Progression overview

May 6, 2018

This is a progression overview of Outlier Factors for Device Profiling. All coding examples can be found at Github (not very tidy an the moment).

We will focus with the following dataset Los Alamos National Laboratorys corporate, internal computer network. A artificial generated dataset could be used as well, but the nature of the data should be closely taken into account.

1 Distribution

The data used comprises of a collection of flows created at different timestamps from different users. In a first attempt we will only consider the number of flows and the bytes sent through these flows. We will also temporarily ignore time relations and drifts in the data.

The beginning of this dataset can be displayed in 1. As we can see most points denote a zero traffic for users in a given time.

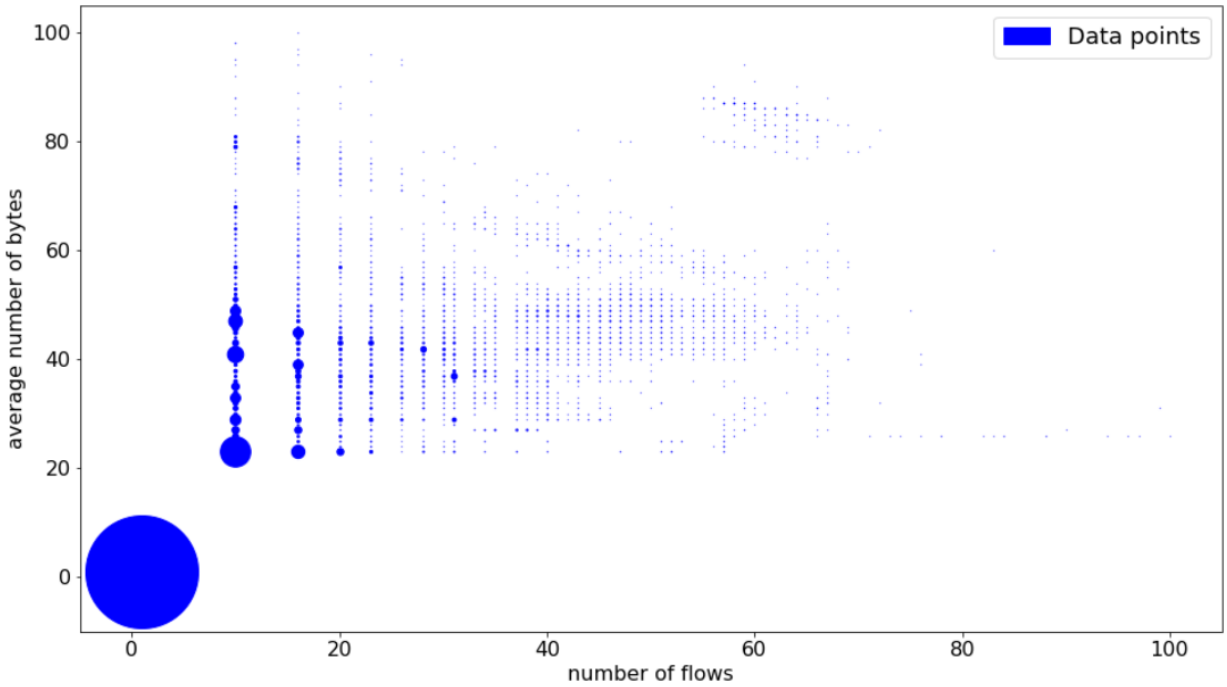

Figure 1: Distribution (the size of each dot denotes the number of points with the same value)

The main interest is how the behavior of a host at a given time (epoch) compare to the host past behavior and the time field.

The fact that most of the hosts have a zero traffic most of the time can be illustrated in 2, where we can see the average traffic for each individual host.

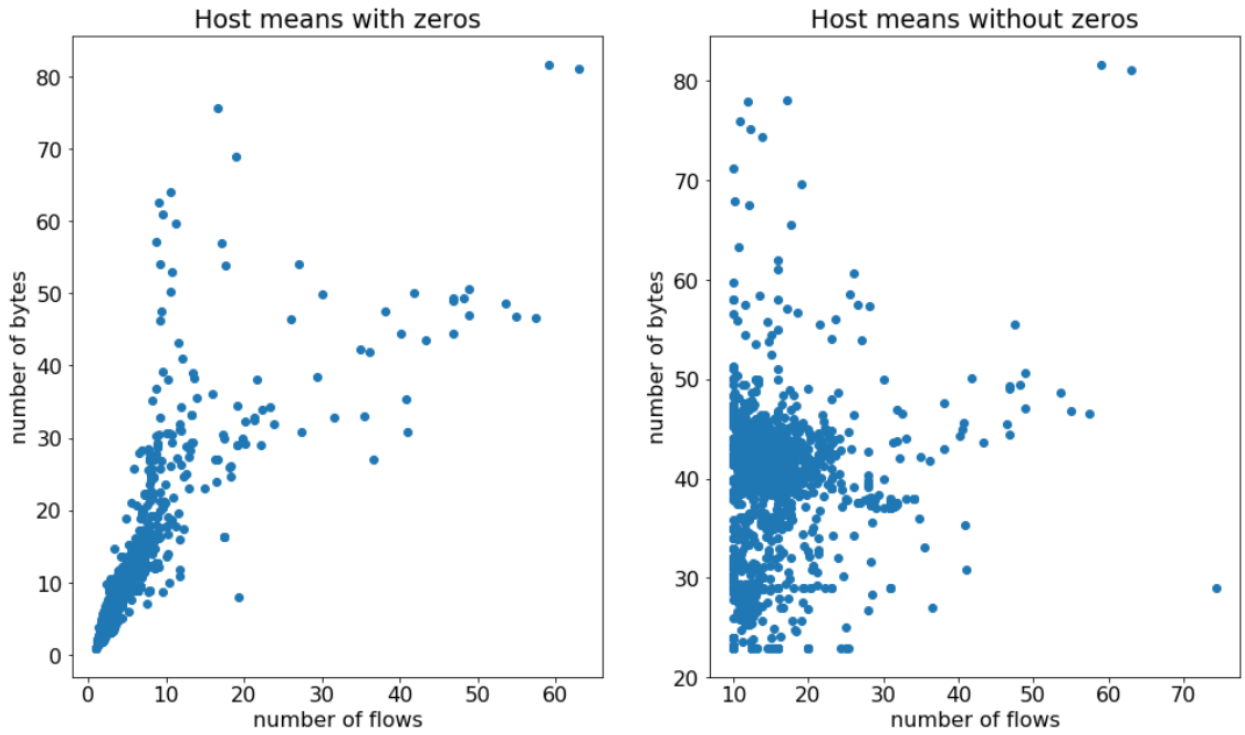

Figure 2: Average traffic for each individual host.

In the figure 3 we can see the average traffic for each individual epoch.

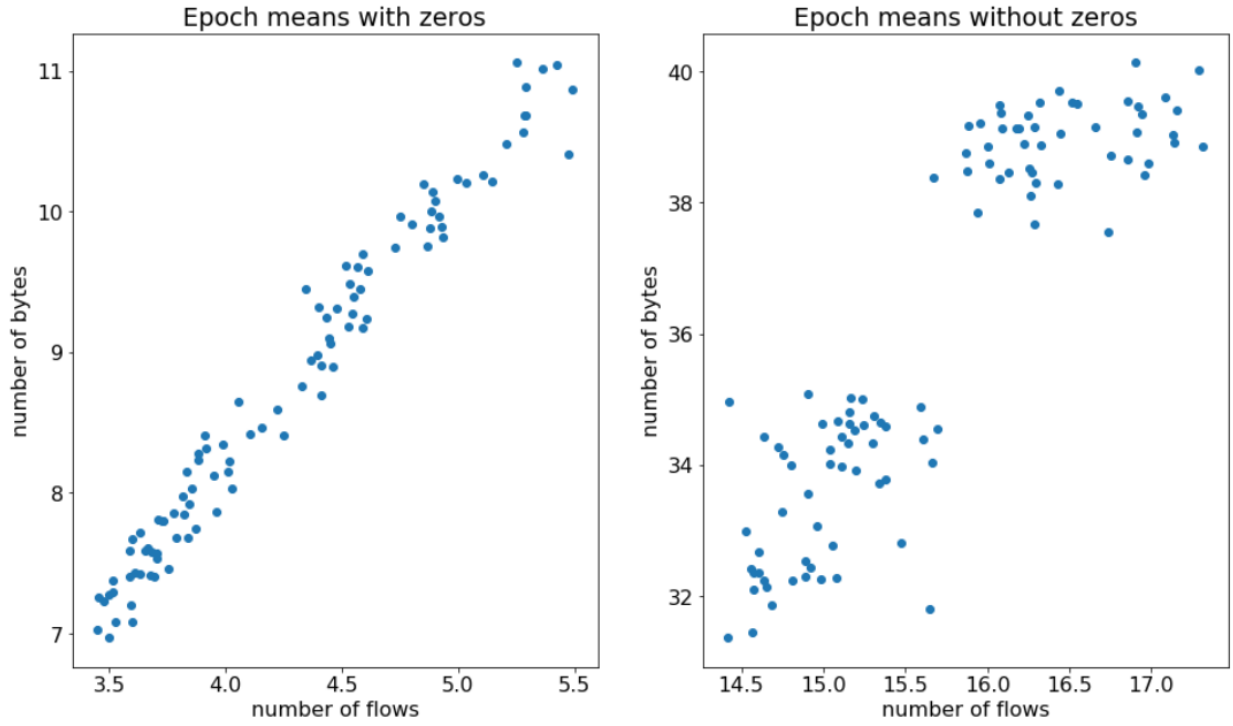

Figure 3: Average traffic for each epoch.

	Mean Square Error
Each point compared to the mean of the host's past	201.481193
New mean for host compared to the mean of the host's past	5.57316684

Table 1: Mean Square error comparing to each host individually

	Percentage of non zero traffic	Mean Square Error
Individual points	0	201.481193
	10	231.04849494
	20	242.53839645
	30	216.13443677
	40	240.51540553
	50	228.43204413
	60	213.30825695
	70	187.704809320
	80	153.79107752
	90	99.99244057
Average of points	0	5.57316684
	10	4.2962690363
	20	4.3410310789
	30	4.387796819
	40	2.1611752767
	50	1.9575657276
	60	2.4987264900
	70	2.9684499999
	80	2.427106741
	90	2.4334405797

Table 2: Mean Square error comparing to each host individually depending on the host's traffic.

To evaluate these methods we will use as a test a new part of the dataset. Firstly we compare new traffic to the host's history. The results can be seen in table 1. We can make some useful observations:

- When we compare each data point separately to the host's past, the mean square error is much higher. This further enforces our point for a mixture of models or clusters.
- The average traffic of each host is relatively stable as shown by the error value.

We will further investigate these facts by examining hosts that experience a higher percentage of traffic (that is less periods with zero traffic). The results can be seen in table 2. As the number of non zero traffic rises, we can see a small increase in the MSE in the case we investigate each individual data point separately. This should probably be expected. After a point the MSE decreases. This is due to the fact that many hosts with high traffic, have a very stable kind of traffic. This perhaps may due to the fact that certain network devices have a predefined role (e.g. DNS resolvers).

Next we will compare relatively to the epochs past. This method obviously has major drawbacks, as comparing each individual datapoint will have as a result a huge loss. An epoch is an average of many, different between them, events. The individual MSE within the same epoch actually is 297.68073.

Percentage of non zero traffic	no clusters	clusters	percentage difference
0	-3.633738904	-3.39167767	-6.661
10	-4.090804627	-3.77547371	-7.708
20	-5.193464837	-4.61586086	-11.12
30	-5.528976489	-4.82095644	-12.80
40	-6.741042750	-5.64690749	-16.23
50	-7.286680580	-5.95971209	-18.21
60	-8.091380270	-6.38126212	-21.13
70	-8.696028025	-6.65151644	-23.51
80	-9.270521837	-6.91730458	-25.38
90	-9.659002637	-7.11701644	-26.31

Table 3: Log-likelihood depending on the amount of traffic from each host.

2 Online EM

Next we will how the onlineEM algorithm can cope with the same data points. In the figure 4, we can see the parameters of the model after the first fit. It is difficult to compare this method to the previous one, so we will just compute the log-likelihood in the case clusters are taken into account or not. The results can be seen in table 3.

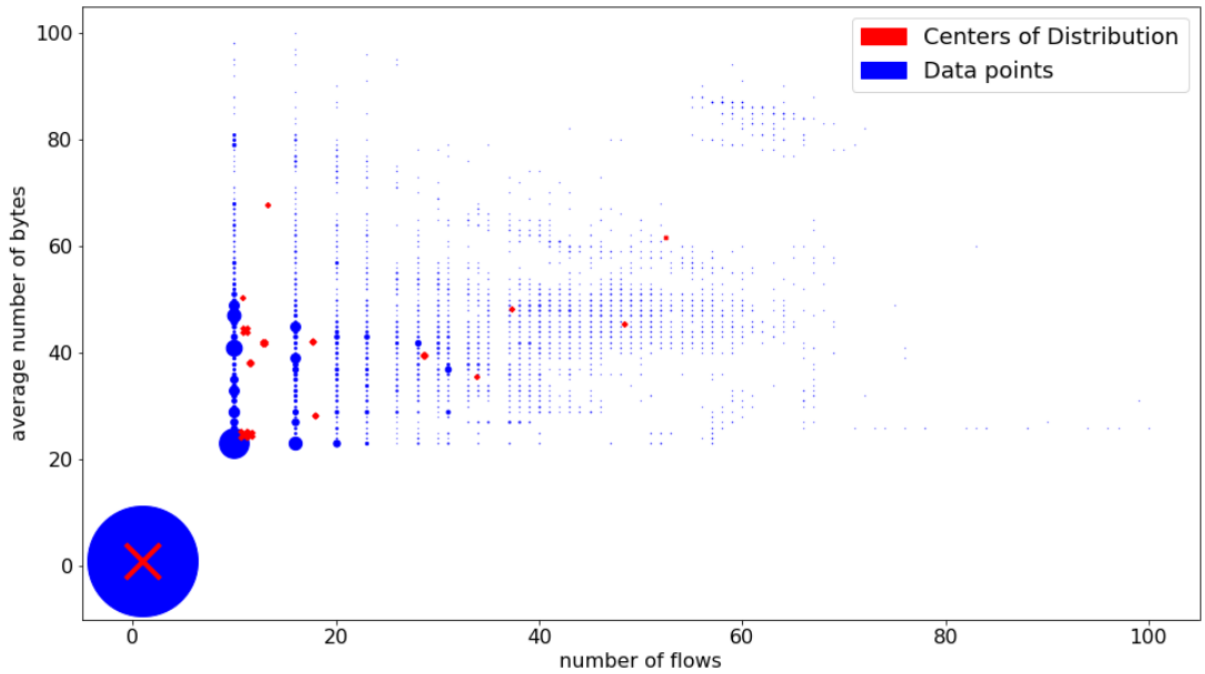

Figure 4: Average traffic for each epoch.

3 Heatmaps

These results are in a different em cluster algorithm, for better results.

The figures 6 and reffig:heatmap-epochs display the heatmaps for the hosts and epochs across all clusters.

Cluster number	number_of_flows	average_number_of_bytes	gamma
0	1.0	1.0	0.77147
1	50.8593034263127	62.60228927598113	0.00181
2	11.3597486125911	25.70856557896085	0.10109
3	12.5993229521697	71.37908447720628	0.00444
4	32.4685201056708	35.82336467695871	0.00444
5	54.1021087553742	43.61614148958785	0.00272
6	28.4390110234685	42.78378197673361	0.00175
7	11.6672545062022	43.89554098770879	0.05558
8	19.8380566847576	27.89077030263194	0.00455
9	43.5480040244836	47.32894224434174	0.00365
10	29.5859142987320	39.06804509826018	0.00678
11	40.9381803503217	47.69561584545236	0.00345
12	11.6696427111448	43.89141472077481	0.01514
13	17.5630258470568	41.20690315839984	0.01234
14	26.8200038000323	40.82639267869020	0.01072

Table 4: The parameters of the em algorithm.

The table 4 shows what the cluster parameters are

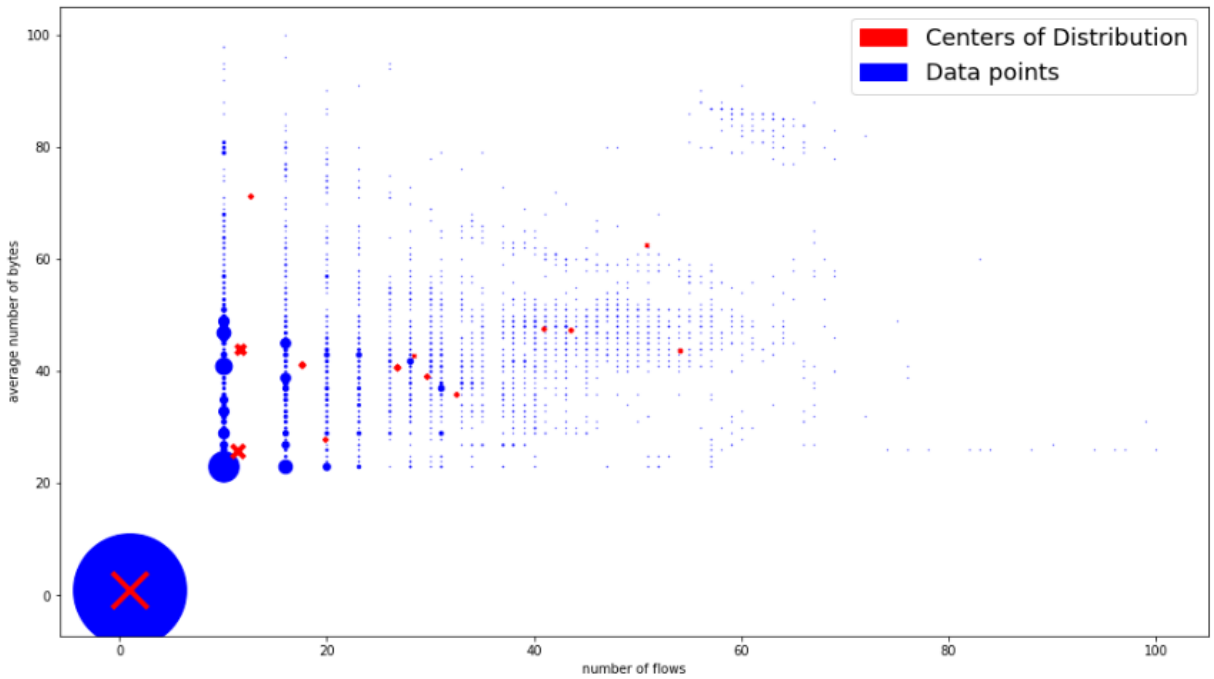

Figure 5: Datapoints

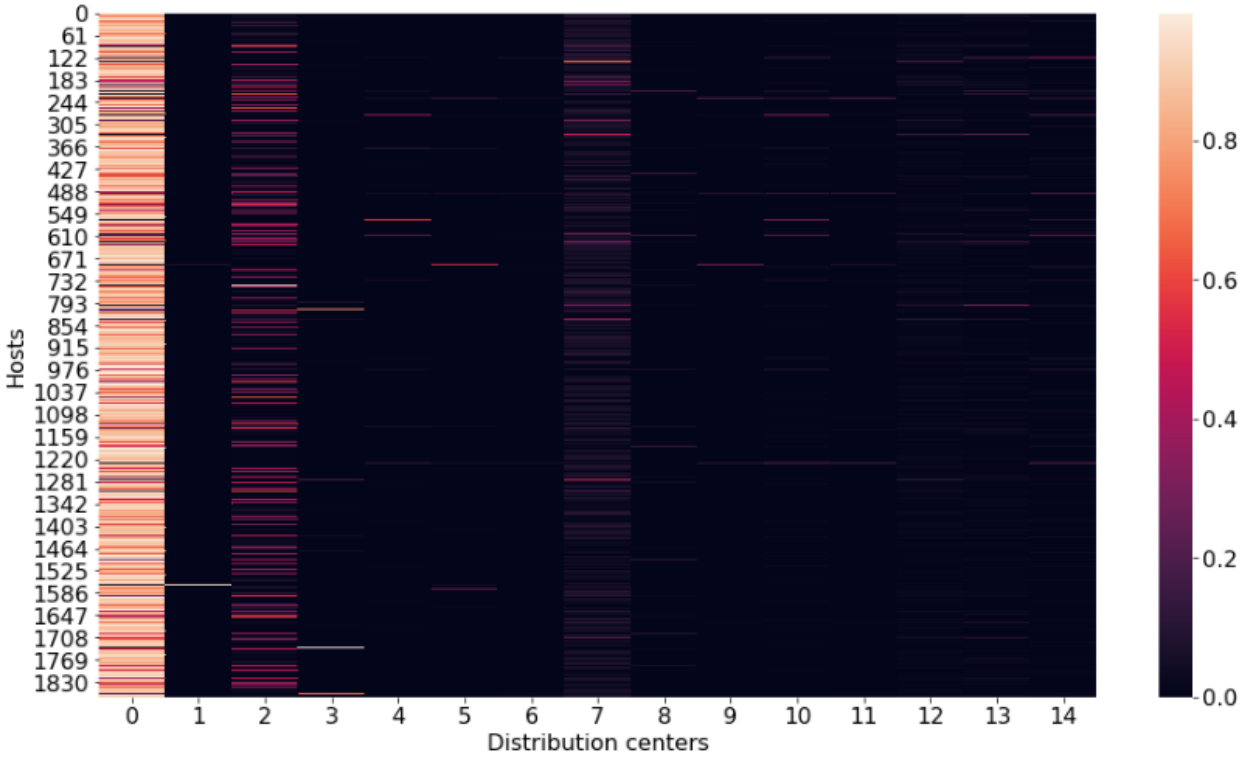

Figure 6: Heatmap for hosts.

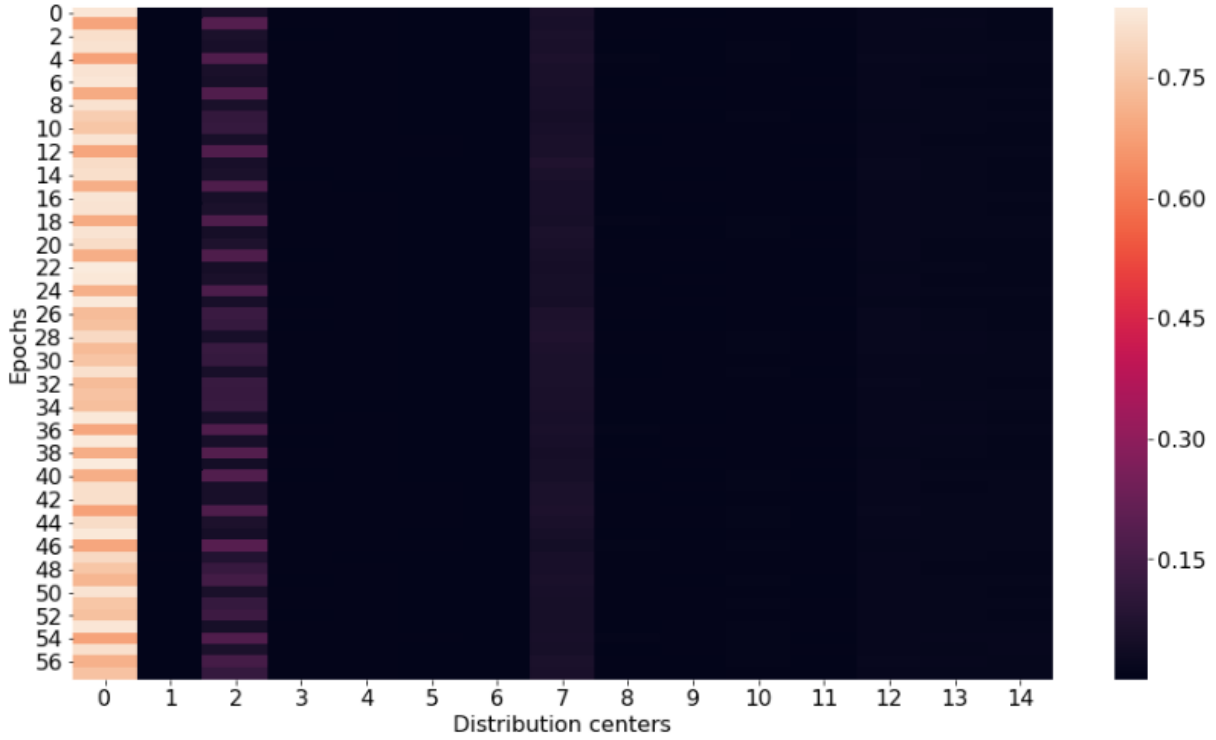

Figure 7: Heatmap for epochs.

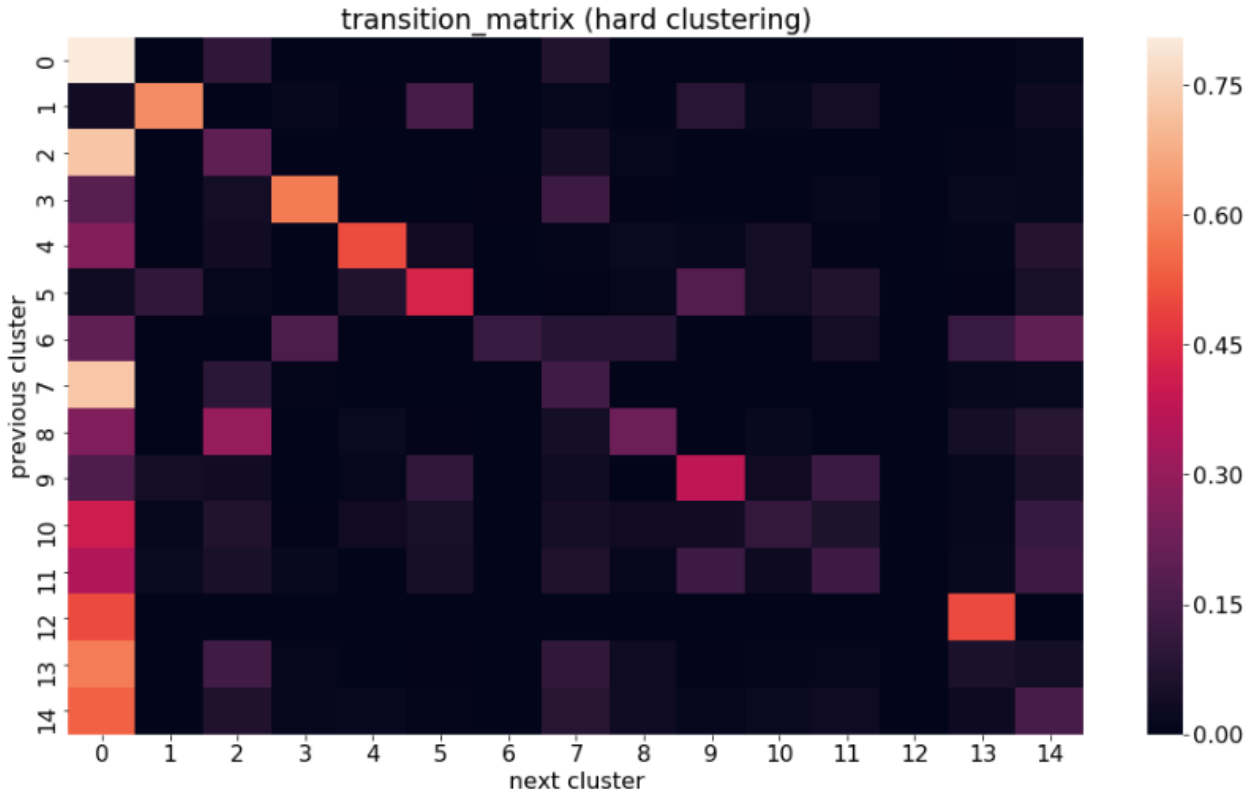

Figure 8: Heatmap for transitions for the em clusters.