PracExtractor: Extracting Configuration Good Practices from Manuals to Detect Server Misconfigurations

Chengcheng Xiang[1], Haochen Huang[1], Andrew Yoo[1], Yuanyuan Zhou[1], Shankar Pasupathy[2]

1 UC San Diego

2 NetApp™

Our lives are largely served by online services today

What serve us are these powerful and complex data center systems

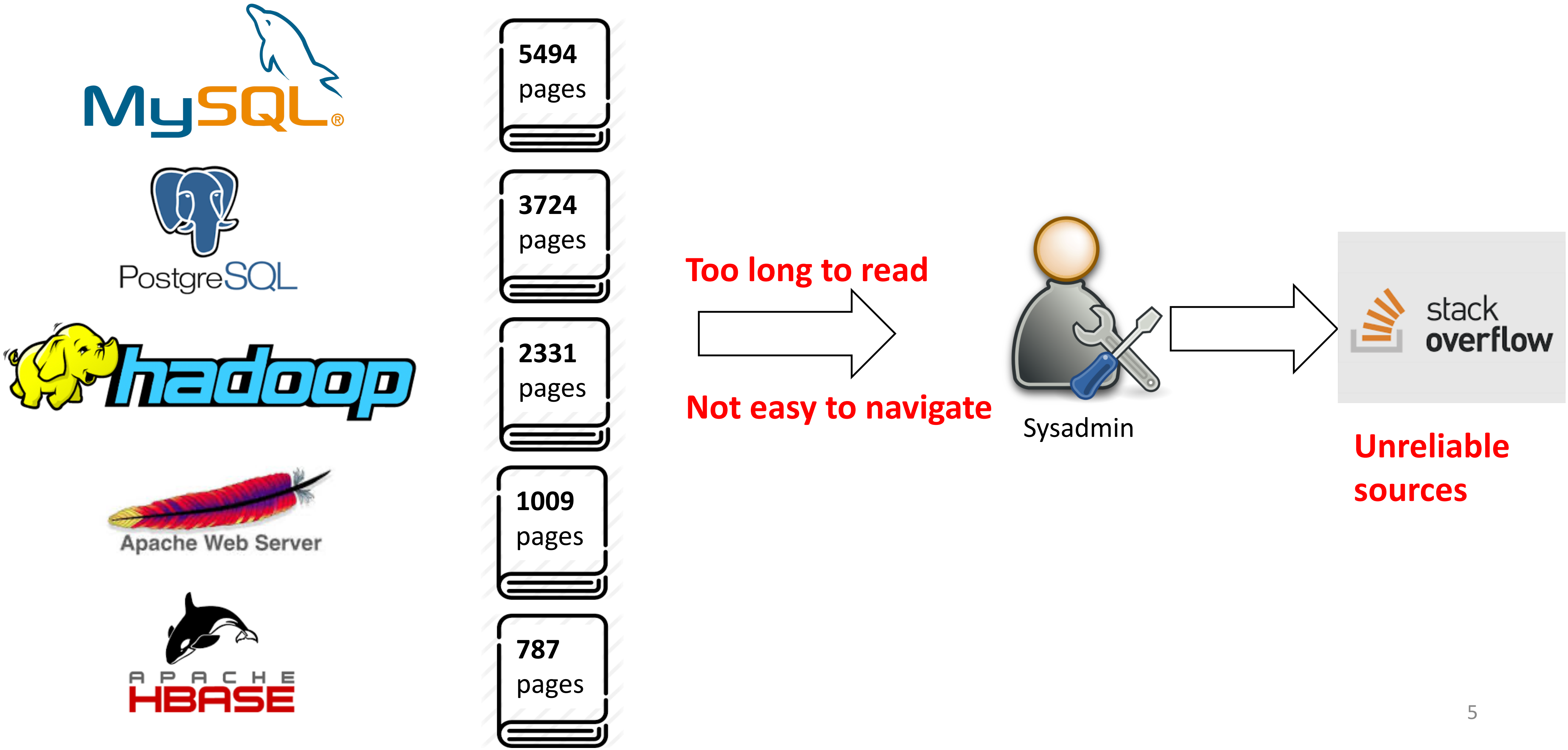
In particular: data center **configuration** has become highly complex

- Too many config parameters

- Parameters are correlated

	No. of parameters
hadoop	1376
MySQL®	940
Apache Web Server	669
PostgreSQL	426

Software release large manuals to assist sysadmins with configurations



5494 pages

3724 pages

2331 pages

1009 pages

787 pages

Too long to read

Not easy to navigate

Sysadmin

Unreliable sources

Is there any useful information that can be automatically extract from manuals?

- **Yes! Good Practices**
  - Describe how to set parameters in a good way from usage experiences
  - Examples

Software	parameter	Good practices	Violation outcomes
Httpd	ExtendedStatus	For highest performance, set ExtendedStatus off.	Performance downgrade
HBase	hbase.regionserver.thrift.framed	Setting this to false will select the default transport, vulnerable to DoS.	Vulnerable to DoS attack
Cassandra	enable_transient_replication	Transient replication is experimental and is not recommended for production use.	Unreliable service

How useful are the good practices in manuals?

Q1: Are good practices specific or general?

General good practices like “set to a large value” are not helpful.

Q2: Are good practices already checked in source code?

If they are, it is non-necessary to extract them from manuals.

Q3: Are good practices always equivalent to default settings?

If they are, then sysadmins can just leave configurations as default.

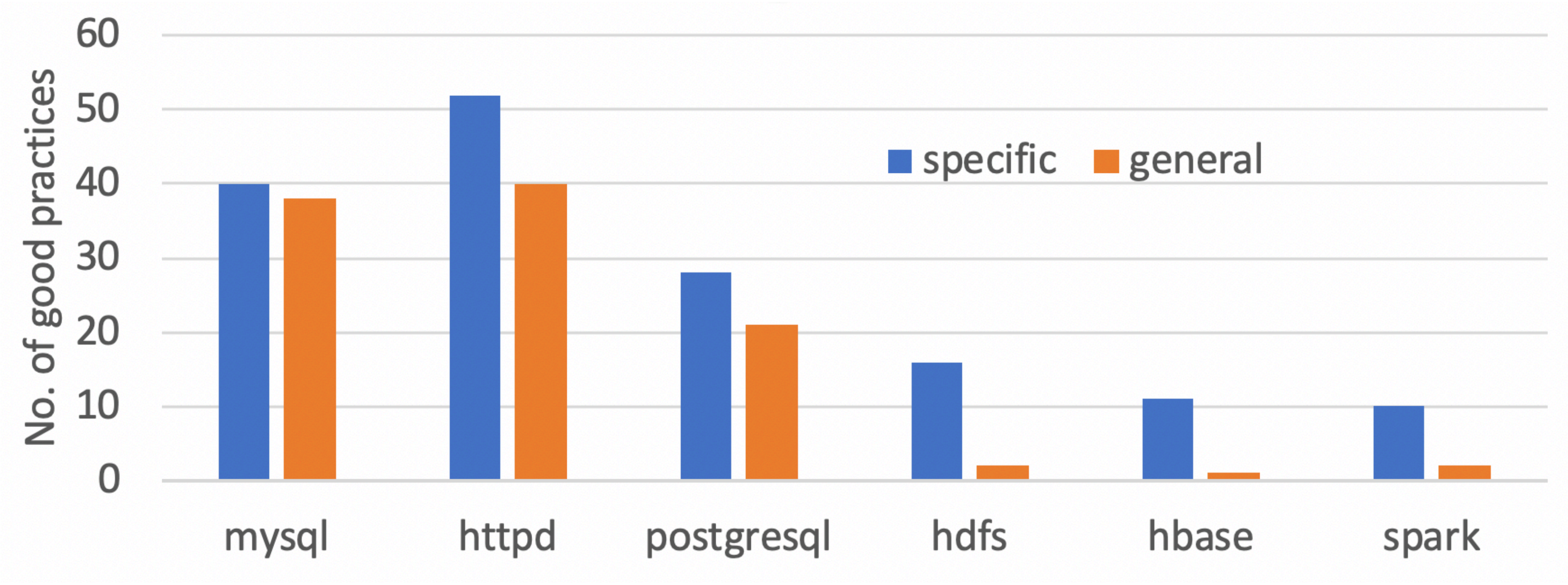
We collected **261 good practices** from **six software manuals** to answer these questions

How useful are the good practices in manuals?

Q1: Are good practices specific or general?

General advice like “set to a large value” is not helpful.

Answer: 60% of studied good practices are specific.

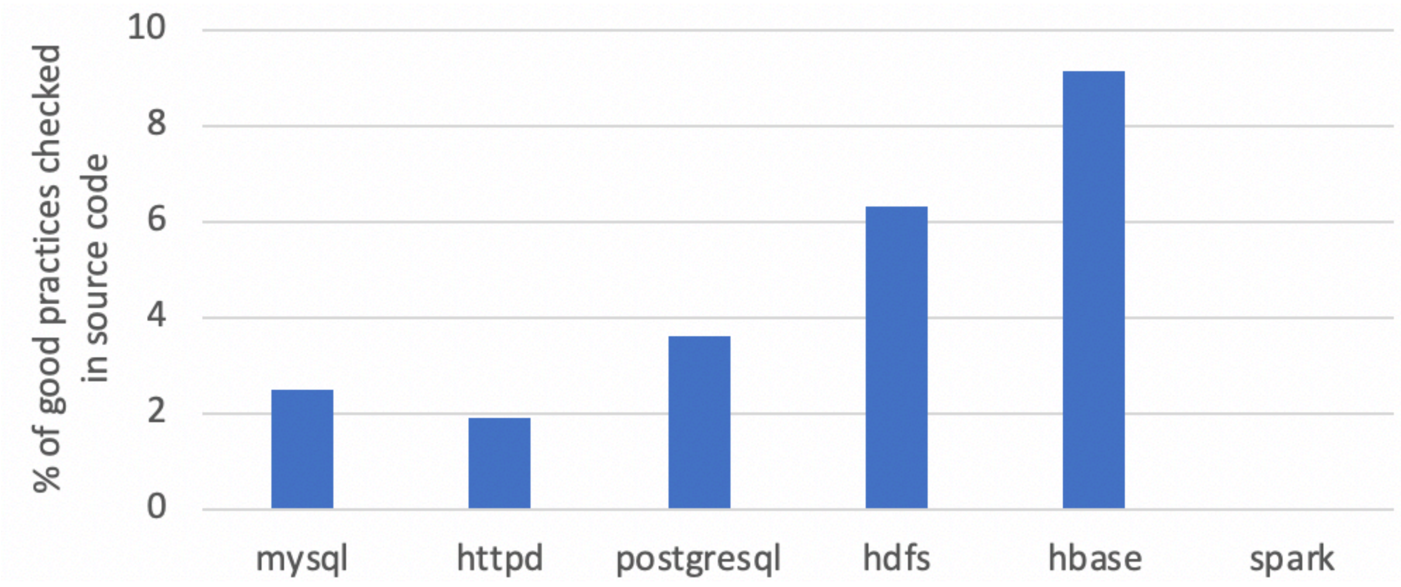


How useful are the good practices in manuals?

Q2: Are good practices already checked in source code?

If they are, it is non-necessary to extract them from manuals.

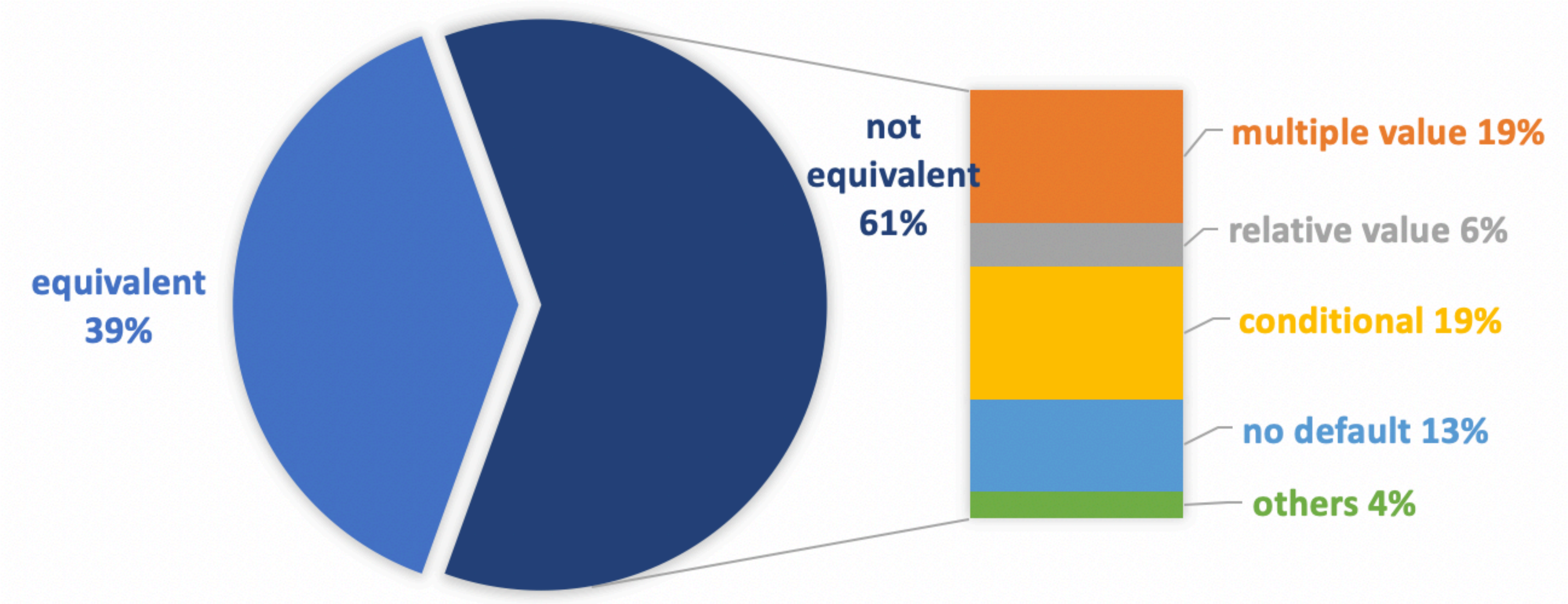
Answer: *only 3% of specific good practices are checked in source code.*



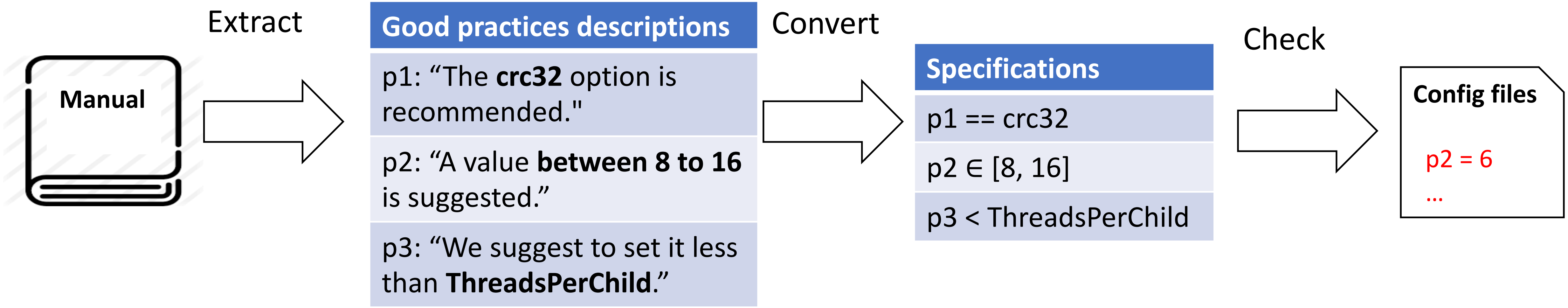
How useful are the good practices in manuals?

Q3: Are good practices always equivalent to default settings?
If they are, then sysadmins can just leave configurations as default.

Answer: 61% of specific good practices are not equivalent to default settings



Based on the study we designed PracExtractor to



Manual

Extract

Good practices descriptions
p1: “The **crc32** option is recommended."
p2: “A value **between 8 to 16** is suggested.”
p3: “We suggest to set it less than **ThreadsPerChild**.”

Convert

Specifications
p1 == crc32
p2 ∈ [8, 16]
p3 < ThreadsPerChild

Check

Config files

p2 = 6

…

Two challenges with PracExtractor

How to effective filter noises and extracts only good practice descriptions?

- 99.6% – 97.3% of sentences in manuals are NOT related to good practices.

How to convert good practice descriptions in free-text into checkable specifications?

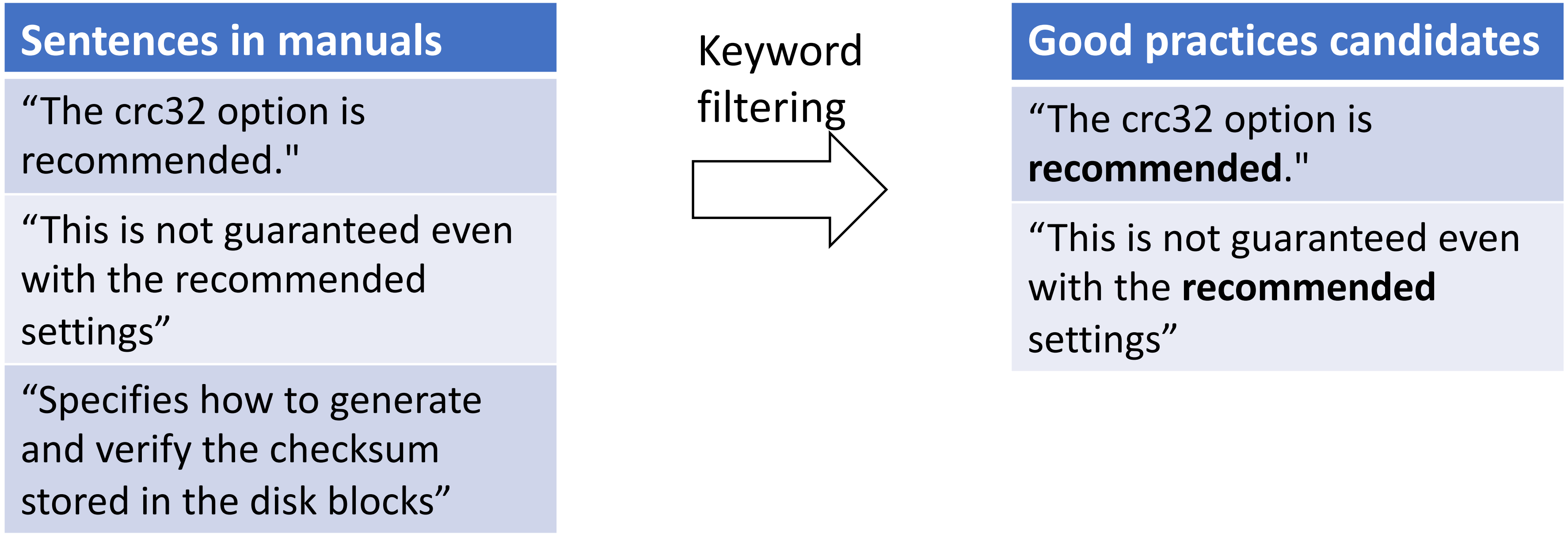
- Sentences like “the crc32 option is recommended” is not directly checkable

Challenge 1: Extract good practice descriptions

- Keyword filtering
- Syntactic-pattern filtering

Challenge 1: Extract good practice descriptions

- Keyword filtering
- Syntactic-pattern filtering



Sentences in manuals
“The crc32 option is recommended."
“This is not guaranteed even with the recommended settings”
“Specifies how to generate and verify the checksum stored in the disk blocks”

Keyword filtering

Good practices candidates
“The crc32 option is **recommended**."
“This is not guaranteed even with the **recommended** settings”

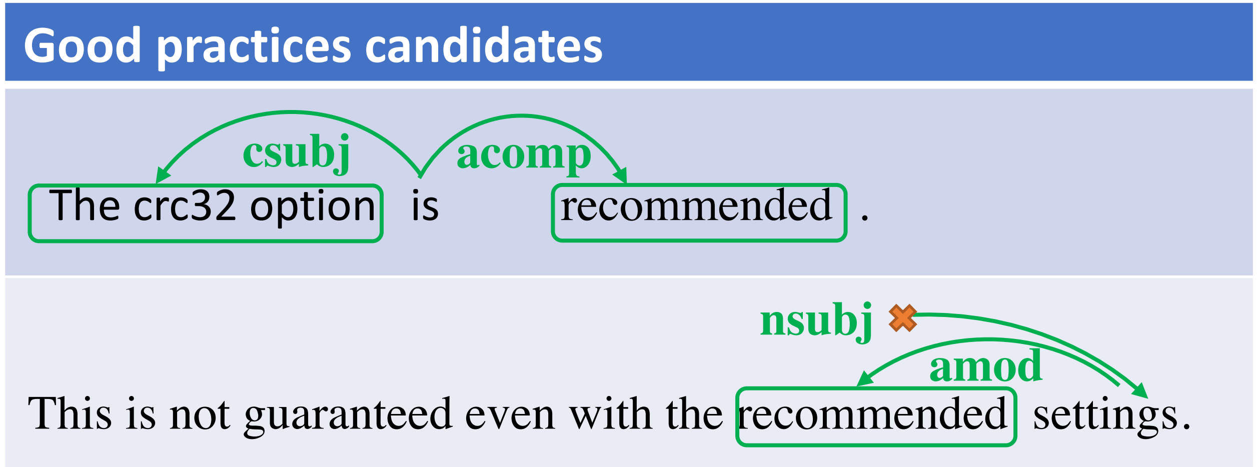
Challenge 1: Extract good practice descriptions

- Keyword filtering
- Syntactic-pattern filtering

Good practices candidates
“The crc32 option is **recommended**."
“This is not guaranteed even with the **recommended** settings”

Challenge 1: Extract good practice descriptions

- Keyword filtering
- Syntactic-pattern filtering



Good practices candidates
The crc32 option is recommended .
This is not guaranteed even with the recommended settings.

Syntactic-pattern filtering

Good practices descriptions
“The crc32 option is recommended."

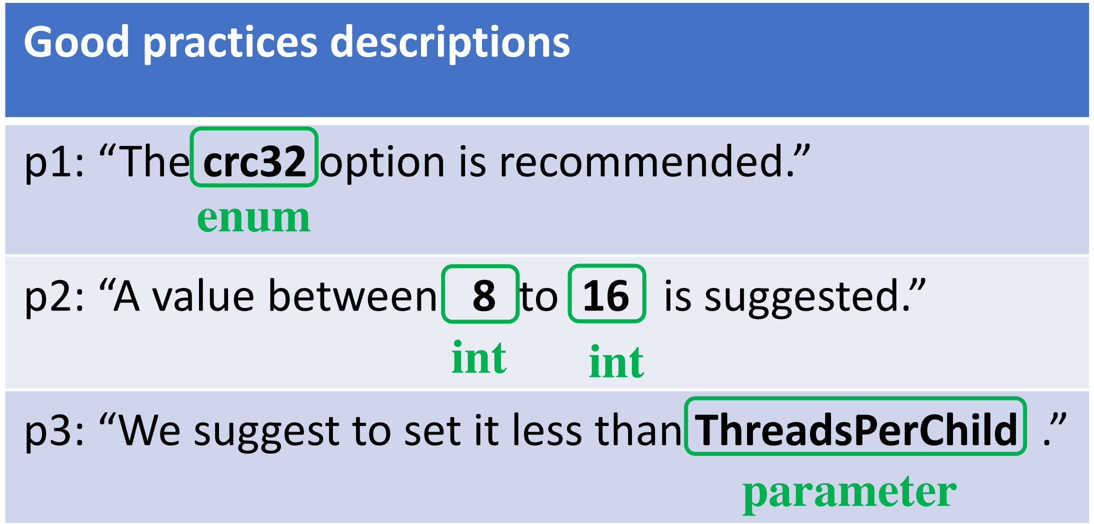
Challenge 2: Convert descriptions into specifications

- Setting entity identification
- Semantic pattern matching

Challenge 2: Convert descriptions into specifications

- Setting entity identification
- Semantic pattern matching

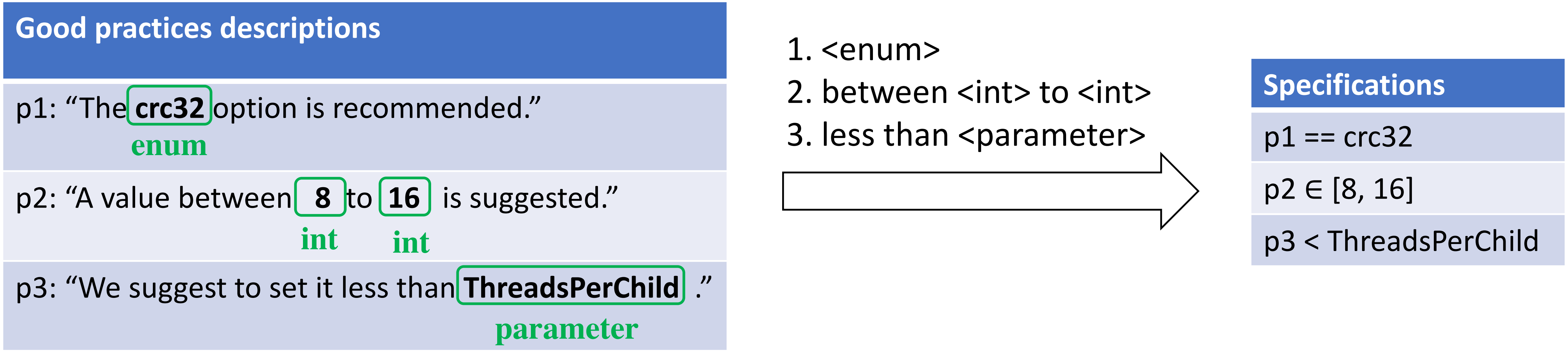
Good practices descriptions
p1: “The crc32 option is recommended.”
p2: “A value between 8 to 16 is suggested.”
p3: “We suggest to set it less than ThreadsPerChild.”



Good practices descriptions
p1: “The **crc32** option is recommended.”
p2: “A value between **8** to **16** is suggested.”
p3: “We suggest to set it less than **ThreadsPerChild** .”

Challenge 2: Convert descriptions into specifications

- Setting entity identification
- Semantic pattern matching



Good practices descriptions
p1: “The **crc32** option is recommended.” enum
p2: “A value between **8** to **16** is suggested.” int int
p3: “We suggest to set it less than **ThreadsPerChild** .” parameter

1. <enum>
2. between <int> to <int>
3. less than <parameter>

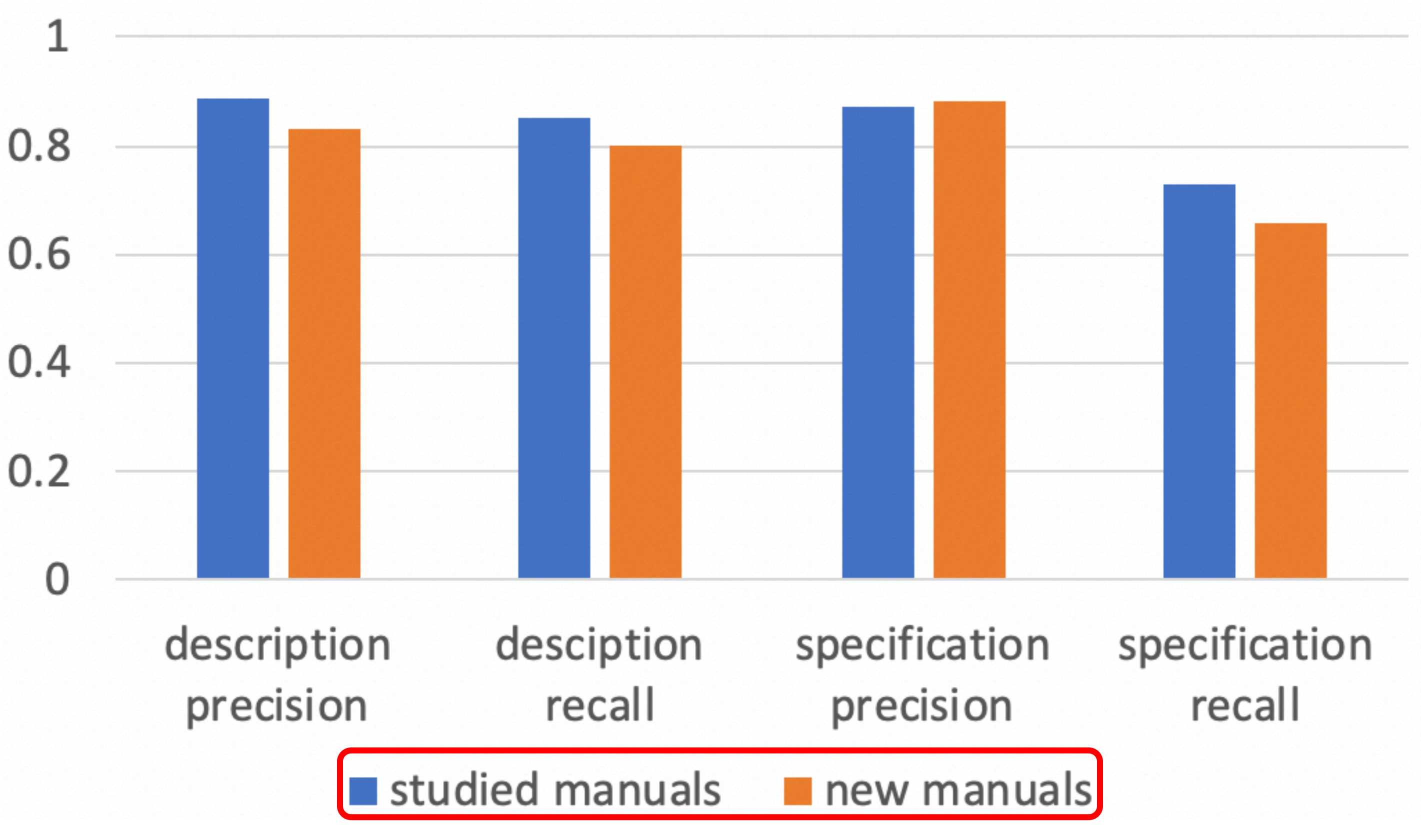
Specifications
p1 == crc32
p2 ∈ [8, 16]
p3 < ThreadsPerChild

Evaluation of PracExtractor

- Extract good practices from software manuals
- Detect real-world configuration errors

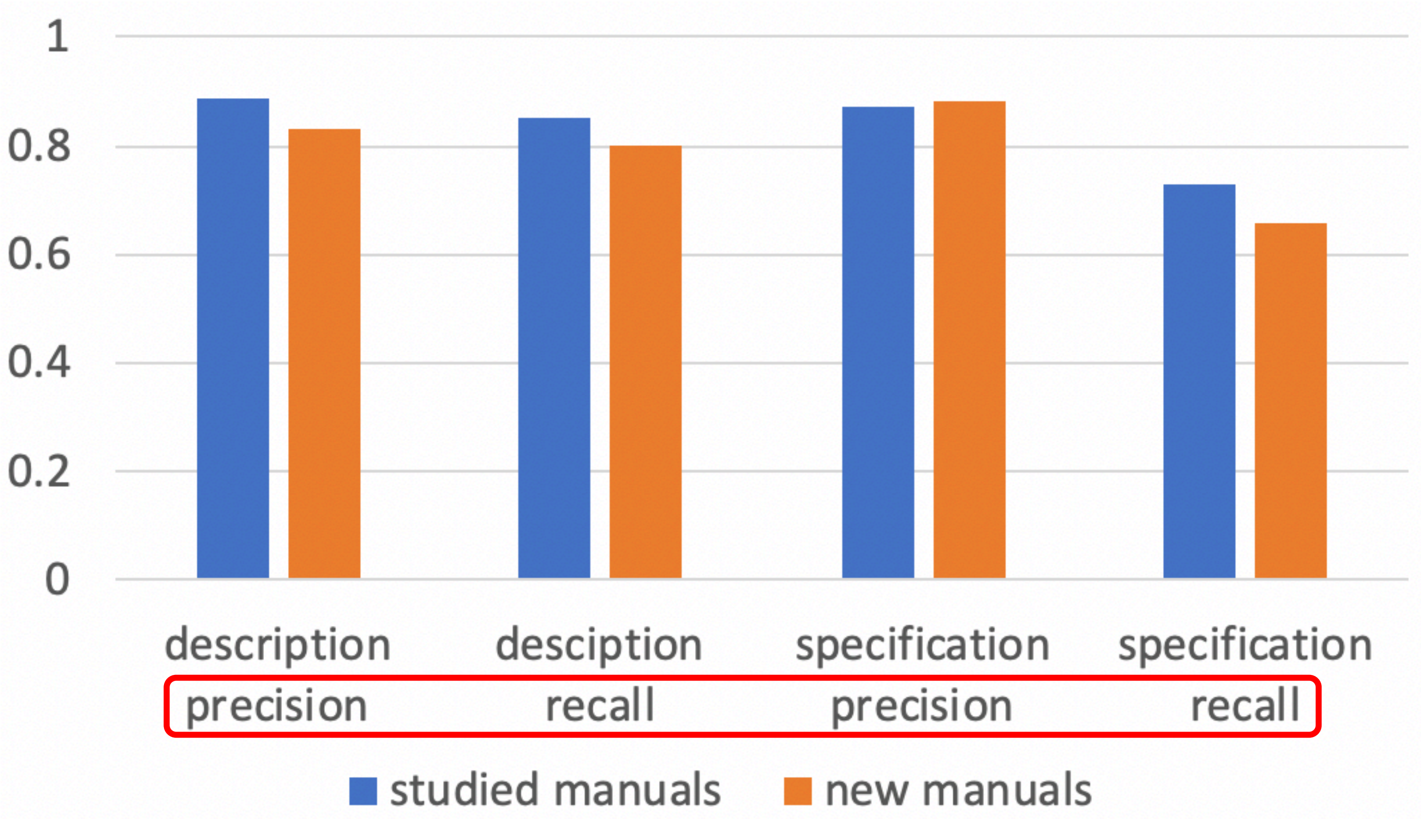
Evaluation of PracExtractor

- Accuracy of good practice extraction
  - Training sets: 6 studied manuals included in our characteristic study
  - Testing sets: 6 new manuals not included in our study



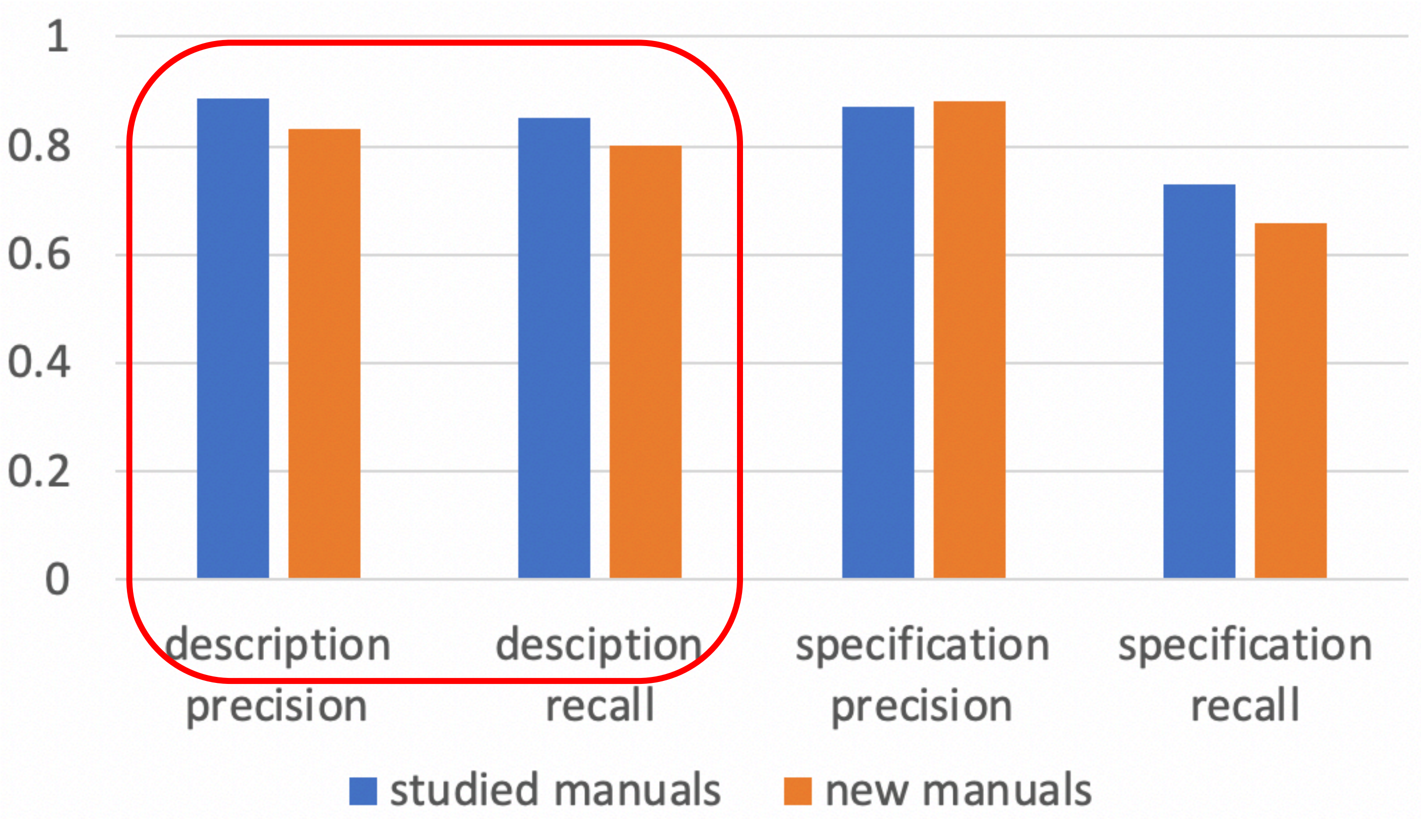
Evaluation of PracExtractor

- Accuracy of good practice extraction
  - Precision: what percentage of good practices extracted are true
  - Recall: what percentage of true good practices are extracted



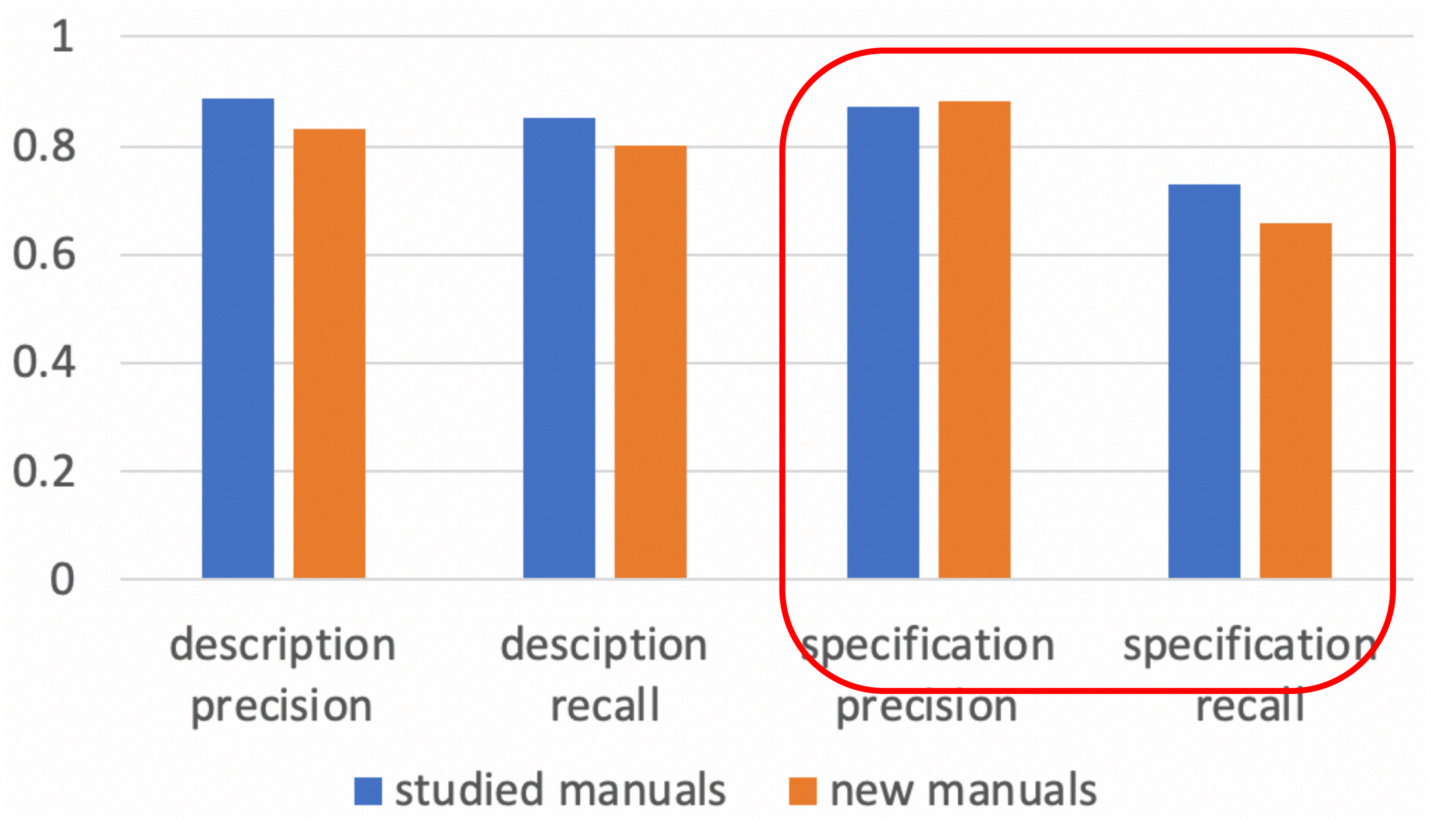
Evaluation of PracExtractor

- Accuracy of good practice extraction
  - Good practice descriptions extraction



Evaluation of PracExtractor

- Accuracy of good practice extraction
  - Good practice specifications extraction

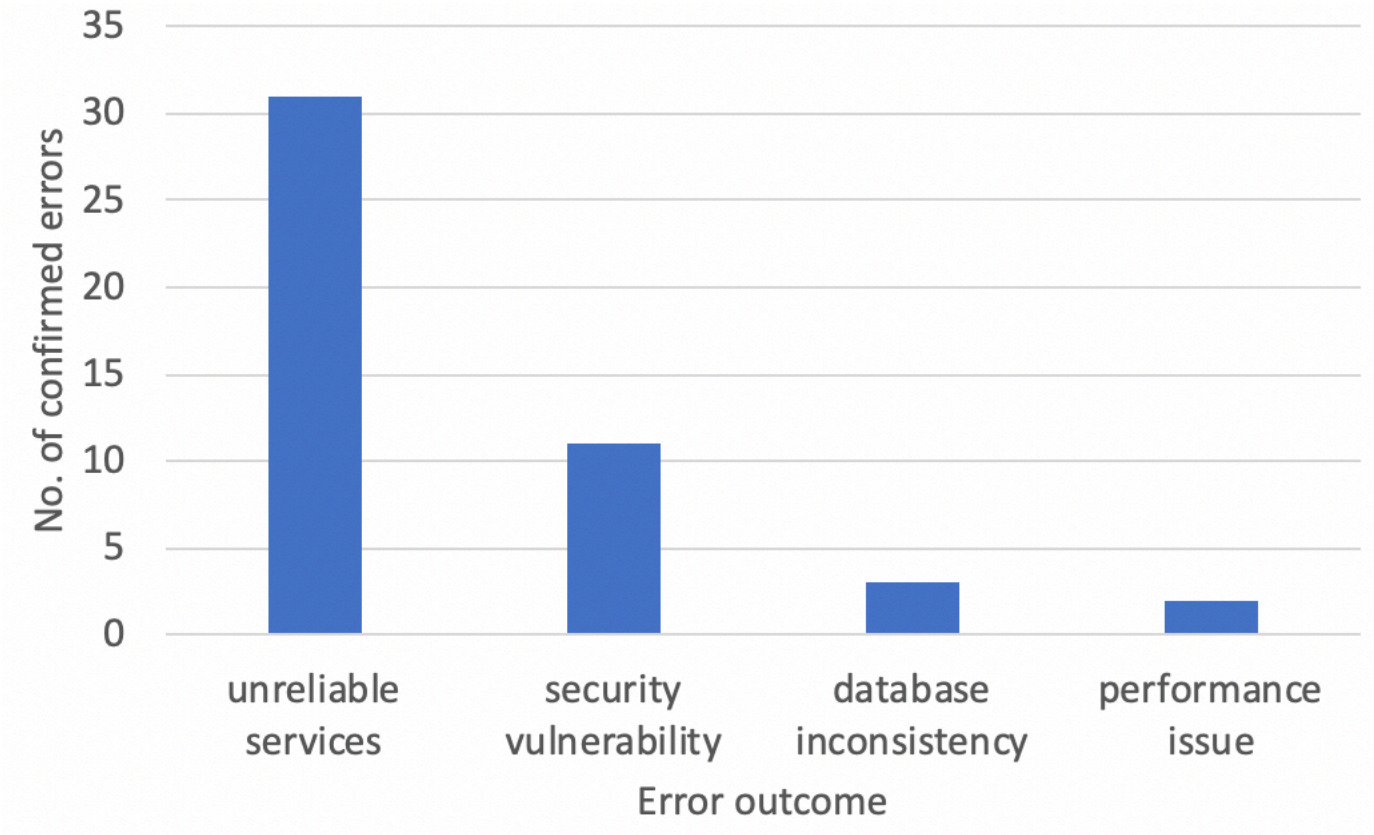


Evaluation of PracExtractor

- Detect real-world configuration errors
  - Downloaded 2200 docker images from docker hub.
  - Detected 1423 practice violations from 853 unique images.
  - Got 47 confirmed as real configuration errors (325 reported in total).

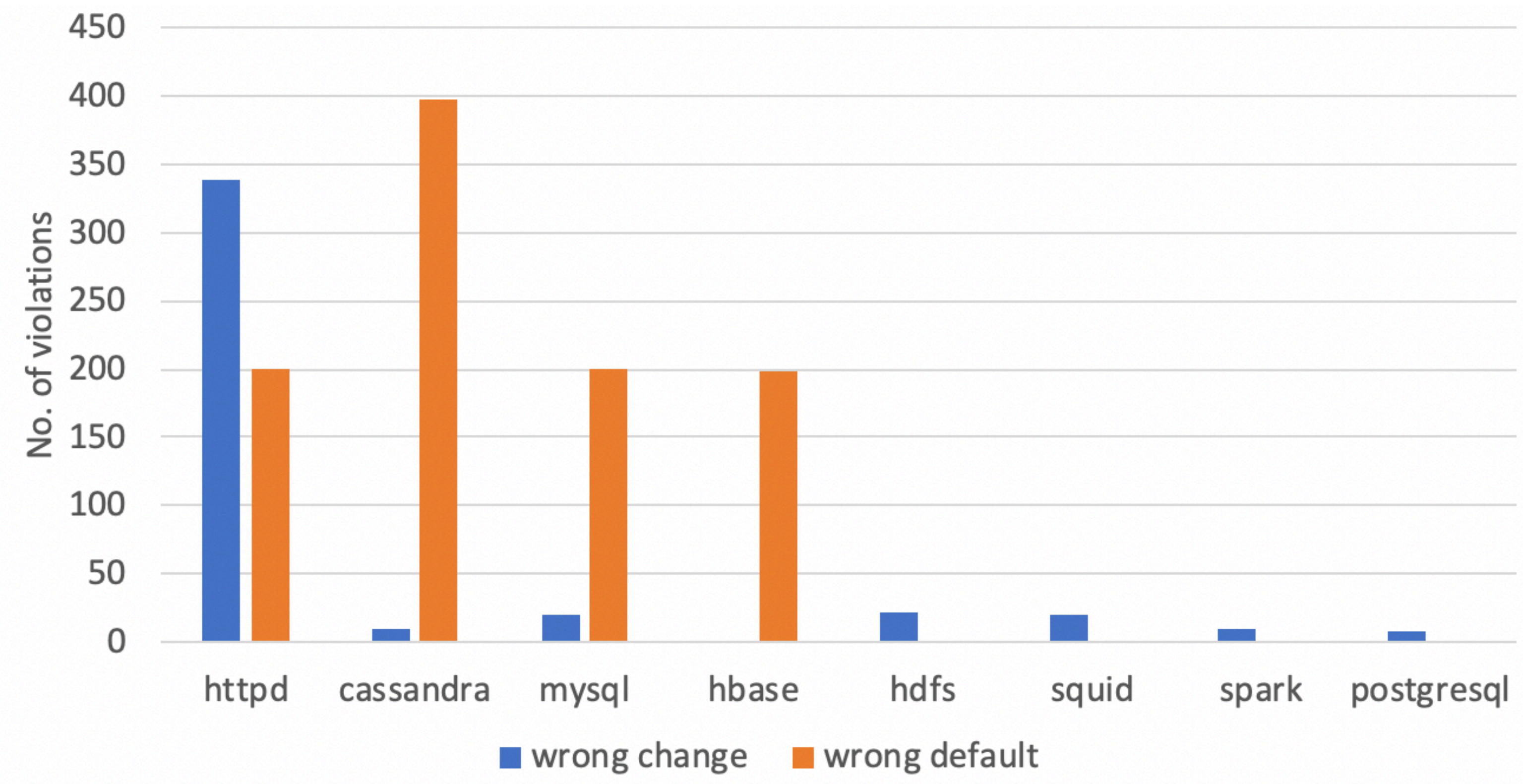
Evaluation of PracExtractor

- Outcome of the confirmed configuration errors



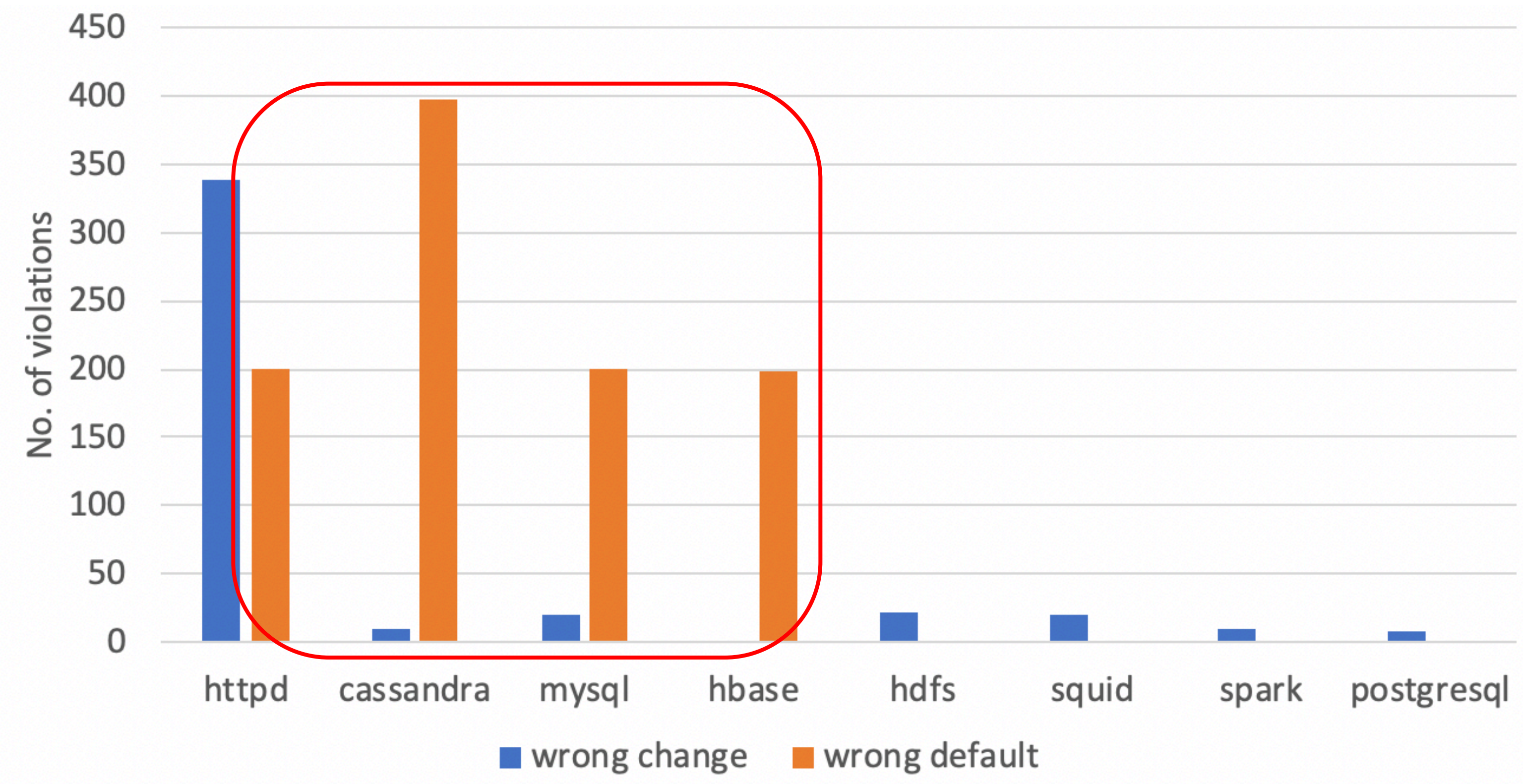
Evaluation of PracExtractor

- Analysis of the detected violations
  - Wrong change: a parameter is changed to a value violating good practices
  - Wrong default: a parameter’s default violate good practices but is not changed



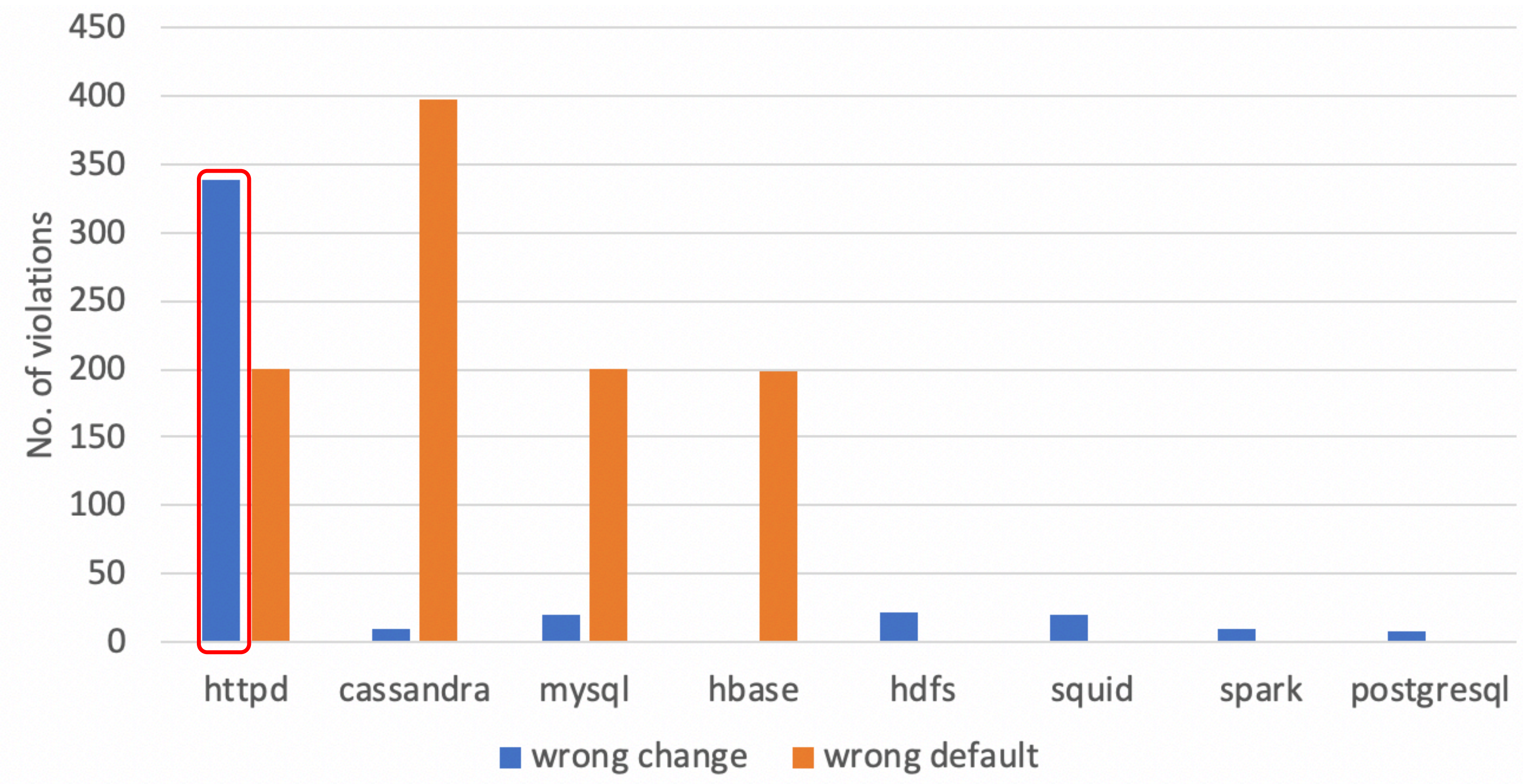
Evaluation of PracExtractor

- Analysis of the detected violations
  - Wrong change: a parameter is changed to a value violating good practices
  - Wrong default: a parameter's default violate good practices but is not changed



Evaluation of PracExtractor

- Analysis of the detected violations
  - Wrong change: a parameter is changed to a value violating good practices
  - Wrong default: a parameter’s default violate good practices but is not changed



Summary of PracExtractor

- Identified good practices as useful information from manuals for configuration validation.
  - Studied **261 good practices** from **six software manuals** to prove usefulness.
- Built PracExtractor to automatically extract good practices from manuals.
  - PracExtractor achieved reasonably high precision and recall.
  - PracExtractor detected 47 real-world configuration errors.

Thank you!

c4xiang@cs.ucsd.edu