Logzip: Extracting Hidden Structures via Iterative Clustering for Log Compression

Jinyang Liu\textsuperscript{1}, Jieming Zhu\textsuperscript{2}, Shilin He\textsuperscript{3}, Pinjia He\textsuperscript{4}, Zibin Zheng\textsuperscript{1, Michael R. Lyu}\textsuperscript{3}

\textsuperscript{1}Sun Yat-Sen University
\textsuperscript{2}Huawei Noah's Ark Lab
\textsuperscript{3}The Chinese University of Hong Kong
\textsuperscript{4}ETH Zurich

Supervisor: Prof. Zibin Zheng and Prof. Michael R. Lyu

The 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)
Systems produce logs to record runtime information
Motivation & Background

System logs are important for

- Diagnose runtime failures
- Identify performance bottlenecks
- Detect security issues
- Market trends prediction
- ......
Motivation & Background

Log data requires **long-term** storage and is **fast-growing**

It is **time-consuming** and **money-consuming**
Writing **less** logging statements in the source code

Risk **missing** key information 😞

Apply **compression tools**: gzip, bzip2, lzma...

**Not specifically designed** for log data 😞

**Logzip**, explores **hidden structures** of log data for better compression 😊
The Chinese University of Hong Kong (CUHK) is a public research university in Shatin, Hong Kong formally established in 1963 by a charter granted by the Legislative Council of Hong Kong. It is the territory's second oldest university and was founded as a federation of three existing colleges.

Motivation & Background

Intuition: repetitive data is more compressible

The Chinese University of Hong Kong is wonderful!
The Chinese University of Hong Kong is wonderful!
The Chinese University of Hong Kong is wonderful!
The Chinese University of Hong Kong is wonderful!
The Chinese University of Hong Kong is wonderful!
The Chinese University of Hong Kong is wonderful!

(≈300 chars)
Motivation & Background

Log Structure

```
logInfo(s"Found block $blockId remotely")
```

HDFS logs (part)
Motivation & Background

Different types of log data share the similar format

```
17/06/09 20:11:11 INFO storage.BlockManager: Found block rdd_42_26 locally
17/06/09 20:11:11 INFO storage.BlockManager: Found block rdd_42_28 locally
17/06/09 20:11:11 INFO storage.BlockManager: Found block rdd_42_27 locally
17/06/09 20:11:11 INFO storage.BlockManager: Found block rdd_42_29 locally
17/06/09 20:11:11 INFO python.PythonRunner: Times: total = 41, boot = 23, init = 17, finish = 1
17/06/09 20:11:11 INFO python.PythonRunner: Times: total = 38, boot = 18, init = 20, finish = 0
17/06/09 20:11:11 INFO python.PythonRunner: Times: total = 42, boot = 18, init = 23, finish = 1
17/06/09 20:11:11 INFO python.PythonRunner: Times: total = 39, boot = 18, init = 20, finish = 1
17/06/09 20:11:11 INFO executor.Executor: Finished task 25.0 in stage 29.0 (TID 1345). 2128 bytes result sent to driver
17/06/09 20:11:11 INFO executor.Executor: Finished task 28.0 in stage 29.0 (TID 1348). 2128 bytes result sent to driver
17/06/09 20:11:11 INFO executor.Executor: Finished task 27.0 in stage 29.0 (TID 1347). 2128 bytes result sent to driver
17/06/09 20:11:11 INFO executor.Executor: Finished task 26.0 in stage 29.0 (TID 1346). 2128 bytes result sent to driver
17/06/09 20:11:11 INFO executor.CoarseGrainedExecutorBackend: Got assigned task 1350
17/06/09 20:11:11 INFO executor.Executor: Running task 30.0 in stage 29.0 (TID 1350)
17/06/09 20:11:11 INFO python.PythonRunner: Times: total = 43, boot = 14, init = 28, finish = 1
```

Spark logs (part)

```
03-17 16:13:38.859 2227 2227 D TextView: visible is system.time.showampm
03-17 16:13:38.861 2227 2227 D TextView: mVisibility.getValue is false
03-17 16:13:38.869 2227 2227 D TextView: visible is system.charge.show
03-17 16:13:38.871 2227 2227 D TextView: mVisibility.getValue is false
03-17 16:13:38.875 2227 2227 D TextView: visible is system.call.count gt 0
03-17 16:13:38.877 2227 2227 D TextView: mVisibility.getValue is false
03-17 16:13:38.881 2227 2227 D TextView: visible is system.message.count gt 0
03-17 16:13:38.882 2227 2227 D TextView: mVisibility.getValue is false
03-17 16:13:38.887 2227 2227 D TextView: visible is system.ownerinfo.show
03-17 16:13:38.888 2227 2227 D TextView: mVisibility.getValue is false
03-17 16:13:38.905 1702 10454 D PowerManagerService: release:lock=233570404, flg=0x0, tag="View Lock",
```

Android logs (part)
Motivation & Background

It is OK to compress the whole file

```
logInfo(s"Found block $blockId remotely")
```
Motivation & Background

It is *better* to compress after hidden structures are extracted.

```
logInfo(s"Found block $blockId remotely")
```
Method: Overview

1. Log Structurization

   - Message Header
     - Date
     - Time
     - Component

   Raw logs

   Message Content

   Structure Extraction
     - Templates
     - EventID
     - Parameters

   Split

   Sub-Fields

   Intermediate Representation

   gzip
   bzip2
   Izma

   Compressed File

2. Structure Extraction

3. Compression
Method: Log Structurization

1. Log Structurization

2. Structure Extraction

3. Compression
Method: Log Structurization

We propose a framework to automatically extract the hidden structure from log messages. The logging framework uses a function `logInfo(s"Found block $blockId remotely")` to log the messages.

### Log Messages

<table>
<thead>
<tr>
<th>Message</th>
<th>Date</th>
<th>Time</th>
<th>Level</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Found block rdd_2_0 locally</td>
<td>17/06/09 20:10:46 INFO storage.BlockManager:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Found block rdd_2_3 locally</td>
<td>17/06/09 20:10:46 INFO storage.BlockManager:</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Date:** changes once a day
- **Time:** 24h, 60m, 60s
- **Level:** INFO, DEBUG, ERROR...
- **Component:** limited numbers

#### Sample regex for Android

```
r'(\[\w\-\]+)+(\([\w\-\]+\)\{2,\}[\w\-\]+)\', r'(\([-\?\+\d]+)\b|\b0[Xx][a-fA-F]\d+\b|\b[a-fA-F]\d\d\b)'
```

- **Date:** Repetitive! Regex!
- **Time:** 24h, 60m, 60s
- **Level:** INFO, DEBUG, ERROR...
- **Component:** limited numbers

Written by human
Non-trivial! Regex?
Method: Structure Extraction

1. Log Structurization

Raw logs → Message Header → Date → Time → Component → Split → Sub-Fields

2. Structure Extraction

Structure Extraction → Templates → EventID → Parameters → Split → Sub-Fields

3. Compression

Intermediate Representation → gzip → bzip2 → Izma → Compressed File
Method: Iterative Structure Extraction (ISE)
Method: Iterative Structure Extraction (ISE)

Sample a small fraction (1%) of the whole dataset
Method: Iterative Structure Extraction (ISE)

Apply sequential clustering to extract templates
Method: Iterative Structure Extraction (ISE)

Match unsampled data with extracted templates
Method: Iterative Structure Extraction (ISE)

Mismatched data goes through the process iteratively
Method: Iterative Structure Extraction (ISE)
Method: ISE-Clustering & Template Extraction

Workflow of Sequential Clustering
Method: ISE-Clustering & Template Extraction

Workflow of Sequential Clustering
Method: ISE-Matching

Input 1: A tokenized log message

A B C *
A B G
A K E
A K F

Build prefix tree

Search

Input 2: Templates

<table>
<thead>
<tr>
<th>Content</th>
<th>Template ID</th>
<th>Template</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>A B C D</td>
<td>E1</td>
<td>A B C *</td>
<td>D</td>
</tr>
</tbody>
</table>

Method: ISE-Matching
Method: Iterative Structure Extraction (ISE)

Mismatched data goes through the process iteratively
Method: Example of Logzip

**Raw Logs**

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Level</th>
<th>Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>17/06/09</td>
<td>20:10:46</td>
<td>INFO</td>
<td>storage.BlockManager:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Found block rdd_2_0 locally</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:46</td>
<td>INFO</td>
<td>storage.BlockManager:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Found block rdd_2_3 locally</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:52</td>
<td>INFO</td>
<td>spark.CacheManager:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Partition rdd_2_1 not found, computing it</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:52</td>
<td>INFO</td>
<td>spark.CacheManager:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Partition rdd_2_3 not found, computing it</td>
</tr>
</tbody>
</table>

**Log Structurization**

<table>
<thead>
<tr>
<th>DATE</th>
<th>TIME</th>
<th>LEVEL</th>
<th>COMPONENT</th>
</tr>
</thead>
<tbody>
<tr>
<td>17/06/09</td>
<td>20:10:46</td>
<td>INFO</td>
<td>storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BlockManager:</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:46</td>
<td>INFO</td>
<td>storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BlockManager:</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:46</td>
<td>INFO</td>
<td>storage</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>BlockManager:</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:52</td>
<td>INFO</td>
<td>spark</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CacheManager:</td>
</tr>
<tr>
<td>17/06/09</td>
<td>20:10:52</td>
<td>INFO</td>
<td>spark</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>CacheManager:</td>
</tr>
</tbody>
</table>

**Field Extraction**

<table>
<thead>
<tr>
<th>[17 / 06 / 09] 20 : 10 : 46</th>
<th>INFO</th>
<th>storage</th>
<th>BlockManager:</th>
</tr>
</thead>
<tbody>
<tr>
<td>[17 / 06 / 09] 20 : 10 : 46</td>
<td>INFO</td>
<td>storage</td>
<td>BlockManager:</td>
</tr>
<tr>
<td>[17 / 06 / 09] 20 : 10 : 46</td>
<td>INFO</td>
<td>storage</td>
<td>BlockManager:</td>
</tr>
<tr>
<td>[17 / 06 / 09] 20 : 10 : 52</td>
<td>INFO</td>
<td>spark</td>
<td>CacheManager:</td>
</tr>
<tr>
<td>[17 / 06 / 09] 20 : 10 : 52</td>
<td>INFO</td>
<td>spark</td>
<td>CacheManager:</td>
</tr>
</tbody>
</table>

**Structure Extraction & Mapping**

**Templates**

| 1: Found block *locally, |
| 2: Partition * not found, |

**Parameters**

| 1: rdd, 2: 3, 4: 0, 5: 3, 6: 1 |

<table>
<thead>
<tr>
<th>EventID</th>
<th>ParaID</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 2, 3, 2, 4</td>
<td>1, 2, 3, 2, 5</td>
</tr>
</tbody>
</table>

**Kernel Compression**

<table>
<thead>
<tr>
<th>Sub-Field Object</th>
<th>Sub-Field Object</th>
<th>Sub-Field Object</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>Parameter Mapping Object</th>
<th>Template Mapping Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>EventID Object</td>
<td>ParaID Object</td>
</tr>
</tbody>
</table>
Method: Example of Logzip

Raw Logs

17/06/09 20:10:46 INFO storage.BlockManager: Found block rdd_2_0 locally
17/06/09 20:10:46 INFO storage.BlockManager: Found block rdd_2_3 locally
17/06/09 20:10:52 INFO spark.CacheManager: Partition rdd_2_1 not found, computing it
17/06/09 20:10:52 INFO spark.CacheManager: Partition rdd_2_3 not found, computing it

① Log Structurization

DATE TIME LEVEL COMPONENT
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:52 INFO spark.CacheManager:
17/06/09 20:10:52 INFO spark.CacheManager:

MESSAGE CONTENT
Found block rdd_2_0 locally
Found block rdd_2_3 locally
Partition rdd_2_1 not found, computing it
Partition rdd_2_3 not found, computing it

② Field Extraction

[17/06/09 20:10:46 INFO storage.BlockManager:]
[17/06/09 20:10:46 INFO storage.BlockManager:]
[17/06/09 20:10:52 INFO spark.CacheManager:]
[17/06/09 20:10:52 INFO spark.CacheManager:]

③ Structure Extraction & Mapping

Templates
1: Found block *locally,
2: Partition * not found, computing it

Parameters
1: rdd, 2: 3, 4: 5, 6: 1

④ EventID: ParalID

1, 2, 3, 4
1, 2, 3, 5
1, 2, 3, 6
1, 2, 3, 5

⑤ Kernel Compression

Sub-Field Object Sub-Field Object Sub-Field Object

Parameter Mapping Object Template Mapping Object EventID ParalID Object

26
Method: Example of Logzip

Raw Logs
17/06/09 20:10:46 INFO storage.BlockManager: Found block rdd_2_0 locally
17/06/09 20:10:46 INFO storage.BlockManager: Found block rdd_2_3 locally
17/06/09 20:10:52 INFO spark.CacheManager: Partition rdd_2_1 not found, computing it
17/06/09 20:10:52 INFO spark.CacheManager: Partition rdd_2_3 not found, computing it

Log Structurization
DATE TIME LEVEL COMPONENT
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:52 INFO spark.CacheManager:
17/06/09 20:10:52 INFO spark.CacheManager:

MESSAGE CONTENT
Found block rdd_2_0 locally
Found block rdd_2_3 locally
Partition rdd_2_1 not found, computing it
Partition rdd_2_3 not found, computing it

Field Extraction
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:52 INFO spark.CacheManager:
17/06/09 20:10:52 INFO spark.CacheManager:

Structure Extraction & Mapping
Templates
1: Found block *locally, 2: Partition * not found, computing it
Parameters
1: rdd, 2: 3-2, 3: 4-0, 5: 3-6, 6: 1

Kernel Compression
Sub-Field Object Sub-Field Object Sub-Field Object
Parameter Mapping Object Template Mapping Object EventID Object ParaOID Object
Method: Example of Logzip

Raw Logs
17/06/09 20:10:46 INFO storage.BlockManager: Found block rdd_2_0 locally
17/06/09 20:10:46 INFO storage.BlockManager: Found block rdd_2_3 locally
17/06/09 20:10:52 INFO spark.CacheManager: Partition rdd_2_1 not found, computing it
17/06/09 20:10:52 INFO spark.CacheManager: Partition rdd_2_3 not found, computing it

① Log Structurization

DATE TIME LEVEL COMPONENT
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:46 INFO storage.BlockManager:
17/06/09 20:10:52 INFO spark.CacheManager:
17/06/09 20:10:52 INFO spark.CacheManager:

② Field Extraction

③ Structure Extraction & Mapping

MESSAGE CONTENT
Found block rdd_2_0 locally
Found block rdd_2_3 locally
Partition rdd_2_1 not found, computing it
Partition rdd_2_3 not found, computing it

④ Template
{1: Found block *locally,
2: Partition * not found,
Parameters {1:rdd, 2: ___, 3: 2,
4:0, 5:3, 6:1}

⑤ Kernel Compression

Parameter Mapping Object
Mapping Object
EventID Object
ParaID Object...
Method: Example of Logzip

1. Log Structurization
   - DATE: 17/06/09
   - TIME: 20:10:46
   - LEVEL: INFO
   - COMPONENT: storage.BlockManager
   - MESSAGE CONTENT: Found block rdd_2_0 locally

2. Field Extraction
   - Sub-Field: Object
   - Parameter: {1: rdd, 2: 3, 4: 0, 5: 3, 6: 1}

3. Structure Extraction & Mapping
   - EventID: 1, 2, 3, 2, 4
   - ParaID: 1, 2, 3, 2, 5

4. Kernel Compression
   - Mapping Object
   - Mapping Sub-Field Object
   - Parameter Mapping Object
   - Template Mapping Object
### Experiment: Setup

**Evaluation Metric:**

Compression Ratio (CR) \[ CR = \frac{Original\ File\ Size}{Compressed\ File\ Size} \] (The larger the better)

**Dataset:**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Description</th>
<th>Time Span</th>
<th>#Messages</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDFS</td>
<td>HDFS system log</td>
<td>38.7 hours</td>
<td>11,175,629</td>
<td>1.58 GB</td>
</tr>
<tr>
<td>Spark</td>
<td>Spark job log</td>
<td>N.A.</td>
<td>33,236,604</td>
<td>2.75 GB</td>
</tr>
<tr>
<td>Android</td>
<td>Android system log</td>
<td>N.A.</td>
<td>30,348,042</td>
<td>3.62 GB</td>
</tr>
<tr>
<td>Windows</td>
<td>Windows event log</td>
<td>227 days</td>
<td>114,608,388</td>
<td>26.09 GB</td>
</tr>
<tr>
<td>Thunderbird</td>
<td>Supercomputer log</td>
<td>244 days</td>
<td>211,212,192</td>
<td>29.60 GB</td>
</tr>
</tbody>
</table>
## Experiment: Effectiveness of Logzip (level 3)

<table>
<thead>
<tr>
<th>Compression</th>
<th>HDFS</th>
<th>Spark</th>
<th>Android</th>
<th>Windows</th>
<th>Thunderbird</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Size</td>
<td>CR</td>
<td>Size</td>
<td>CR</td>
<td>Size</td>
</tr>
<tr>
<td>Raw</td>
<td>1,618</td>
<td>1</td>
<td>3,011</td>
<td>1</td>
<td>3,707</td>
</tr>
<tr>
<td>Cowic</td>
<td>373.6</td>
<td>4.3</td>
<td>707.4</td>
<td>4.3</td>
<td>1196.7</td>
</tr>
<tr>
<td>LogArchive</td>
<td>114.2</td>
<td>14.2</td>
<td>102.1</td>
<td>29.5</td>
<td>278.7</td>
</tr>
<tr>
<td>gzip</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>improvement</td>
<td>51.7%</td>
<td>2.1x</td>
<td>36.0%</td>
<td>1.6x</td>
<td>47.8%</td>
</tr>
<tr>
<td>logzip (gzip)</td>
<td>149</td>
<td>10.9</td>
<td>175</td>
<td>17.2</td>
<td>439</td>
</tr>
<tr>
<td>bzip2</td>
<td>108</td>
<td>15.0</td>
<td>107</td>
<td>28.1</td>
<td>257</td>
</tr>
<tr>
<td>logzip (bzip2)</td>
<td>63</td>
<td>25.7</td>
<td>85</td>
<td>35.4</td>
<td>145</td>
</tr>
<tr>
<td>improvement</td>
<td>41.7%</td>
<td>1.7x</td>
<td>20.6%</td>
<td>1.3x</td>
<td>43.6%</td>
</tr>
<tr>
<td>lzma</td>
<td>96</td>
<td>16.9</td>
<td>122</td>
<td>24.7</td>
<td>167</td>
</tr>
<tr>
<td>logzip (lzma)</td>
<td>61</td>
<td>26.5</td>
<td>72</td>
<td>41.8</td>
<td>122</td>
</tr>
<tr>
<td>improvement</td>
<td>36.5%</td>
<td>1.6x</td>
<td>41.0%</td>
<td>1.7x</td>
<td>26.9%</td>
</tr>
</tbody>
</table>
CR logzip (gzip) achieved by processing first 1GB data of each datasets
Log data is generally compressed \textbf{once} before a \textbf{long-term} storage.

\textbf{One-off} high computing consumption is acceptable.
Conclusion: Logzip

✓ Iterative clustering for hidden structure extraction
✓ Efficient template matching
✓ Effective and efficient log data compression
LogAdvisor
• Learning to log: A framework for determining optimal logging points
  [ICSE’14, ICSE’15]

Loglizer
• A log analysis toolkit for automated anomaly detection
  [ISSRE’16]

LogParser
• A toolkit for automated log parsing
  [ICSE’19, TDSC'18, DSN’16]

LoggingDescriptions
• A collection of Software Logging Statements
  [ASE’18]

Log3C
• Log-based Problem Identification
  [FSE’18]

Logzip
• An effective and efficient log compression tool
  [ASE’19]
Questions and Cooperation are welcome!

www.logpai.com
info@logpai.com