FAE-st

September 5, 2022

1 Discovering Frequent Patterns in Big Data Using FAE Algorithm

In this tutorial, we will discuss two approaches to find frequent patterns in big data using FAE algorithm.

1. Basic approach: Here, we present the steps to discover frequent patterns using a single specified counte

---

1.1 Basic approach: Executing FAE on a single dataset at a particular specified counte value

Step 1: Import the FAE algorithm

```
[1]: from PAMI.frequentPattern.topk import FAE  as alg
```

Step 2: Specify the following input parameters

```
[2]: inputFile = 'transactional_T10I4D100K.csv'

     kCount=100   #Users can also specify this constraint between 0 to 1.

     seperator='\t'
```

Step 3: Execute the FAE algorithm

```
[3]: obj = alg.FAE(iFile=inputFile, k=kCount, sep=seperator)      #initialize
     obj.startMine()                 #Start the mining process
```

```
100
FAE has successfully generated top-k frequent patterns
```

Step 4: Storing the generated patterns

Step 4.1: Storing the generated patterns in a file

```
[4]: obj.savePatterns(outFile='frequentPatternsMinSupCount100.txt')
```

Step 4.2. Storing the generated patterns in a data frame

```
[5]: frequentPatternsDF= obj.getPatternsAsDataFrame()
```

Step 5: Getting the statistics

Step 5.1: Total number of discovered patterns

```
[6]: print('Total No of patterns: ' + str(len(frequentPatternsDF)))
```

```
Total No of patterns: 100
```

Step 5.2: Runtime consumed by the mining algorithm

```
[7]: print('Runtime: ' + str(obj.getRuntime()))
```

```
Runtime: 1.5778701305389404
```

```
[8]: ##### Step 5.3: Total Memory consumed by the mining algorithm
```

```
[9]: print('Memory (RSS): ' + str(obj.getMemoryRSS()))
     print('Memory (USS): ' + str(obj.getMemoryUSS()))
```

```
Memory (RSS): 214642688
Memory (USS): 176705536
```