Assignment 4

James Lowrey

Clustering is a memory intensive process, and when using large datasets various methods can be employed to allow clustering to process the data. One of the most common methods is to take representative samples of the data for clustering to process. This allows the number of data points processed to be much smaller than the original corpus and to maintain results similar to what would be expected if clustering was ran on the entire database. In this lab, the data size was reduced by a little more than ¼ by restricting the number of sample points to 5000. First the data was partitioned into 20 sections, and the same number of samples were selected from each partition without replacement. The partitioning is to ensure that the data is relatively representative, and attempts to limit the effects of randomly many nearby data points. Data was not replaced in the sampling to prevent selecting the same piece of data multiple times, further improving the representativeness of the sampling (3). Each partition has about 1078 samples, and 250 samples are selected from each partition, almost 25% of the partition. If data replacement was used there would be a non-negligible chance of reselecting the same data.

**Time of Raw (Non-Sampled) Clustering Algorithm Prediction (seconds, 21578 samples)**

|  |  |  |
| --- | --- | --- |
|  | Euclidean | Cosine |
| K-Means | 30.2773 | 27.8734 |
| DBSCAN | 137.5356 | MemError |

**Time of Sampled Clustering Algorithm Prediction (s****econds, 5000 samples)**

|  |  |  |
| --- | --- | --- |
|  | Euclidean | Cosine |
| K-Means | 7.7462 | 8.5513 |
| DBSCAN | 11.7726 | 4.2628 |

The sampled data was then vectorized to allow clustering to occur. The word (feature) and class vectors of each document were vectorized before being fed to the SKLearn clustering toolkit. Word/feature vectorization was done by first finding all unique words and classes in the processed/reduced article data, defining a set ordering of the values, and keeping the ordering consistent across all document data points. Class labels were simply transformed into integers. In the original corpus, each document can have 0 or more class labels. After vectorization, documents were allowed to have more than one label but any document without labels was given the class label 0 to allow for easier manipulation of data.

SKLearn's packages were used for the clustering implementations. The two clustering schemas used were K-Means (Euclidean) and DBSCAN. K-Means is a fast, relatively simple clustering algorithm that separates a given number of clusters based upon distance from a calculated centroid. Initial centroid choices are important as poor centroids can lead to poorly clustered data (as the centroid can lie between data). SKLearn uses a default of 10 different centroid seeds which are chosen “in a smart way” using its k-means++ algorithm (4). I believe this runs k-means on a selected representative sample. DBSCAN is another clustering methodology that finds core data points of high density and expands clusters outwards from them. It is a strong algorithm for finding clusters of similar density, is resistant to noise, and can create irregularly shaped clusters (5). DBSCAN finds core points based upon it neighbors, there must be a minimum number of neighbor points within an epsilon distance. SKLearn uses NearestNeighbor algorithm modules to decide upon neighbors.

In this lab, Spherical K-Means was emulated through usage of L2 normalization of data and use of Euclidean K-Means. Usually, spherical K-Means implies a cosine distance function and the constraint that centroid vectors are forced to lie on the unit circle (L2 normalized). SKLearn does not have a default Spherical K-Means implementation, but using L2 normalization on the original data, the default K-Means becomes proportional to the Spherical K-Means, and the results are similar. L2 normalization on the data makes Euclidean K-Means proportional to the Cosine Distance function. When the magnitude of direction/feature vectors are constrained to equal 1, the square of the Euclidean distance is equal to twice the magnitude of the cosine distance.

Proof of the Relationship Between Squared Euclidean Distance and Cosine Distance (2)

This initial normalization of the data is important as it allows the subsequent Euclidean K-Means to achieve results proportional to Spherical K-Means. As the results are proportional, clusterings will be similar (2). Spherical K-Means typically performs better on text classification as representative feature vectors are classically very sparse (3).

A variety of metrics were used for clustering quality.

Sources:

1. “Margin-based local regression for adaptive filtering”. Proceedings of the 12th International Conference on Information and Knowledge Management, pages 191-198, 2003. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.307.1537>

2. Ashok N. Srivastava, Mehran Sahami. “Text Mining: Classification, Clustering and Applications”, pages 162-163. [https://books.google.com/books?id=BnvYaYhMl-MC&pg=PA162&lpg=PA162&dq=clustering+l2+normalization+spherical+k+means&source=bl&ots=oj1IxpWWgg&sig=7r7u\_3gCwBQu9NgJg-nrZrCgPyo&hl=en&sa=X&ved=0CEEQ6AEwBGoVChMIzerw8aH6yAIVyz8-Ch3UDg2c#v=onepage&q=clustering%20l2%20normalization%20spherical%20k%20means&f=false](https://books.google.com/books?id=BnvYaYhMl-MC&pg=PA162&lpg=PA162&dq=clustering+l2+normalization+spherical+k+means&source=bl&ots=oj1IxpWWgg&sig=7r7u_3gCwBQu9NgJg-nrZrCgPyo&hl=en&sa=X&ved=0CEEQ6AEwBGoVChMIzerw8aH6yAIVyz8-Ch3UDg2c" \l "v=onepage&q=clustering l2 normalization spherical k means&f=false)

3. Mary Parker, “Sampling With Replacement and Sampling without Replacement”. <https://www.ma.utexas.edu/users/parker/sampling/repl.htm>

4. “sklearn.cluster.Kmeans”. Scikit Learn. <http://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html>

5. “sklearn.cluster.DBSCAN”. Scikit Learn. <http://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html>