**Abstract.** This project aims to mining Sequential pattern data by using Genetic algorithm based on Hadoop Mapreduce. The basic idea is to extract the fittest rules from a huge sequential datasets.

* ***The Framework should be as following:***

1. The job should be done with many rounds until a **maximum number of generations** is reached.
2. **First round :** 
   1. **Map Class**: Each Mapper has its own sub-population → Generate **N** chromosomes, each record consider as a chromosome; Generate all possible rules from these N chromosomes. (In the 2nd Mapper Class we will define the **maximum number of generations** to check if we have to proceed or stop.)
   2. **Combiner** : Should calculate the fitness value which will be consider as the ***support*** of the rule.. ***Support*** how many times the rule occurs in this generation/ **88158 *// number of records in the dataset***
   3. **Reducer**: Select the best rules (highest finesses) and greater than or equal   
      **mininmum\_Fitness**; Hint: the reducer must take the maximum fitness (values) for each rule (key) not the summation.
3. **Second round:**
   1. **Map Class**: Do the crossover and mutation on the best rules and generate all new possible rules.
   2. **Combiner:** recalculates the fitness for those new rules.
   3. **Reducer**: Selection operation is conducted between the parents and the children, and the best rules chosen between the parents and the children; The better replaced the other one.
4. **These two around are repeated until the maximum number of generations is reached, In which case the Output file should contains all the best rules and their corresponding fitness values e.g. Chromosome 1: [3] 🡪 [39, 48]; Fitness = 0.66…..etc**

**Input:**

population size: N

Maximum number of generations: G

**Output:**

Interesting sequence pattern: S

**Algorithm:**

1. Initialize counter t = 0
2. Generate population Pold of size N

**1st Map Class**

1. For each chromosome iPold
2. Generate rule i from chromosome i

**1st Combiner**

* 1. Calculate fitness of rule i

1. Initialize empty population Pnew of size N
2. For each chromosome i and i+1Pold

**2nd Map Class**

* 1. Mutate and crossover chromosome i with i+1
  2. Place reproduced chromosome in Pnew
  3. Generate rule i from the reproduced chromosome
  4. Calculate fitness of rule i

**2nd Combiner**

1. Select fittest rules from Pold&Pnew then place it in Pold

**Reducer Class**

1. Set t = t +1
2. If (t > G) then

S = Pold

**2nd Map Class**

Stop algorithm

Else

Go to Step 5