Automatic Classification of System Logs

Siavash Ghiasvand[§] and Florina M. Ciorba[★]

§Technische Universität Dresden, Germany ★University of Basel, Switzerland

1

```
starting acron
starting monitor
Acron started on 2018-03-01
Jobs will be executed sequentially
Normal exit (3 jobs run)
finished acron
(root) CMD (run-parts)
```

Sample system log entries

Current state of practice

- Log entry messages are unstructured[1]
- Most of log entry messages are repetitive (high frequency)[2]
- Manual analysis is not efficient for large scale computing systems[4]
- Automatic classification requires pre-classified/labeled sample logs
- Available methods are system-specific[5]

Goal

- Automatic classification of system logs

Contributions

- The approach does not require data pre-classification and labeling
- General approach

2

Proposed classification approach

Marking common terms in system log entries
(marking terms as invariant)

- Functions
- Parentheses and quotations
- File/application addresses
- Dates, Hours, IP addresses
- Names, Numbers

Classifying log entries into classes, based on Levenshtein[3] similarity metric

Harmonizing differences across entries of the same class

Refining and adjusting the resulting regular expressions

Live demo and sample script:
ghiasvand.net/u/isc18

Sample system log entries

```
(sshd:session): session closed for root
Job `cron.daily' terminated 96734
Job `cron.weakly' terminated 537352
(sshd:session): session closed for siavash
Job `cron.hourly' terminated 241325
(sshd:session): session closed for florina
(sshd:session): session closed for s125342
Job `cron.daily' terminated 14038
```

CLASS 1
```
#PARA#: session closed for root
#PARA#: session closed for siavash
#PARA#: session closed for florina
#PARA#: session closed for s125342
```

CLASS 2
```
Job `cron.hourly' terminated #DGIT#
Job `cron.daily' terminated #DGIT#
Job `cron.weakly' terminated #DGIT#
Job `cron.daily' terminated #DGIT#
```

CLASS 1
```
#PARA#: session closed for #VARI#
```

CLASS 2
```
Job #VARI# terminated #DGIT#
```

RegEx 1
```
(\(.+?\))\: #: session closed for (.+)
```

RegEx 2
```
Job (.+) terminated ([0-9]+)
```

Conclusion

- Automatic classification of system logs is possible
- Low frequency log entries reduce the overall classification accuracy
- No relation between number of log entries and number of automatically generated classes
- Strong relation between log frequency and classifiers accuracy

Future work

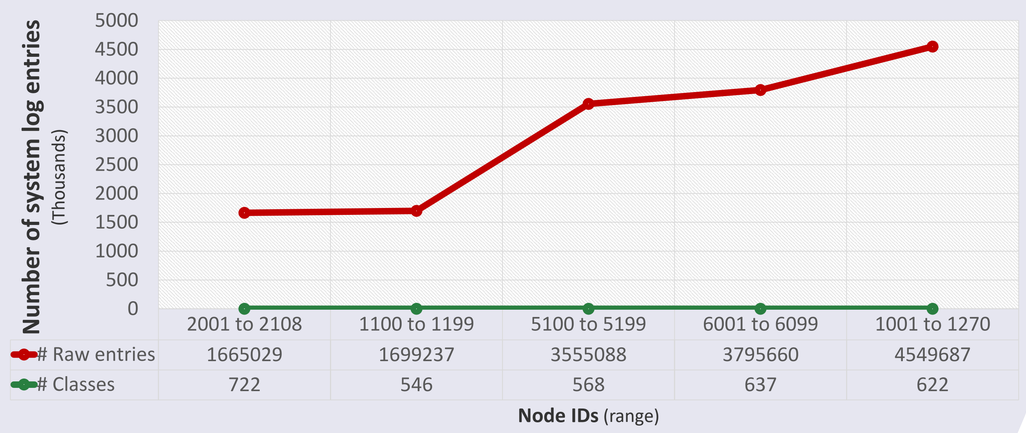
- Addressing entries with two terms and syslog entries that form blocks

4

Evaluation and preliminary results

	Datasets		Classes
	1 month of system logs		
1.	108 nodes (	1,665,029 entries)	722
2.	100 nodes (	1,699,237 entries)	546
3.	100 nodes (	3,555,088 entries)	568
4.	99 nodes (	3,795,660 entries)	637
5.	270 nodes (	4,549,687 entries)	622
6.	677 nodes (	15,264,701 entries)	860
	1 year of system logs		
7.	99 nodes	(136,609,978 entries)	1090

Automatic Classification



3

Observations

- Number of generated classes remains almost identical
- Heterogeneity of nodes has no significance on the number of classes
- High frequency log entries improve classification
- 41% of classes are common in all datasets
- 63% of classes are common in several sets

References

[1] "The syslog protocol," http://tools.ietf.org/html/rfc5424, [Online; accessed March 2018]
[2] S. Ghiasvand and F. M. Ciorba, "Anonymization of system logs for preserving privacy and reducing storage," in *Proc. of the Future of Information and Communications Conference*, 2018
[3] Levenshtein, VI. "Binary Codes Capable of Correcting Deletions, Insertions and Reversals." *Soviet Physics Doklady* 10 (1966): 707.
[4] T. Li et. al. "FLAP: An End-to-End Event Log Analysis Platform for System Management". In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '17). ACM, New York, NY, USA, 2017
[5] K. Berkay "Unsupervised Anomaly Detection in Unstructured Log-Data for Root-Cause-Analysis", master thesis, Tampere University of Technology, Finland, 2015

Acknowledgment

This work is in part supported by the German Research Foundation (DFG) within the Cluster of Excellence 'Center for Advancing Electronics Dresden (cfaed)', and by Eucor - The European Campus, within 'Data Analysis for Improving High Performance Computing Operations and Research'.

TECHNISCHE UNIVERSITÄT DRESDEN

ZIH Center for Information Services & High Performance Computing

cfaed

Universität Basel