GraphX: A Resilient Distributed Graph System on Spark
Aman Shankar(as5171)

1. Introduction

With the increase in the usage of big graphs usage there is a need for robust graph construction mechanism, computation and fault tolerant and support for machine learning algorithms such as data mining. GraphX combines the advantages of data-parallel and graph-parallel systems.

2. Goals

To create an interactive fault tolerant engine which uses Spark framework and its RDDs to provide users with interactively compute load, transform and compute on massive graphs.

3. Approach

GraphX extends Resilient Distributed Abstraction to Resilient Distributed Graphs. Each RDGs has properties of Pregels and PowerGraphs. Additionally, stores the information about vertices and edges. The engine creates a Directed Acyclic Graphs(DAGs). Furthermore, RDG's are comes with the methods such as filterVertices(predicate), mapVertices(f), mapEdges(f), edges(), vertices(),etc. For parallel computation, it requires each each vertex and corresponding edges to be processed with respect to its neighbors.

4. Results

GraphX was compared over Apache Mahout and PowerGraph and it was run over Apache EC2 instance. GraphX was 7x slower than PowerGraph and 8x time faster than Apache Mahout.

5. Conclusion

GraphX is a graph engine that uses the parallel system from both data and graph. Additionally, it simplifies the abstraction, construction and transformation of graph simple. RDGs also bring Pregel and PowerGraph.

6. Comments

GraphX uses tries to unify two programming frameworks into one which is a positive aspect. Additionally, it uses the advantage of Spark and RDD to give a fault-tolerant engine.