CS677 final proposal

What is the computation and why is it important?

k-means clustering is rather easy to apply to even large data sets. It has been successfully used in market segmentation, computer vision.

Abstraction of computation:

1. Calculate the Euclidean distance of two points to set clusters.
2. In each cluster, calculate the average distance to its center, and calculate the total distance of all cluster(add).

$$J_{SSE} = \sum_{i=1}^{k} \sum_{x \in D_i} || x - \mu_i ||^2$$

Suitability for GPU acceleration:

1. Each points distance is independent, and permanent , it is the core calculation in the k-means algorithm. we can use parallel computing.
2. Synchronization and Communication: The shared memory data cluster_info should be synchronized, and in each loop, should update the cluster center, then calculate the distance to compare.
   Data structure: struct Tuples{`int num, int dimension, int k, float *tuples `} to calculate distance, The distance calculation and the cluster_info update should be synchronize.

Calculation process:

1. Random init cluster_info, use dimension * n threads to calculate the clusters_sum of each dimension with atomic operator.
2. use the cluster_info to calculate an int *clusters_num, the number of each cluster.
3. use k * d threads calculate the centers with clusters_sum / clusters_num.
4. Use n * k threads to calculate distance, each thread calculate the distance between center and point(The centers and the points use many times may store in the shared memory).

5. Update the clusters_info with the distance()
   If k is small in precess 3 , there are not enough active thread to hide latency, may calculate on CPU.

Difficulties:

Copy Overhead: If the data size is large, copy the data from CPU to GPU may take a long time.

How to update the new center on GPU, Use shared memory in process 4 calculate the distance.