CLOUD COMPUTING APPLICATIONS

Frequent Pattern Mining

Roy Campbell & Reza Farivar

Spark mllib and fpm

spark.mllib provides a parallel implementation of FP-growth, a popular algorithm to mining frequent itemsets.

FP-growth algorithm: Given a dataset of transactions,

1. Calculate item frequencies and identify frequent items,
2. Use a suffix tree (FP-tree) structure to encode transactions,
3. Extract frequent itemsets from the FP-tree.

Frequent Pattern Mining: FP-growth

FPgrowth takes a RDD of transactions, where each transaction is an Array of items of a generic type.

Calling FPGrowth.run with transactions returns
an FPGrowthModel that stores the frequent item sets with their frequencies.

```
from pyspark.mllib.fpm import FPGrowth

data = sc.textFile("data/mllib/sample_fpgrowth.txt")
transactions = data.map(lambda line: line.strip().split(' '))
model = FPGrowth.train(transactions, minSupport=0.2, numPartitions=10)
result = model.freqItemsets().collect()
for fi in result:
    print(fi)
```