CS5425 Assignement2 Task2 Report

Name: Guo Shijia ID:A0191309E

Analyse the result

medianScore	averageScore	Dominant Domain	(%percent)	Questions
1	2	Deep-learning	(100.0%)	94266
1	2	Algorithm	(100.0%)	316131
1	2	Machine-Learning	(100.0%)	364106
1	2	Computer-Systems	(100.0%)	113597
1	1	Big-Data	(100.0%)	149495
1	3	Silicon Valley	(100.0%)	54756
1	1	Compute-Science	(100.0%)	349779
1	2	Data-Analysis	(100.0%)	358556
2	3	Software-Engineering	(67.0 %)	21634
2	3	Security	(100.0%)	180299
2	3	Internet-Service-Providers	(100.0%)	24001
3	5	Programming-Language	(100.0%)	13198
4	7	Cloud-services	(100.0%)	10566
9	10	Big-Data	(100.0%)	21830
10	12	Compute-Science	(100.0%)	29063
41	45	Big-Data	(100.0%)	2880
44	53	Data-Analysis	(100.0%)	5419
45	49	Compute-Science	(100.0%)	3915
69	83	Deep-learning	(100.0%)	1190
77	98	Security	(100.0%)	1159
87	108	Silicon Valley	(100.0%)	619
112	135	Machine-Learning	(100.0%)	1739
127	134	Compute-Science	(100.0%)	884
127	134	Big-Data	(100.0%)	561
172	210	Computer-Systems	(100.0%)	359
204	230	Data-Analysis	(100.0%)	529
276	291	Compute-Science	(100.0%)	237
287	297	Big-Data	(100.0%)	153
316	430	Algorithm	(100.0%)	128
331	359	Deep-learning	(100.0%)	127
489	585	Security	(100.0%)	68
524	565	Machine-Learning	(100.0%)	214
546	557	Silicon Valley	(100.0%)	34
564	583	Big-Data	(100.0%)	65
580	621	Compute-Science	(100.0%)	62
618	726	Data-Analysis	(100.0%)	63
766	940	Computer-Systems	(100.0%)	26
823	921	Deep-learning	(100.0%)	19
1154	1192	Big-Data	(100.0%)	18
1300	1378	Compute-Science	(100.0%)	16
1474	1558	Machine-Learning	(100.0%)	49
3335	3770	Big-Data	(100.0%)	3
3636	3636	Security	(100.0%)	2
4441	5007	Machine-Learning	(100.0%)	5
10271	10271	Compute-Science	(100.0%)	2

Ans: From the cluster result, we can observe that cluster result is good, normally one cluster contains only one tag.(Because we choose a big DomainSpread)

1) A lot of hot topics like Machine-Learning, Deep-learning, there exist a lot of questions, but most of them do not got a good answer.

2)The same tags can be cluster different clusters, because the cluster number is large than the domain numbers.

3) The cluster result is imbalance, because exist about one third of clusters size is lower than 100.

Analysis of the parameters in k-means

DomainSpread : it used to split the different questions by tag, it totally based on our requirement, if we want to split the different tags into different clusters, we need to use a big number, if we want to focuses on the score and want to mixed the different tags, we can use a small number. Normally it will converged more quickly if we use number of changed points as converge condition.

KmeansKernels: The number of clusters, actually it is hard to choose a suitable cluster number. if the kmeansKernels is big, it will cost more time in each iteration, but will require less iteration number to converge. But the trend is a big kmeansKernels will need more time to converge and the total loss will be small.

KmeansEta: The converge condition, it's the average distance from each points to its centroids, the KmeansEta determined the cluster quality, a small KmeansEta means a good quality cluster, but will need more time to converge.

KmeansMaxIterations: Another converge condition, if we cannot meet the KmeansEta converge condition, we will use these to terminate our process. The big KmeansMaxIterations will cost more execution time normally.

Further discussion on the system performance

1) From Principle

We know, the quality of initial centroids is important for k-means, if we choose a suitable centroids, we will get a good cluster result and converge more quickly.

In my k-mean version, I just take distinct random points as initial centroids. Actually, we can use the kmean++ method to select the initial centroids. First we random pick one point, when we pick the second point, the probability we choose that point is inversely proportional to the distance between the two points. In that way, we can choose the initial centroids points that are sparse.

2)From the implementation

1) cache all the points

2) when we compute the points belong to which clusters or new centroids, we can parallel these 2 operations

3) we can use additional converge condition, for example, the number of points changed in this iterations.