> Syslog Analysis

Researcher: Andrei Batyrov, (C) 2021-2024

Table of Contents

1. Description

Customer network devices collect piles of system messages, collectively referred to as "syslog". Some messages are well documented and rather easily identifiable, while many others either do not have meaningful description or occur not frequently but still may indicate degradation of service. *The goal* is to cluster the messages into meaningful groups. Messages falling in the same group ideally should have similar semantics.

Challenges:

- large amount of data,
- text/csv format,
- customer wants to convert raw text data into meaningful actionable items.

Suggested approach:

- use Apache Spark Resilient Distributed Dataset, DataFrame, and SQL to work with big data,
- use Natural Language Processing techniques to extract features from the messages,
- use clustering, embedding, and dimensionality reduction techniques to group messages,
- assess and extract new information from the obtained groups.

2. Load and Parse Data

Raw message count and sample 5 messages. Right now it looks as chaotic.

Out[ ]: 22632

Out[ ]:
```
['CCM-TFTP-MSK;;;2019-1-15 23:45:04;;3;%UC_AUDITLOG-3-AdministrativeEvent: %[ UserID =avsamok3][ ClientAddress =10.73.222.40][ Severity =3][ EventType =UserLogging][ ResourceAccessed=cucm-uds][ EventStatus =Failure][ CompulsoryEvent =No][ AuditCategory =AdministrativeEvent][ ComponentID =Cisco CCM Application][ AuditDetails =Login Authentication Failed][App ID=Cisco Tomcat][Cluster ID=][Node ID=CCM-TFTP-MSK]: Audit Event is generated by this application ;540',
 'ccm-pub-msk;;;2019-1-15 23:45:05;;3;%UC_AUDITLOG-3-AdministrativeEvent: %[ UserID =CCM_TMS][ ClientAddress =10.73.222.65][ Severity =3][ EventType =UserLogging][ ResourceAccessed=Soap Real Time Service 2][ EventStatus =Success][ CompulsoryEvent =No][ AuditCategory =AdministrativeEvent][ ComponentID =Cisco CCM Application][ AuditDetails =Login Authentication Successful][App ID=Cisco Tomcat][Cluster ID=][Node ID=ccm-pub-msk]: Audit Event is generated by this application ;65602',
 'ccm-sub3-msk;;;2019-1-15 23:45:06;;3;%UC_CALLMANAGER-3-EndPointTransientConnection: %[Connecting Port=5060][Device name=SEP00156286A44D][Device IP address=10.34.232.229][Device type=30007][Reason Code=7][Protocol=SIP][Device MAC address=00156286A44D][IPAddressAttributes=0][LastSignalReceived=SIPRegisterInd][StationState=wait_register][App ID=Cisco CallManager][Cluster ID=CCM1-Cluster][Node ID=ccm-sub3-msk]: An endpoint attempted to register but did not complete registration;224461',
 'ccm-pub-msk;;;2019-1-15 23:45:10;;3;%UC_AUDITLOG-3-AdministrativeEvent: %[ UserID =AurusPhoneUP_MRP][ ClientAddress =10.34.5.13][ Severity =3][ EventType =UserLogging][ ResourceAccessed=Apache-Axis2][ EventStatus =Success][ CompulsoryEvent =No][ AuditCategory =AdministrativeEvent][ ComponentID =Cisco CCM Application][ AuditDetails =Login Authentication Successful][App ID=Cisco Tomcat][Cluster ID=][Node ID=ccm-pub-msk]: Audit Event is generated by this application ;21806',
 'ccm-sub6-msk;;;2019-1-15 23:45:14;;3;%UC_CALLMANAGER-3-EndPointTransientConnection: %[Connecting Port=5060][Device name=][Device type=336][Reason Code=20][Protocol=SIP][LastSignalReceived=SIPRegisterInd][StationState=wait_register][App ID=Cisco CallManager][Cluster ID=CCM1-Cluster][Node ID=ccm-sub6-msk]: An endpoint attempted to register but did not complete registration;18620']
```

Parsed messages already look much more meaningful.

```
+--------------------+--------------------+---------+-------------+------------+--
------+----------------+--------------------+--------------------+----------+
|           date_time|                date|time_hour|           ip|    hostname|se
verity|          userid|              header|                body|date_human|
+--------------------+--------------------+---------+-------------+------------+--
------+----------------+--------------------+--------------------+----------+
|java.util.Gregori...|java.util.Gregori...|       23| 10.73.222.40|CCM-TFTP-MSK|
3|        avsamok3|%UC_AUDITLOG-3-Ad...|Audit Event is ge...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23| 10.73.222.65| ccm-pub-msk|
3|         CCM_TMS|%UC_AUDITLOG-3-Ad...|Audit Event is ge...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23|         5060|ccm-sub3-msk|
3|            NULL|%UC_CALLMANAGER-3...|An endpoint attem...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23|   10.34.5.13| ccm-pub-msk|
3|AurusPhoneUP_MRP|%UC_AUDITLOG-3-Ad...|Audit Event is ge...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23|         5060|ccm-sub6-msk|
3|            NULL|%UC_CALLMANAGER-3...|An endpoint attem...| 2019-1-15|
|                    |                    |         |             |            |
|                |                    |                     |      NULL|
|java.util.Gregori...|java.util.Gregori...|       23|         5060|ccm-sub6-msk|
3|            NULL|%UC_CALLMANAGER-3...|An endpoint attem...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23| 10.73.222.65|CCM-TFTP-MSK|
3|         CCM_TMS|%UC_AUDITLOG-3-Ad...|Audit Event is ge...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23|10.73.253.143| ccm-pub-msk|
3|          tiukin|%UC_AUDITLOG-3-Ad...|Audit Event is ge...| 2019-1-15|
|java.util.Gregori...|java.util.Gregori...|       23|            1|ccm-sub4-msk|
3|            NULL|%UC_-3-LastOutOfS...| Information rela...| 2019-1-15|
+--------------------+--------------------+---------+-------------+------------+--
------+----------------+--------------------+--------------------+----------+
only showing top 10 rows
```

3. Basic Apache Spark Transformations and Actions

Group messages by header

```
+-----------------------------------------------------------+-----+
|header                                                     |count|
+-----------------------------------------------------------+-----+
|%UC_-3-LastOutOfServiceInformation                         |10076|
|%UC_CALLMANAGER-3-EndPointUnregistered                     |5332 |
|%UC_CALLMANAGER-3-RecordingCallSetupFail                   |3727 |
|%UC_AUDITLOG-3-AdministrativeEvent                         |2093 |
|%UC_-3-DeviceTLInfo                                        |609  |
|%UC_CALLMANAGER-3-EndPointTransientConnection              |330  |
|%UC_CTI-3-CtiDeviceOpenFailure                             |29   |
|%UC_-3-DeviceApplyConfigResult                             |27   |
|%UC_CALLMANAGER-3-RecordingSessionTerminatedUnexpectedly|23   |
|%UC_-3-DeviceImageDownloadStart                            |21   |
|%UC_-3-DeviceImageDownloadSuccess                          |19   |
|%UC_CALLMANAGER-3-DeviceUnregistered                       |11   |
|%UC_CALLMANAGER-3-SIPNormalizationResourceWarning          |10   |
|%UC_CALLMANAGER-3-SIPTrunkOOS                              |8    |
|%UC_CALLMANAGER-3-DeviceTransientConnection                |6    |
|%UC_CALLMANAGER-1-DeviceUnregistered                       |4    |
|%UC_-3-DeviceImageDownloadFailure                          |3    |
|%UC_Proxy-3-UASCBFindFailed                                |3    |
|%UC_CALLMANAGER-2-DChannelOOS                              |2    |
|%UC_CTI-3-CtiLineOpenFailure                               |1    |
+-----------------------------------------------------------+-----+
only showing top 20 rows
```

Aggregation: get mean message severity

```
+------------------+
|     avg(severity)|
+------------------+
|2.9995523123069345|
+------------------+
```

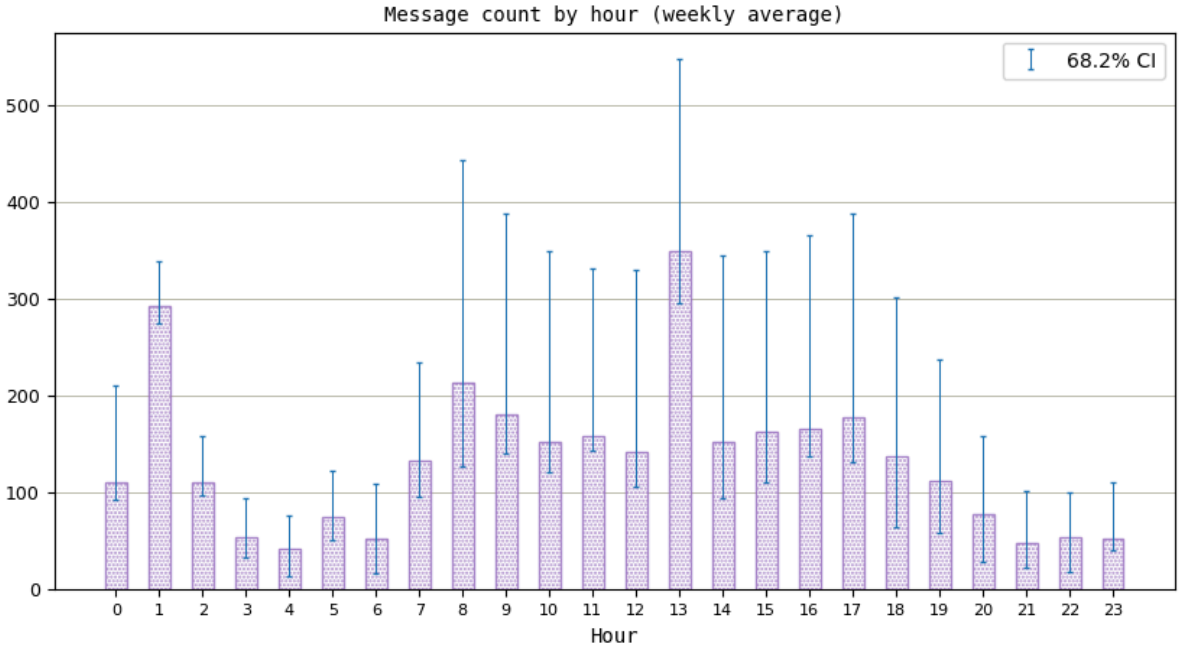
Count unique ip addresses

```
+------------------+
|count(DISTINCT ip)|
+------------------+
|              3876|
+------------------+
```

Count night message headers

```
+--------------------+--------+
|              header|count(1)|
+--------------------+--------+
|%UC_-3-LastOutOfS...|    3058|
|%UC_CALLMANAGER-3...|    2438|
|%UC_AUDITLOG-3-Ad...|    1239|
| %UC_-3-DeviceTLInfo|     260|
|%UC_CALLMANAGER-3...|     241|
|%UC_CALLMANAGER-3...|     209|
|%UC_-3-DeviceAppl...|      18|
|%UC_CTI-3-CtiDevi...|      16|
|%UC_-3-DeviceImag...|      14|
|%UC_-3-DeviceImag...|      14|
+--------------------+--------+
only showing top 10 rows
```

4. Message Distribution over Time



We can see message spikes at `1am` and `1pm`. Also, the one-sigma quantile intervals (68.2% of all messages) signal fat tails of message count distributions virtually for every hour. Let's show the top 10 hostnames and message headers for these hours.

1am

```
+----------+------------+--------------------------------------------+--------+
|date_human|hostname    |header                                      |count(1)|
+----------+------------+--------------------------------------------+--------+
|2019-1-16 |ccm-sub4-msk|%UC_CALLMANAGER-3-EndPointUnregistered      |1015    |
|2019-1-16 |ccm-sub4-msk|%UC_-3-LastOutOfServiceInformation          |210     |
|2019-1-16 |ccm-sub6-msk|%UC_-3-LastOutOfServiceInformation          |204     |
|2019-1-16 |ccm-pub-msk |%UC_AUDITLOG-3-AdministrativeEvent          |108     |
|2019-1-16 |ccm-sub6-msk|%UC_CALLMANAGER-3-EndPointTransientConnection|73      |
|2019-1-16 |ccm-sub4-msk|%UC_-3-DeviceTLInfo                         |68      |
|2019-1-16 |ccm-sub6-msk|%UC_CALLMANAGER-3-EndPointUnregistered      |61      |
|2019-1-16 |ccm-sub3-msk|%UC_CALLMANAGER-3-EndPointUnregistered      |54      |
|2019-1-20 |ccm-sub3-msk|%UC_-3-LastOutOfServiceInformation          |23      |
|2019-1-19 |ccm-sub4-msk|%UC_-3-LastOutOfServiceInformation          |22      |
+----------+------------+--------------------------------------------+--------+
```

1pm

```
+----------+------------+---------------------------------------+--------+
|date_human|hostname    |header                                 |count(1)|
+----------+------------+---------------------------------------+--------+
|2019-1-16 |ccm-sub4-msk|%UC_CALLMANAGER-3-EndPointUnregistered |944     |
|2019-1-16 |ccm-sub4-msk|%UC_-3-LastOutOfServiceInformation     |287     |
|2019-1-16 |ccm-sub3-msk|%UC_-3-LastOutOfServiceInformation     |91      |
|2019-1-16 |ccm-sub4-msk|%UC_-3-DeviceTLInfo                    |86      |
|2019-1-17 |ccm-sub3-msk|%UC_-3-LastOutOfServiceInformation     |65      |
|2019-1-18 |ccm-sub3-msk|%UC_-3-LastOutOfServiceInformation     |55      |
|2019-1-15 |ccm-sub3-msk|%UC_-3-LastOutOfServiceInformation     |48      |
|2019-1-17 |ccm-sub3-msk|%UC_CALLMANAGER-3-RecordingCallSetupFail|40      |
|2019-1-15 |ccm-sub4-msk|%UC_-3-LastOutOfServiceInformation     |39      |
|2019-1-16 |ccm-sub3-msk|%UC_CALLMANAGER-3-RecordingCallSetupFail|35      |
+----------+------------+---------------------------------------+--------+
```

We can quickly spot the top "offending" hostnames and messages.

5. Message Embeddings

First, let's encode message bodies as token (word) features with Tf-Idf. We might also use Word2Vec or even BERT models for feature extraction to get even better results.

Tokenize message bodies with our tokenizer

```
+------------------------------------------------------------------+-------------
-------------------------------------------------------------+
|body                                                                |tokenized_bod
y                                                            |
+------------------------------------------------------------------+-------------
-------------------------------------------------------------+
|Audit Event is generated by this application                        |[audit, even
t, is, generated, by, this, application]                          |
|Audit Event is generated by this application                        |[audit, even
t, is, generated, by, this, application]                          |
|An endpoint attempted to register but did not complete registration|[an, endpoin
t, attempted, to, register, but, did, not, complete, registration]|
|Audit Event is generated by this application                        |[audit, even
t, is, generated, by, this, application]                          |
|An endpoint attempted to register but did not complete registration|[an, endpoin
t, attempted, to, register, but, did, not, complete, registration]|
+------------------------------------------------------------------+-------------
-------------------------------------------------------------+
only showing top 5 rows
```

Compute Tf-Idf values for each token (word)

```
['323' 'a' 'action' 'administration' 'all']
[0.40101003 0.40055591 0.40101003 0.40101003 0.39959662]
N of extracted features (words): 104
```

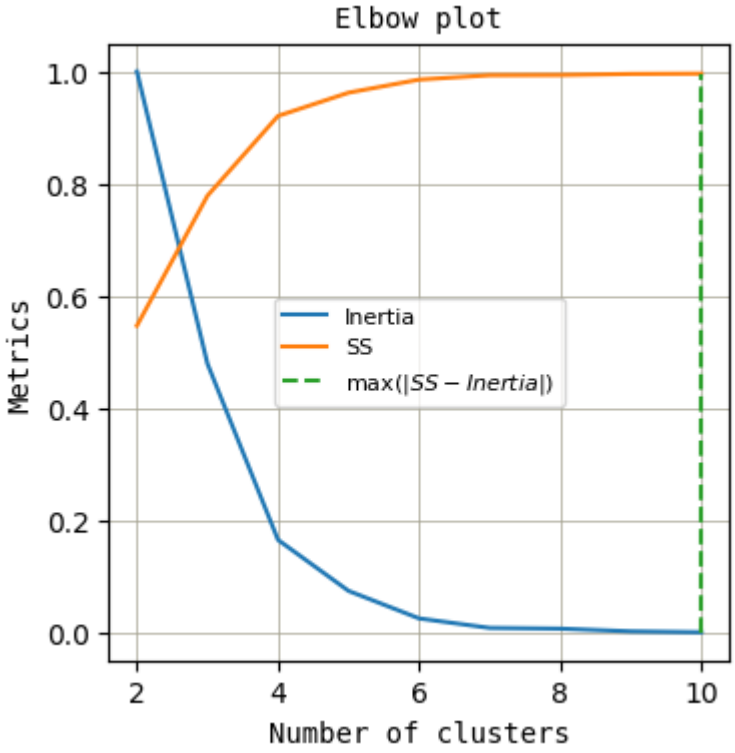
Now, for every message we have introduced 104 new features -- unique tokens (words) extracted from message bodies. That is, every message body is now embedded into $\mathbb{R}^{104}$ feature space.

Next, we will group all messages into a reasonable number of clusters, say under 10, using the well-known K-Means algorithm. To assess the quality of clustering we'll look at two metrics:

- Inertia (within-cluster sum-of-squares). Inertia decreases as the number of clusters increases. The optimal value is 0.0;
- Silhouette Score (SS) (mean intra-cluster distance). SS may increase/decreases as the number of clusters increases, but the optimal value is 1.0 here.

The idea is to find the number of clusters such that Inertia is minimized while SS is maximized at the same time: $k_{opt} = \underset{k}{argmax}(|SS - Inertia|)$.

```
Optimal number of clusters: 10
```



Finally, since it is unimaginable how to assess 104-dimension data, let's embed our features into a much smaller dimension space. We may want to "reduce" the dimensionality to at least three or better two coordinates. To do that, let's use the well-known Stochastic Neighbor Embedding with t-distribution algorithm. We will display message headers for the embedded points. Each such header will be colored as per the message's cluster number. So, ideally we should see similar messages, in terms of their bodies "meanings", clustered together and having the same color. On the contrary, message having non-overlapping "meanings" should be spread apart and have different colors.

> Important note: The whole idea of extracting and comparing "meanings" with NLP models is based on the the so-called distributional semantics *hypothesis*: linguistic items with similar distributions have similar meanings. So, care should be taken when assessing results obtained using purely distributional (mathematical) approaches to work with word and text

> meanings in a broader sense. Decision-making is totally at the researcher's discretion.

Now each message has only 2 new features (coordinates), instead of 104 which we had initially after tf-idf encoding.

Out[ ]:
```
array([[-18.197277 ,  -8.686308 ],
       [-18.197092 ,  -8.686219 ],
       [ 27.51425  ,  12.208876 ],
       ...,
       [  5.6797004, -13.398742 ],
       [  4.5499406,  -8.912752 ],
       [  2.85576  , -10.157701 ]], dtype=float32)
```

Number of unique headers

```
+----------------------+
|count(DISTINCT header)|
+----------------------+
|                    23|
+----------------------+
```

Out[ ]:

Unique message headers and their clusters

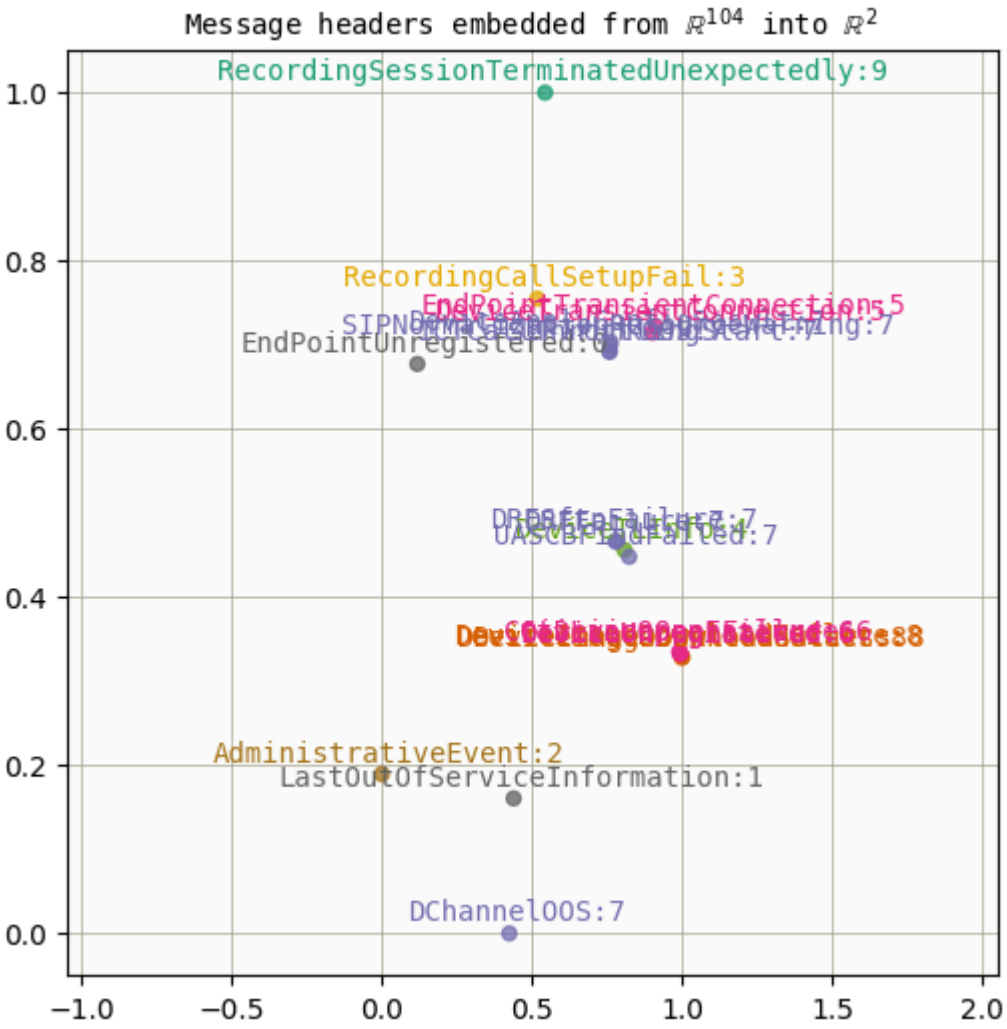
	header_short	cluster
0	EndPointUnregistered	0
1	LastOutOfServiceInformation	1
2	AdministrativeEvent	2
3	RecordingCallSetupFail	3
4	DeviceTLInfo	4
5	EndPointTransientConnection	5
6	DeviceTransientConnection	5
7	CtiLineOpenFailure	6
8	CtiDeviceOpenFailure	6
9	DeviceUnregistered	6
10	DeviceUnregistered	6
11	DRFFailure	7
12	SIPTrunkOOS	7
13	SIPNormalizationResourceWarning	7
14	DeviceApplyConfigResult	7
15	DRFSftpFailure	7
16	DChannelOOS	7
17	ICTCallThrottlingStart	7
18	UASCBFindFailed	7
19	DeviceImageDownloadSuccess	8
20	DeviceImageDownloadStart	8
21	DeviceImageDownloadFailure	8
22	RecordingSessionTerminatedUnexpectedly	9

Cluster 7 is the most populous.

6. Plot Results

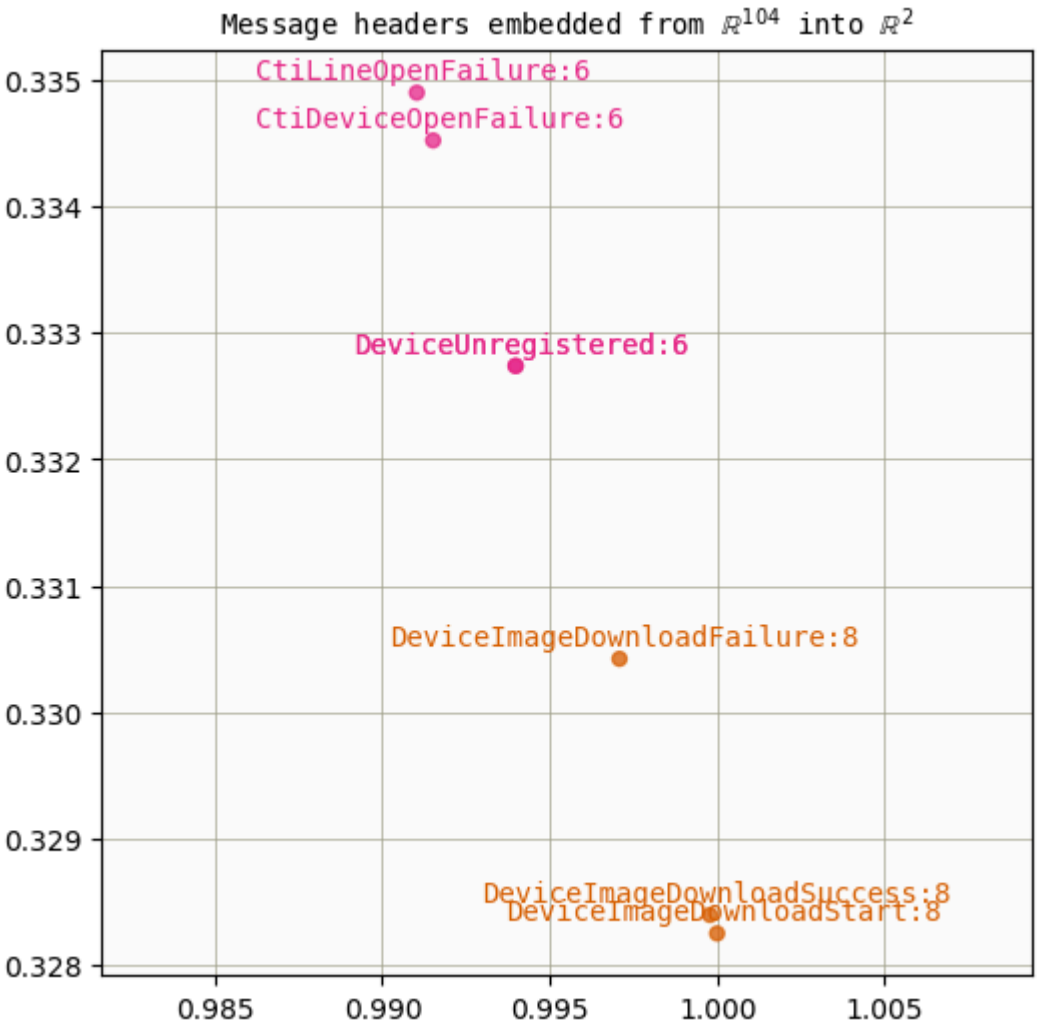
Now it's time to visualize our results.

All headers.



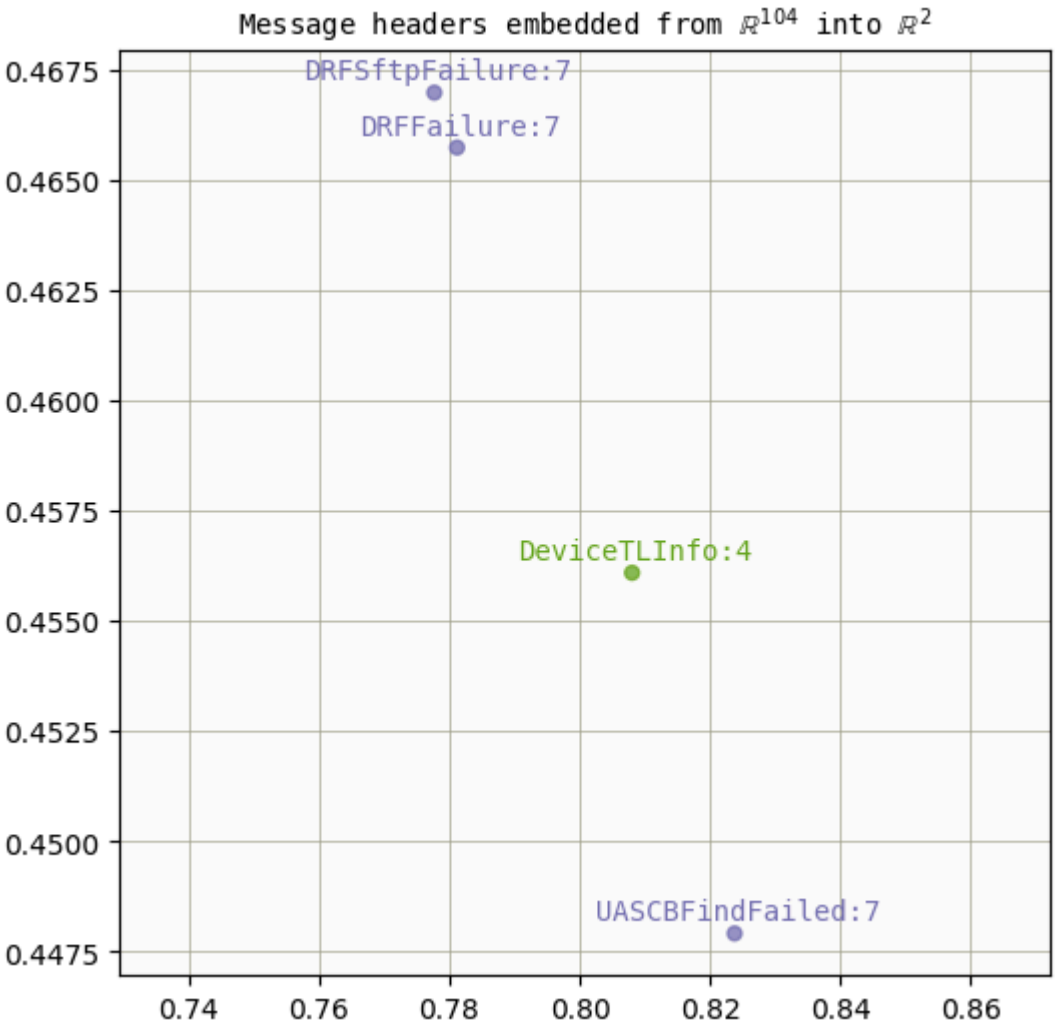
Our top "offender" message (cluster 0) looks to be rather distant from other messages indeed, which means that this issue should not be related to other events and thus should be examined further individually.

t-SNE put clusters 6 and 8 close to each other. Let's take a closer look at them.



Looking at the headers, groupings seem to be reasonable.

Next, we can see that the cluster 4 and several messages in the cluster 7 sort of overlap as suggested by t-SNE. Again, looking at them closer.



Interestingly, cluster 7 is spread out in fact, as embedded by t-SNE.

Also, we can see that the cluster 5 and several messages in the cluster 7 lie closely together as suggested by t-SNE. Zooming in.

Looking at the headers, groupings seem to be reasonable.

7. Summary

Overall, we were able to achieve our goal:

- syslog messages were divided into meaningful groups based on text analysis of message bodies,
- in general, K-Means clustering and t-SNE reduction converged to the same groups,
- there are also some divergences:
  - clusters 6 and 8 are believed to be closely embedded by t-SNE,
  - clusters 4 is believed to be closely embedded within cluster 7 by t-SNE,
  - cluster 7 is actually spread out into distant points after t-SNE reduction.

These results can be communicated to the customer.