- Write code
 - Search for bugs
 - Fix bugs
- 

Does AI replace software developers?

**Powerful tool, but (so far) only for
small-scale code completion**

Motivation

Example 1:

```
if len(bits) != 4 or len(bits) != 6:  
    raise template.TemplateSyntaxError("%r takes  
        exactly four or six arguments (second argument  
        must be 'as' )" % str(bits[0]))
```

Motivation

Example 1:

Always True

Doesn't
match the
message

```
if len(bits) != 4 or len(bits) != 6:  
    raise template.TemplateSyntaxError("%r takes  
        exactly four or six arguments (second argument  
        must be 'as' )" % str(bits[0]))
```


Motivation

Example 1: **Always True**

```
if len(bits) != 4 or len(bits) != 6:  
    raise template.TemplateSyntaxError("%r takes  
        exactly four or six arguments (second argument  
        must be 'as')" % str(bits[0]))
```

**Doesn't
match the
message**

Example 2:

```
if n2 > n1 :  
    raise ValueError('Total internal reflection  
        impossible for n1 > n2')
```

Motivation

Example 1:

Always True

```
if len(bits) != 4 or len(bits) != 6:  
    raise template.TemplateSyntaxError("%r takes  
        exactly four or six arguments (second argument  
        must be 'as' )" % str(bits[0]))
```

Doesn't
match the
message

Example 2:

Condition and
message are
inconsistent

```
if n2 > n1 :  
    raise ValueError('Total internal reflection  
        impossible for n1 > n2')
```

CMI-Finder

Goal:

Detect condition-message inconsistencies

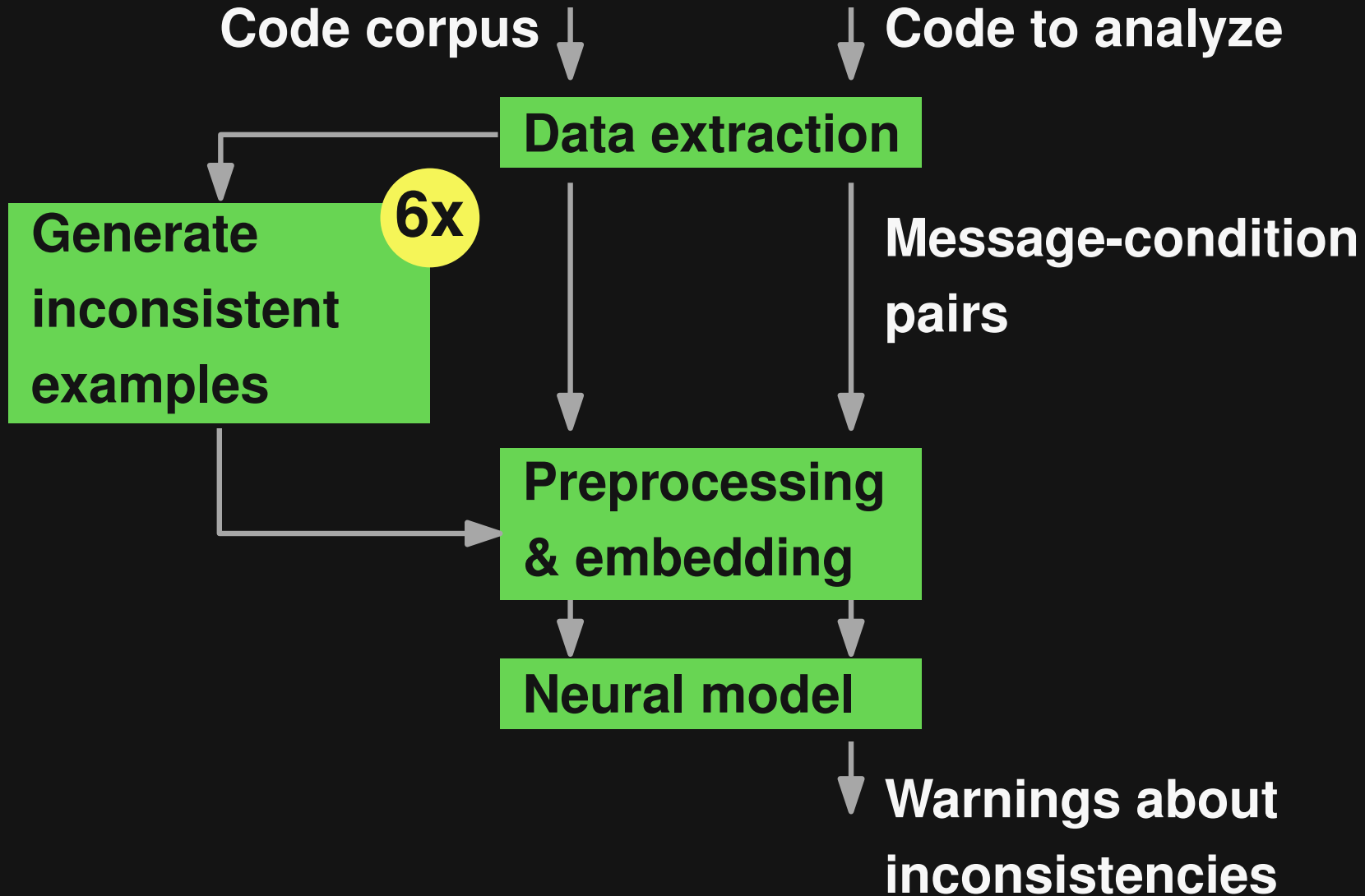
■ Why?

- Incorrect conditions may raise unnecessary warnings or suppress expected warnings
- Incorrect messages make debugging unnecessarily hard

■ Hard problem!

- Must understand both NL and PL

Overview of CMI-Finder



Does It Work?

- **78% precision and 72% recall**
on historic bugs
- **50 new inconsistencies** in previously
unseen projects
- **Complements** traditional **linters**

Does It Work?

- **78% precision and 72% recall** on historic bugs
- **50 new inconsistencies** in previously unseen projects
- **Complements** traditional **linters**

But: Limited to a specific kind of bug

Three Examples

Software development activities
(traditionally done by hand):

- Write code
- Search for bugs
- Fix bugs

Three Examples

Software development activities
(traditionally done by hand):

- Write code
- Search for bugs
- Fix bugs



Does AI replace software developers?

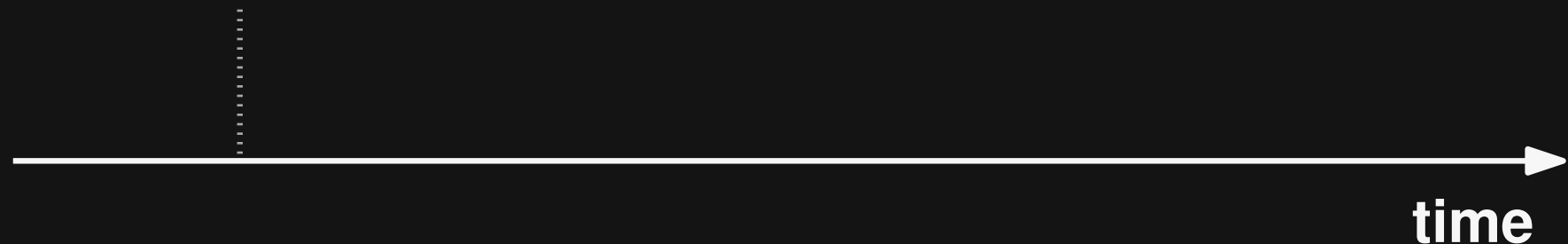
Makes bug detection easier, but human judgement still required

Types in Python

Typical evolution of a Python project:

Code without
type annotations

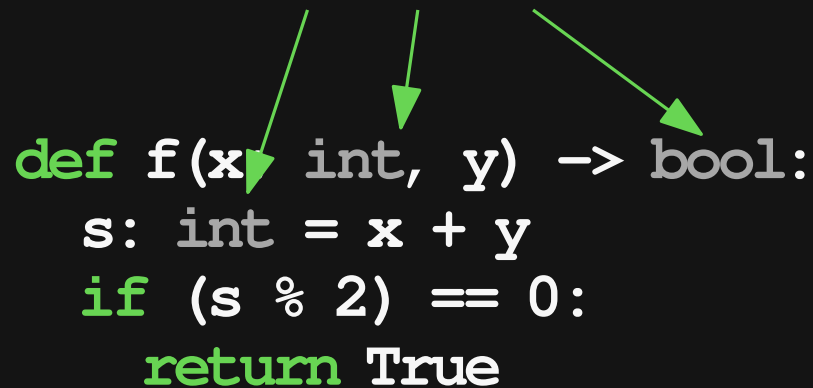
```
def f(x, y):  
    s = x + y  
    if (s % 2) == 0:  
        return True
```



Types in Python

Typical evolution of a Python project:

Partially annotated code



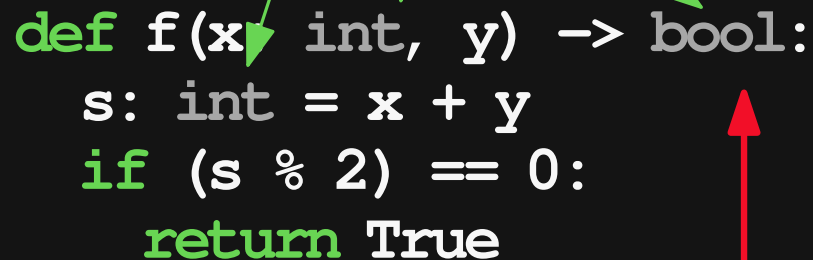
```
def f(x: int, y) -> bool:
    s: int = x + y
    if (s % 2) == 0:
        return True
```

time

Types in Python

Typical evolution of a Python project:

Partially annotated code



```
def f(x: int, y) -> bool:
    s: int = x + y
    if (s % 2) == 0:
        return True
```

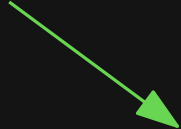
Type error!

time

Types in Python

Typical evolution of a Python project:

Fixed type error



```
def f(x: int, y) -> Optional[bool]:  
    s: int = x + y  
    if (s % 2) == 0:  
        return True
```

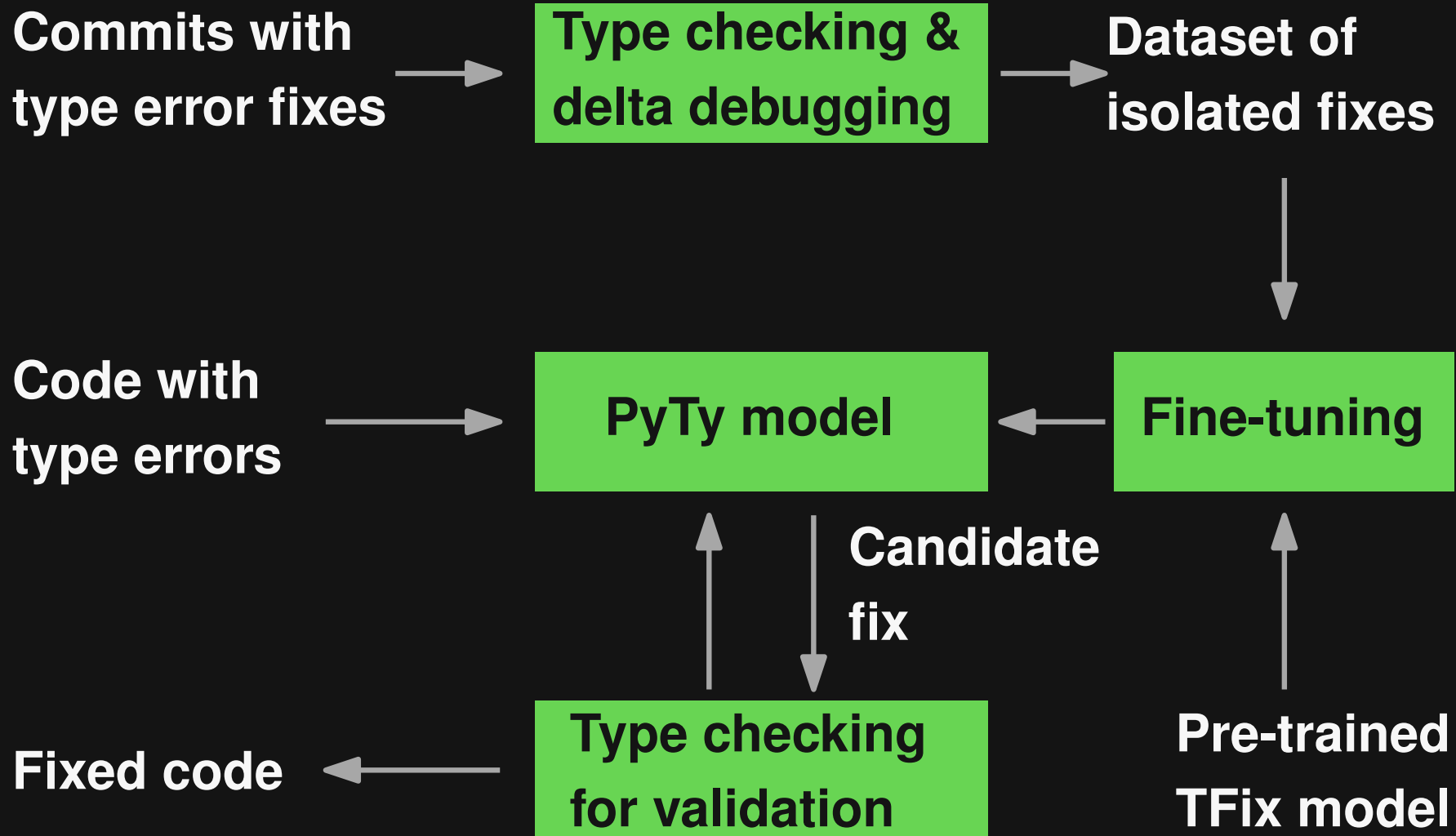


time

Too Many Type Errors

- Most **existing Python code** bases:
Plenty of **static type errors**
- **Easy to detect** by gradual type checker
- But: **No time to fix** them all

PyTy: Approach



Does It Work?

Classes of type errors	Samples (test set)		Effectiveness of PyTy	
			Error removal	Exact match
Incompatible variable type	821	(83)	90.4%	65.1%
Incompatible parameter type	600	(60)	80.0%	36.7%
Incompatible return type	296	(30)	73.3%	43.3%
Invalid type	291	(30)	100.0%	83.3%
Unbound name	258	(26)	76.9%	42.3%
Incompatible attribute type	258	(26)	92.3%	73.1%
Unsupported operand	124	(13)	76.9%	38.5%
Strengthened precondition	59	(6)	83.3%	50.0%
Weakened postcondition	51	(6)	50.0%	0.0%
Call error	8	(1)	100.0%	100.0%
Total	2,766	(281)	85.4%	54.4%

Examples

Code with **type error**:

```
vprint(f"{prefix} {lineno}: {action_name}  
  Constrain Mouse: {'yes' if constraint > 0  
    else ('no' if constrained == 0 else 'check stack') }")  
                        Unbound name
```

PyTy finds **exactly the developer fix**:

```
vprint(f"{prefix} {lineno}: {action_name}  
  Constrain Mouse: {'yes' if constraint > 0  
    else ('no' if constraint == 0 else 'check stack') }")
```

Examples

Code with **type error**: Declared to have type `str`

but used as bytes

```
string = _fmt(string)
return lib.TCOD_console_get_height_rect_fmt(
    self.console_c, x, y, width, height, string
)
```

PyTy finds a **valid fix**:

```
byte_string = _fmt(string)
return lib.TCOD_console_get_height_rect_fmt(
    self.console_c, x, y, width, height, byte_string
)
```

Developer fix (semantically equivalent):

```
return lib.TCOD_console_get_height_rect_fmt(
    self.console_c, x, y, width, height, _fmt(string)
)
```

Three Examples

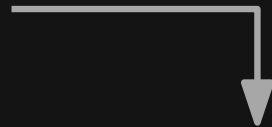
Software development activities
(traditionally done by hand):

- Write code
- Search for bugs
- Fix bugs

Three Examples

Software development activities
(traditionally done by hand):

- Write code
- Search for bugs
- Fix bugs



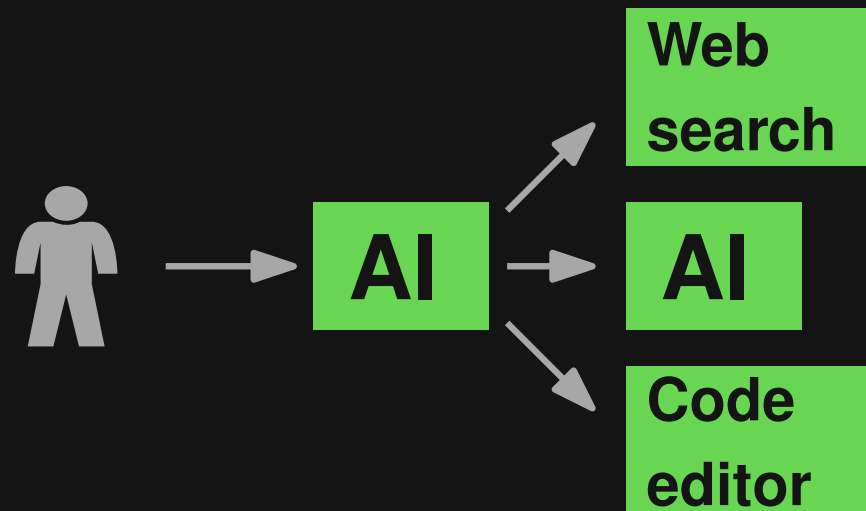
Does AI replace software developers?

**Successfully automates bug fixing for a
specific class of bugs**

What's Next?

Current **AI-based development tools** are only the beginning!

- **Autonomous** software development agents



- **Neuro-symbolic** program analysis



Big Picture

Key feature of humans:
Ability to develop tools



**Software
development
tools**

Big Picture

Key feature of humans:
Ability to develop tools



**Software
development
tools**

Traditionally:
**Compilers and
hand-crafted
program analyses**



Now:
**Learning-based
tools**

**Does AI replace
software developers?**

**Does AI replace
software developers?**

**Yes, but only those
who don't adapt**