CSE 494 Project C

Garrett Wolf

Introduction

The main purpose of this project task was for us to implement the simple k-means and buckshot clustering algorithms. Once implemented, we were asked to vary the number of clusters *k* and the number of top *N* documents on which the clustering is performed. The Vector Space Similarity and PageRank + Vector Space Similarity ranking algorithms from Project B were used to obtain results to the users query. Subsets of these results were then taken and clustered using the simple k-means and buckshot clustering algorithms. This write-up will provide a background on the k-means and buckshot algorithms, an analysis of the affects on the results from changing the values of *k* and *N*, an evaluation as to whether or not the clusters seem to correspond to any natural categories, and a comparison of the clusters produced by the k-means and buckshot algorithms.

K-Means Algorithm Background

K-Means is one of the classical clustering methods that use partitioning to arrive at the desired clusters. In the algorithm, the number of clusters, *k,* must first be selected. Next, *k* documents are chosen at random to be the centroids of each of the *k* clusters. The centroid of a cluster is a vector *c* defined as

$$c = \frac{1}{|S|}\sum_{d \in S} d$$

where S is the set of documents we are trying to cluster. After choosing the initial *k* starting centroids, each of the *N* documents is put in the cluster whose centroid is the closest to that document. To measure the closeness or similarity between a document and a centroid, we used the vector similarity metric which is defined as

$$sim(d,c) = \frac{d \bullet c}{\|c\|}$$

Notice that the dot product in the quotient is only divided by the norm of the centroid. This is because the document term weight vector *d* is already normalized whereas the centroid vector *c* was computed as the average of all the document vectors and thus might not have a length of one. After assigning each document to its initial cluster, the cluster centroids are then recomputed using the equation shown earlier. After computing the new centroids, each document is reassigned to the cluster whose centroid is closest to that document. After each iteration, we check to see if the any of the clusters have changed. We continue computing the new centroids and reassigning each document until there is no change in the clusters

between two consecutive iterations. When this happens, the calculations have converged and we are left with the final clusters. Because of the way k-means determines similarity between a document and a centroid, we tend to make clusters that have a spherical shape. Because the initial centroids are chosen at random, the resulting clusters can differ significantly based on their selection. An attempt to improve this notion can be seen in the write-up on the Buckshot algorithm. To determine the quality of the clusterings, we use two measures, namely, intracluster similarity and intercluster similarity. Intracluster similarity is the average similarity between all pairs of documents and is defined as

$$\frac{1}{|S|^2}\sum_{\substack{d\in S\\ d'\in S}} d \bullet d' = \frac{1}{|S|}\sum_{d\in S} d \bullet \frac{1}{|S|}\sum_{d'\in S} d' = c \bullet c = \|c\|^2$$

which is equal to the square of the centroid's magnitude. High quality clusters will have a high intracluster similarity which means that the clusters are tight. The second measure of quality is intercluster similarity which is the average similarity between each centroid and every other centroid. Intercluster similarity shows how far apart the clusters are. A high quality clustering will have a low intercluster similarity. When computing the similarity between two centroids, the following formula is used.

$$sim(c_1, c_2) = \frac{c_1 \bullet c_2}{\|c_1\| * \|c_2\|}$$

which is different that the formula used to calculate similarity between a document and a centroid. In conclusion, a high quality clustering will maximize the intracluster similarity and minimize the intercluster similarity.

Buckshot Algorithm Background

Buckshot is a hybrid clustering method that combines the partitioning and hierarchical clustering methods. More precisely, it combines Hierarchical Agglomerative Clustering (HAC) and K-Means Clustering by using "HAC to bootstrap K-Means". In the algorithm, the number of clusters, *k,* must first be selected. Next we create the starting clusters by randomly selecting $\sqrt{N}$ documents from *S,* and putting each of the $\sqrt{N}$ documents in its own cluster. We then compute the similarity between every cluster and every other cluster and merge the two closest clusters into one. We continue merging the two closest clusters until we are left with *k* clusters. The centroids of the *k* clusters are then used as the starting centroids for the k-means algorithm described above. The main reason for using HAC to select the k-means starting centroids is that it can help avoid selecting bad starting seeds. HAC is known to produce quality cluster but is often too costly to run. Therefore, by using HAC to select the starting centroids for k-means, we get the benefits from both algorithms. Another way to improve the quality of k-means clusters is to look for a kink in the intracluster similarity vs number of clusters curve which we will see later.

Effect of Changing the Number of Clusters

By changing the number of clusters *k*, the intracluster similarity increases while the intercluster similarity decreases. This makes sense because as more clusters are created, they contain fewer points and thus are tighter and more intraclusterally similar. For example, when we have *N* clusters where each cluster consists of just one point, the intracluster similarity is 1 because each document is 100% similar to itself. Thus, the closer *k* gets to *N*, the higher the intracluster similarity. Likewise, because the points are being broken into more clusters, the clusters are not as general and are thus further apart making them less interclusterally similar. Appendix A shows the intracluster similarity graphs in for the queries "parking decal" and "computer science" which both show that as the number of clusters increases, so does the intracluster similarity. Similarly, they show that as the number of clusters increases, the intercluster similarity decreases for both of the queries. We notice that the change in similarity is higher for the simple k-means plot as *k* increases than it is for the buckshot plot. When *k* is at the lower end, the buckshot's intracluster and intercluster similarity is pretty close to the simple k-means intracluster and intercluster similarity values for *k* and the higher end.

Effect of Changing the Number of Top Documents

By changing the number of starting top documents *N*, the change in intracluster and intercluster similarity is not quite clear. Appendix B shows the plots of intracluster and intercluster similarity using simple k-means with values of *k* equal to 3, 6, and 9 for the query "parking decal". The similarities are calculated at values of *N* ranging from 50 to 100. Notice that the curves seem to jump up and down based on the value of *N* but that the curve where *k* equal 3 seems to jump the most. This is because when we only have 3 clusters, adding 10 more documents can significantly change the similarity values for better or worse. Whereas the curve for *k* equals 9 doesn't seem to change very much. The more clusters we have, the less the effect from adding another document to the clustering corpus.

Natural Categories

When clustering using the simple k-means and buckshot algorithms, clear natural categories appear among the clusters. In Appendix C, the resulting clusters from the query "computer science" are shown. In this example, 3 clusters were created and they each appear to have their own categories. Almost all the results are related to the computer science program and include information regarding transferring, requirements, class flowcharts, scholarships, and testing. They appear to fall into categories based on their URL as well as their content. For the simple k-means clustering, the first cluster gives information regarding the undergraduate computer science program. The second cluster is more related to the office of the Vice President and gives the formal descriptions of the different computer majors. The third cluster is not as closely related as the previous two but it includes pages about tests required to get into ASU as well as scholarships to help students get in financially. The buckshot clustering results, also shown in Appendix C, have clusters identical to the first two clusters in the k-means example. The third cluster in the buckshot clusters deals with the social science computing cluster. Because only one page is shown in this cluster, there are probably not too many pages dealing with the social science computing cluster. Because of this, we know that the intracluster similarity for this third cluster is much higher than the rest of the clusters given.

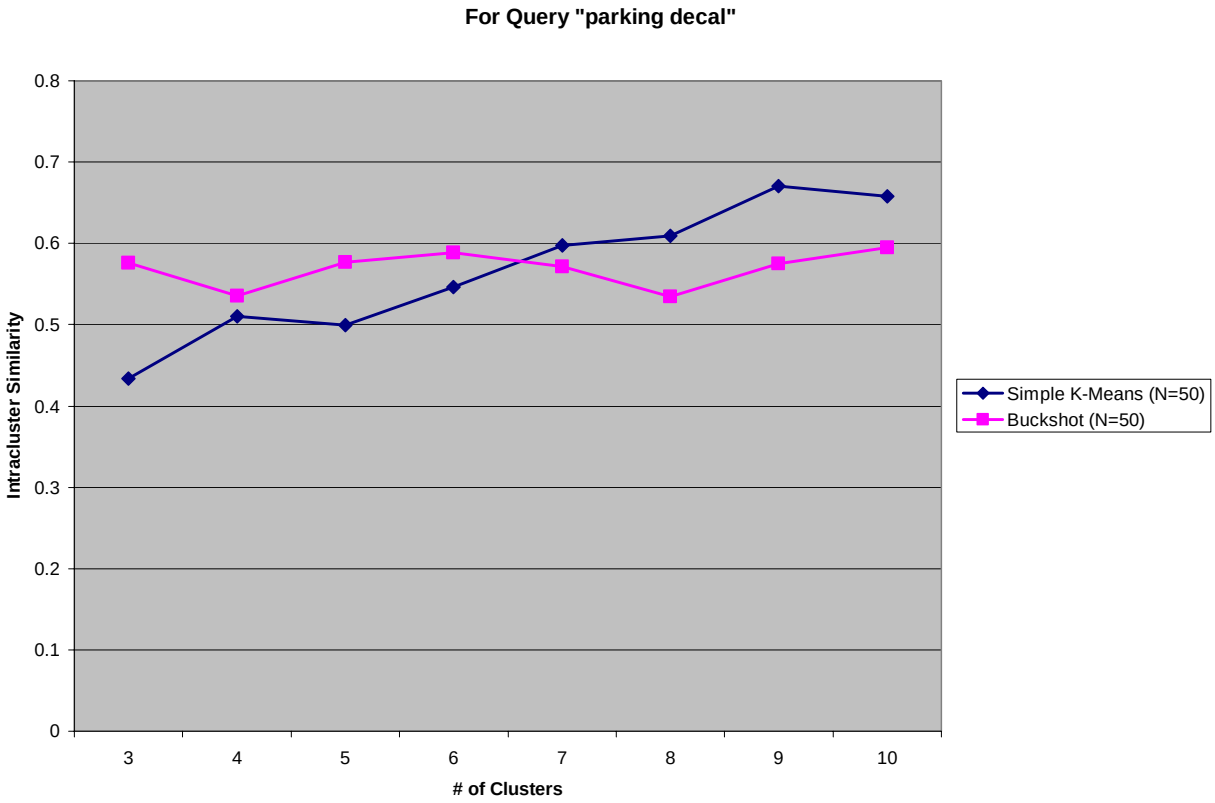
K-Means vs. Buckshot

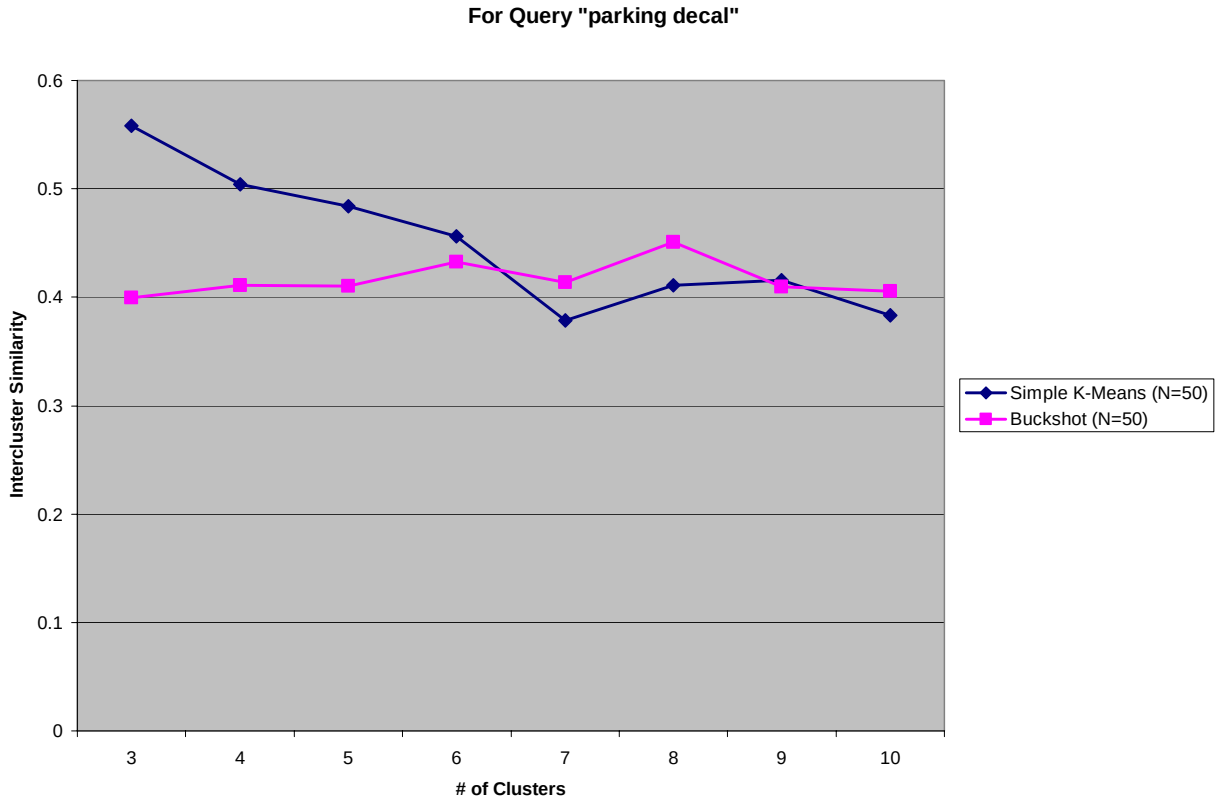
The clusters given by the k-means and buckshot clustering algorithms are usually quite similar when only displaying the top 3 representative documents from each cluster. As all four graphs in Appendix A show, the buckshot algorithm has better intracluster and intercluster similarity when *k* equals 3. This is because it uses HAC to determine better starting centroids which results in better clusters. However, notice that for larger values of *k* such as 10, the simple k-means algorithm produced better intracluster and intercluster similarity results. Although slightly higher better than the buckshot similarity values where *k* equals 10, notice that they are only slightly higher than buckshot's values when *k* was equal to 3. Because the similarity values get better as *k* increases, this is just showing how simple k-means develops better clusters as *k* grows but that buckshot provides good similarity scores at all values of *k*. This fact shows that the buckshot algorithm produced better results. To create these graphs, 5 trials at each value of *k* for each clustering algorithm were recorded and averaged. As stated in the previous section, the buckshot and k-means algorithms had two identical clusters for the query "computer science" shown in Appendix C. However, the third cluster in the k-means results is a mix of pages that are not that similar. Conversely, the third cluster in the buckshot results consists of a single rare page which is most definitely similar to itself.

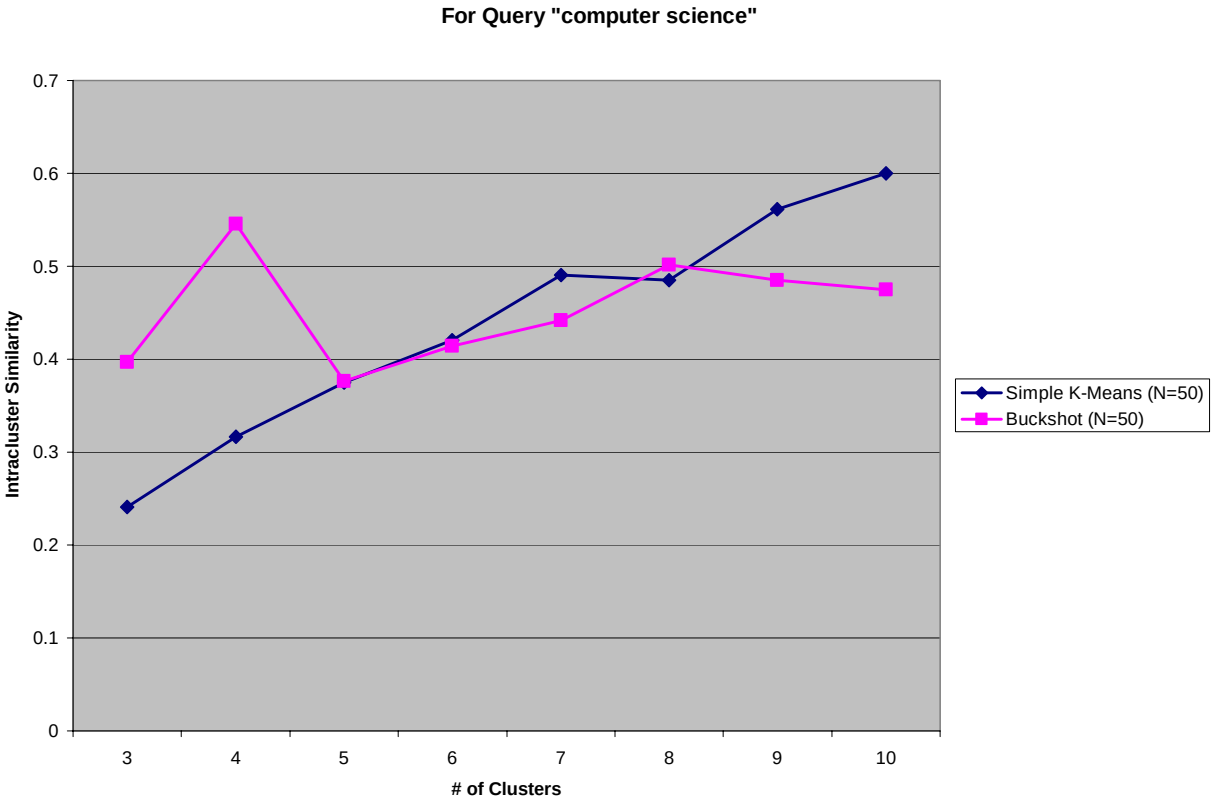
Conclusion

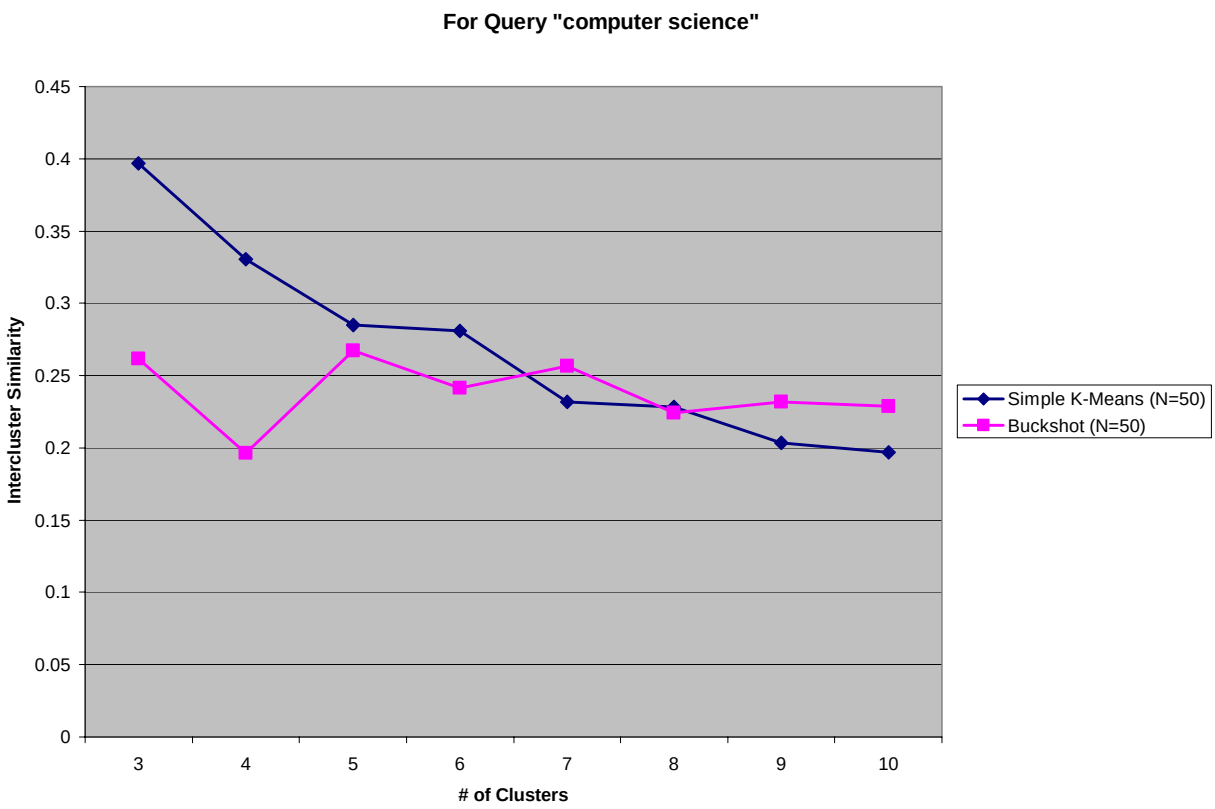
The k-means and buckshot clustering algorithms appear to provide quality results which correspond to somewhat natural categories. Because the buckshot uses HAC to select its starting centroids, it benefits from more constant, better intracluster and intercluster similarity scores for all tested values of *k*. HAC produces high quality clusters but is too costly to implement, therefore the hybrid buckshot algorithm only uses HAC to select good starting centroids for k-means. The HAC algorithm has a running time of $O(n^2)$ so by performing it on only $\sqrt{N}$ documents, buckshot is able to maintain the k-means running time of O(n) and avoid the problems of bad seed selection. Another way to improve the quality of the k-means clusters would be to look at the Similarity vs. Number of Clusters graphs in Appendix A and choose a value for *k* that corresponds to a kink in the graph. Overall, buckshot provides the better quality clusters on average and

Appendix A

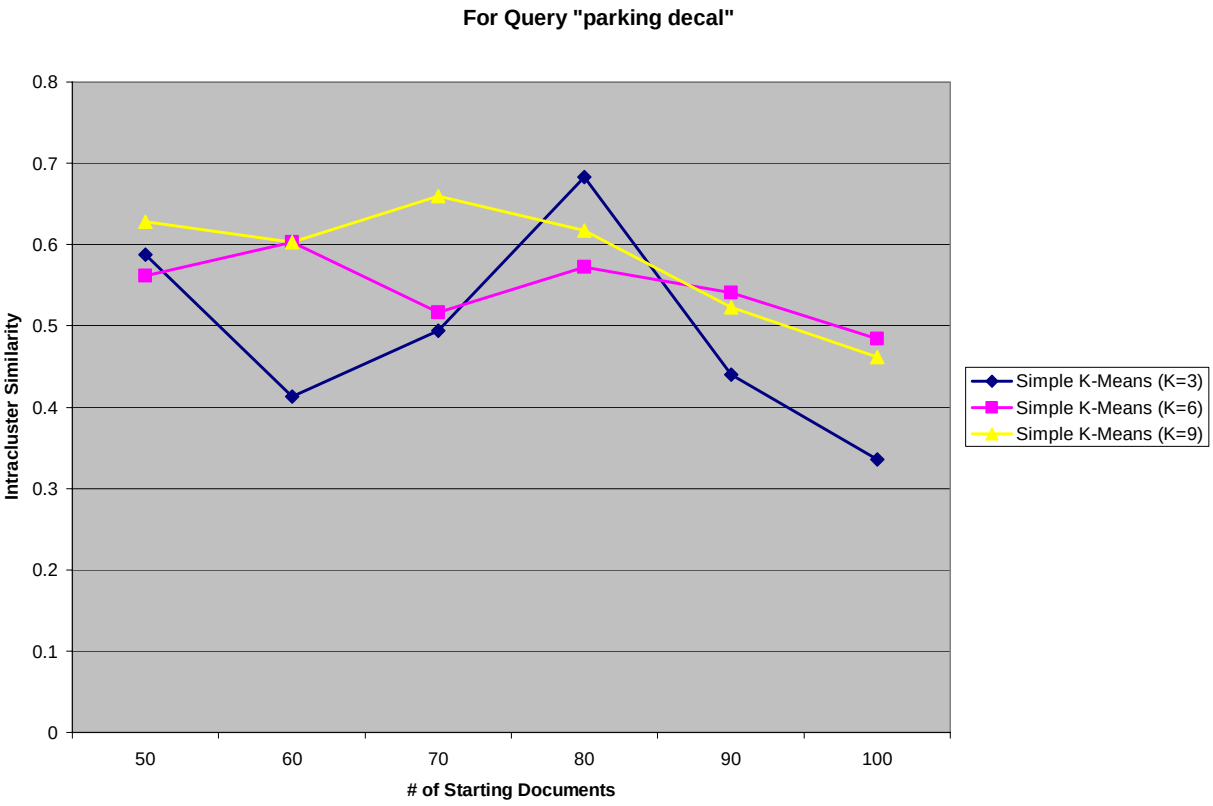

For Query "parking decal"
Intracluster Similarity
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
3
4
5
6
7
8
9
10
# of Clusters
Simple K-Means (N=50)
Buckshot (N=50)

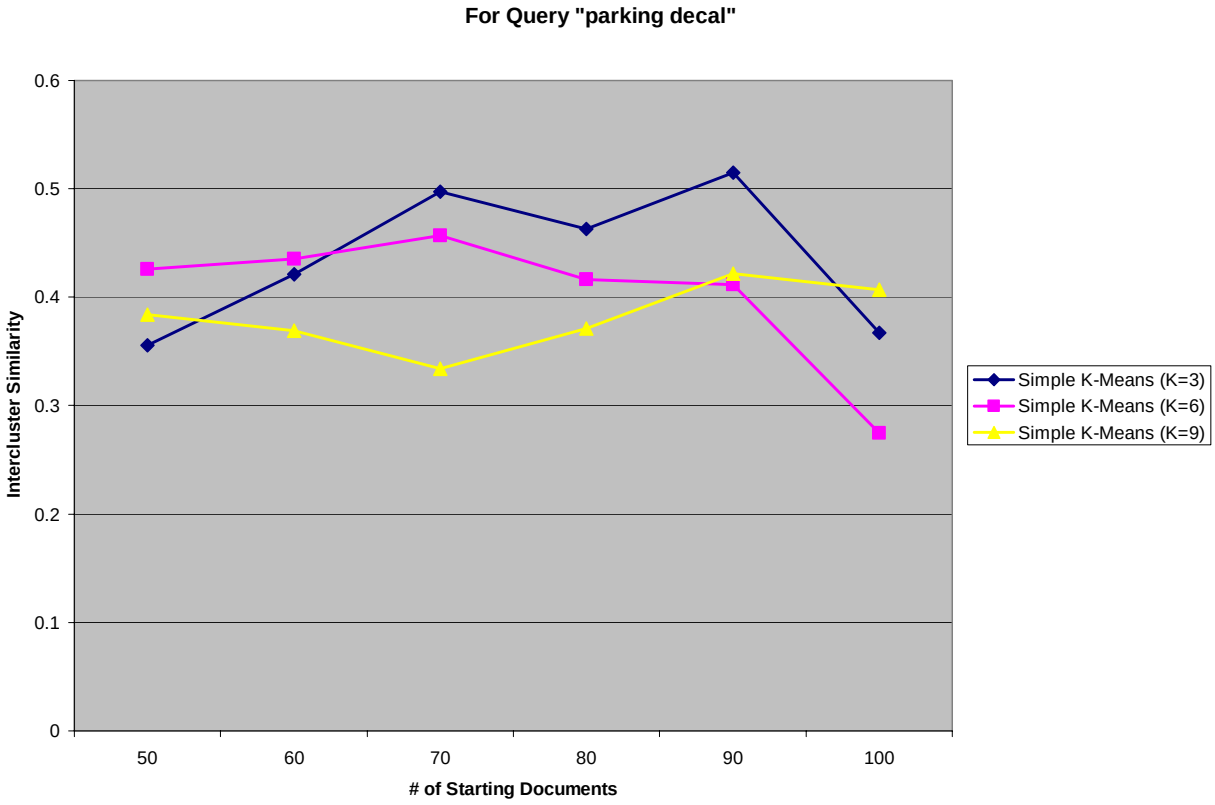

For Query "parking decal"
Intercluster Similarity
0.6
0.5
0.4
0.3
0.2
0.1
0
3
4
5
6
7
8
9
10
# of Clusters
Simple K-Means (N=50)
Buckshot (N=50)


For Query "computer science"
Intracluster Similarity
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
3
4
5
6
7
8
9
10
# of Clusters
Simple K-Means (N=50)
Buckshot (N=50)


For Query "computer science"
Intercluster Similarity
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
3
4
5
6
7
8
9
10
# of Clusters
Simple K-Means (N=50)
Buckshot (N=50)

Appendix B


For Query "parking decal"
Intracluster Similarity
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
50
60
70
80
90
100
# of Starting Documents
Simple K-Means (K=3)
Simple K-Means (K=6)
Simple K-Means (K=9)


For Query "parking decal"
Intercluster Similarity
0.6
0.5
0.4
0.3
0.2
0.1
0
50
60
70
80
90
100
# of Starting Documents
Simple K-Means (K=3)
Simple K-Means (K=6)
Simple K-Means (K=9)

Appendix C

```
Query: Computer Science
Searching for: computer science

************************************************
               Simple K-Means
************************************************
# of Clusters: 3
Top-N: 50
Avg. Intracluster Similarity: 0.3277882158504025
Avg. Intercluster Similarity: 0.37636840853019504

********** CLUSTER 1 **********
0.2964575020094826     www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%StuNotAdmit.htm
0.27667469594977323    www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%Transfer.htm
0.2639248966859938     www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%CheckFlowBS.htm

********** CLUSTER 2 **********
0.46786310491741995    prism.asu.edu%%education_studentresearch.asp
0.3729480664567661     www.asu.edu%%provost%%smis%%ceas%%bs%%csbs.html
0.3211684973515285     www.asu.edu%%provost%%smis%%ceas%%bse%%csebse.html

********** CLUSTER 3 **********
0.30128595140899445    enuxgs.eas.asu.edu%%ccolbou%%src%%personal.html
0.2948817364000159     www.eas.asu.edu%%~csedept%%Students%%Scholarships%%CSEscholarships.htm
0.28543782349168306    www.asu.edu%%uts%%compbasedtesting%%whole.htm




Query: Computer Science
Searching for: computer science

****************************************************
       CLUSTERS DERIVED AFTER HAC
****************************************************
************************************************
               Buckshot
************************************************
# of Clusters: 3
Top-N: 50
Avg. Intracluster Similarity: 0.7974236432015335
Avg. Intercluster Similarity: 0.11572593490636474

********** CLUSTER 1 **********
0.2964575020094826     www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%StuNotAdmit.htm
0.27667469594977323    www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%Transfer.htm
0.2639248966859938     www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%CheckFlowBS.htm

********** CLUSTER 2 **********
0.18981386481506848    www.asu.edu%%clas%%sociology%%sscc%%index.html

********** CLUSTER 3 **********
0.46786310491741995    prism.asu.edu%%education_studentresearch.asp
0.3729480664567661     www.asu.edu%%provost%%smis%%ceas%%bs%%csbs.html
0.3211684973515285     www.asu.edu%%provost%%smis%%ceas%%bse%%csebse.html


****************************************************
       CLUSTERS DERIVED AFTER K-MEANS
****************************************************
************************************************
               Buckshot
************************************************
# of Clusters: 3
Top-N: 50
Avg. Intracluster Similarity: 0.5544911450003173
Avg. Intercluster Similarity: 0.22181637182704725
```

********** CLUSTER 1 **********
0.2964575020094826 www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%StuNotAdmit.htm
0.27667469594977323 www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%Transfer.htm
0.2639248966859938 www.eas.asu.edu%%~csedept%%AcademicPrograms%%Undergraduate%%CheckFlowBS.htm

********** CLUSTER 2 **********
0.18981386481506848 www.asu.edu%%clas%%sociology%%sscc%%index.html

********** CLUSTER 3 **********
0.46786310491741995 prism.asu.edu%%education_studentresearch.asp
0.3729480664567661 www.asu.edu%%provost%%smis%%ceas%%bs%%csbs.html
0.3211684973515285 www.asu.edu%%provost%%smis%%ceas%%bse%%csebse.html