

---

# Log Anomaly Detection

Aamir Khan, Conrad Yeung

---



# Steps

1. Log Collection
2. Log Parsing
3. Feature Extraction
4. Anomaly Detection

# Log Collection

What logs we will be using:

- Zookeeper (**Zoo**)
- Hadoop Distributed File System (**HDFS**)

<https://github.com/logpai/loghub>

TABLE II: Parsing Accuracy of Log Parsing Methods (Raw/Preprocessed)

	BGL	HPC	HDFS	Zookeeper	Proxifier
SLCT	<b>0.61/0.94</b>	0.81/0.86	0.86/0.93	0.92/0.92	0.89/-
IPLoM	0.99/0.99	0.64/0.64	0.99/1.00	0.94/0.90	0.90/-
LKE	0.67/0.70	0.17/0.17	<b>0.57/0.96</b>	0.78/0.82	0.81/-
LogSig	<b>0.26/0.98</b>	0.77/0.87	0.91/0.93	0.96/0.99	0.84/-

*An Evaluation Study on Log Parsing and Its Use in Log Mining, He et al.*  
<https://pinjiahe.github.io/papers/DSN16.pdf>

# Log Parsing

What parsing method will we use:

- Iterative Partitioning Log Mining (IPLoM)

Reason:

- Deterministic
- High parsing accuracy
- No need for pre-processing
- One optional parameter

<https://github.com/logpai/logparser>

```
081109 203815 148 INFO dfs.DataNode$PacketResponder: PacketResponder 1 for block blk_3885049064139660 terminating
081109 203807 222 INFO dfs.DataNode$PacketResponder: PacketResponder 0 for block blk_-695229586487656571 terminating
081109 204005 35 INFO dfs.FSNamesystem: BLOCK* NameSystem.addStoredBlock: blockMap updated: 10.251.73.220:50010 is added to blk_7128370237687728475
size 67108864
081109 204015 308 INFO dfs.DataNode$PacketResponder: PacketResponder 2 for block blk_8229193803249955061 terminating
081109 204106 329 INFO dfs.DataNode$PacketResponder: PacketResponder 2 for block blk_-667095862368987959 terminating
081109 204132 26 INFO dfs.FSNamesystem: BLOCK* NameSystem.addStoredBlock: blockMap updated: 10.251.43.115:50010 is added to blk_3050920587428079149
size 67108864
081109 204134 34 INFO dfs.FSNamesystem: BLOCK* NameSystem.addStoredBlock: blockMap updated: 10.251.203.80:50010 is added to blk_7888946331804732825
size 67108864
081109 204453 34 INFO dfs.FSNamesystem: BLOCK* NameSystem.addStoredBlock: blockMap updated: 10.250.11.85:50010 is added to blk_2377150260128098806
size 67108864
081109 204525 512 INFO dfs.DataNode$PacketResponder: PacketResponder 2 for block blk_572492839287299681 terminating
081109 204655 556 INFO dfs.DataNode$PacketResponder: Received block blk_3587080140851953248 of size 67108864 from /10.251.42.84
081109 204722 567 INFO dfs.DataNode$PacketResponder: Received block blk_5402801568134525440 of size 67108864 from /10.251.214.112
081109 204815 653 INFO dfs.DataNode$DataReceiver: Receiving block blk_579248980879196128 src: /10.251.30.6:3345 dest: /10.251.30.6:50010
081109 204842 663 INFO dfs.DataNode$DataReceiver: Receiving block blk_372475784874353110 src: /10.251.111.130:49851 dest: /10.251.111.130:50010
081109 204908 11 INFO dfs.FSNamesystem: BLOCK* NameSystem.addStoredBlock: blockMap updated: 10.251.110.8:50010 is added to blk_801591322471308318
size 67108864
```



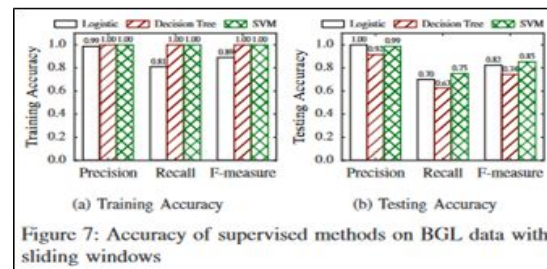
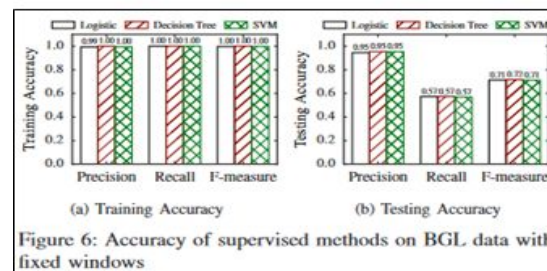
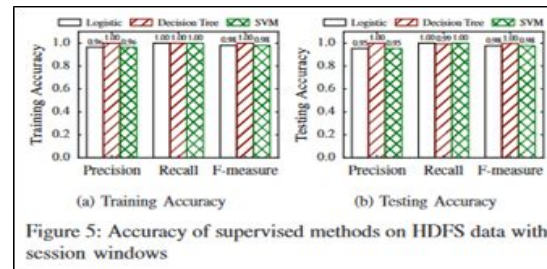
Unqid	Date	Time	Pid	Level	Component	Content	EventId	EventTemplate	ParameterList
1	81109	203815	148	INFO	dfs.DataNode\$PacketRes	dc2c74b7	PacketRes[1]	'blk_3885049064139660'	
2	81109	203807	222	INFO	dfs.DataNode\$PacketRes	dc2c74b7	PacketRes[0]	'blk_-695229586487656571'	
3	81109	204005	35	INFO	dfs.FSNameBLOC	* N Seaa2a11	BLOCK* N	'10.251.73.220', 'blk_7128370237687728475', '67108864'	
4	81109	204015	308	INFO	dfs.DataNode\$PacketRes	dc2c74b7	PacketRes[2]	'blk_8229193803249955061'	
5	81109	204106	329	INFO	dfs.DataNode\$PacketRes	dc2c74b7	PacketRes[2]	'blk_-667095862368987959'	
6	81109	204132	26	INFO	dfs.FSNameBLOC	* N Seaa2a11	BLOCK* N	'10.251.43.115', 'blk_3050920587428079149', '67108864'	
7	81109	204324	34	INFO	dfs.FSNameBLOC	* N Seaa2a11	BLOCK* N	'10.251.203.80', 'blk_7888946331804732825', '67108864'	
8	81109	204453	34	INFO	dfs.FSNameBLOC	* N Seaa2a11	BLOCK* N	'10.250.11.85', 'blk_2377150260128098806', '67108864'	
9	81109	204525	512	INFO	dfs.DataNode\$PacketRes	dc2c74b7	PacketRes[2]	'blk_572492839287299681'	



EventId	EventTemplate	Occurrences	Accuracy	99.85%
dba996ef	Deleting block <*> file <*>	263		
dc2c74b7	PacketResponder <*> for block <*> terminating	311		
e3df2680	Received block <*> of size <*> from <*>	292		
b2cd0462	BLOCK* NameSystem.delete <*> is added to invalidSet of <*>	224		
46769d4f	BLOCK* ask 10.251.128.3:50010 to delete blk_-901656746707	1		
5d8c5df5	BLOCK* NameSystem.allocateBlock <*> <*>	115		
32777b38	Verification succeeded for <*>	20		
3a7f0f8e	<*> 50010 Served block <*> to <*>	80		
b4ec8d10	BLOCK* ask <*> 50010 to delete <*>	2		
6af214df	Receiving block <*> src <*> <*> dest <*> 50010	292		
45b212df	<*> 50010 Got exception while serving <*> to <*>	80		
dd4fd19c	<*> <*> <*> to <*> <*> <*> <*> 50010	2		
5eaa2a11	BLOCK* NameSystem.addStoredBlock blockMap updated <*>	314		
44378071	Received block blk_-4411589101766561890 src /10.250.14.38	1		
634b67e0	Received block blk_1473848624870719319 src /10.251.29.239	1		
a215b0b5	BLOCK* ask <*> 50010 to delete <*> <*> <*> <*> <*> <*> <*>	2		

# Feature Extraction

- Fixed Window: Zoo & HDFS
- Sliding Window: Zoo & HDFS
- Session Window: **HDFS** (blk\_id)



# Model Selection

- Several possible directions
- Supervised learning
  - Use “anomaly\_label.csv” to add labels column
  - Possible models to explore: Logistic regression or Random forest
- Unsupervised learning
  - Use the log data directly
  - Possibly use PCA to find anomalous patterns.
  - K-means clustering to gather alike events.

# Model Selection

- Use *normal distribution* to detect anomaly
- The more an event occurs away from the distribution curve, more chances that it is an anomaly
- We can add level of certainty to our decisions.

# References

- [https://netman.aiops.org/~peidan/ANM2018Fall/6.LogAnomalyDetection/LectureCoverage/2016ISSRE System%20Log%20Analysis%20for%20Anomaly%20Detection.pdf](https://netman.aiops.org/~peidan/ANM2018Fall/6.LogAnomalyDetection/LectureCoverage/2016ISSRE%20System%20Log%20Analysis%20for%20Anomaly%20Detection.pdf)
- <https://pinjiahe.github.io/papers/DSN16.pdf>
- <https://www.kaggle.com/shelars1985/anomaly-detection-using-gaussian-distribution>