Big Data Challenge
Clique a Milano
Schema di implementazione con Hadoop

Michele Carignani, Alessandro Lenzi

4 marzo 2014

1 Goals

1. Split and aggregate MItoMI by hours
2. Split and aggregate TelcoMI by hours
3. Compute sum of MItoMI hours
4. Compute arcs probabilities in MItoMI
5. Look for Cliques (SCC? Clustering? WCC?)
6. Aggregate Cliques and get statistics
7. Integrarate with other datasets (?)
    - Execute same research for cliques on milan but using MI to Provinces dataset
    - Look for tweets in discovered cliques
    - look for events in milano today near cliques
    - Compare with same analysis on Trento (?)

2 Sub schemas

2.1 1, (2?), 3

```
Mapper<Timestamp, Value, "Day:Hour:Source" , "Dest:Value">
Reducer<"Day:Hour:Source", iterator<Dest:Value>, "Day:Hour:Source:Sum", "String/List">
```

Issues

- How to split the telco? use same map-reduce with switch on mapper key content or different map reduce? Note: Files are distinct, data is not mixed, so performances should be the same
- is it possible to double pass on the data and compute here the probabilities?

2.2 4

```
Mapper<D:H:S:Sum, Dest:Val, "Day:Hour:Source" , "Dest:Prob">
no Reducer
```

2.3 5

2.4 6

Come aggregare i dati sulle cricche orarie giorno per giorno?

- Cercare persistenza nella stessa fascia oraria su giorni differenti
- Cercare persistenza fra fasce orarie contigue
- Cercare clique che si ripetono (non importa se nella stessa fascia oraria)
- analisi sulla media: creare una distribuzione media di archi su tutto il periodo
- cercare eventi in corrispondenza di clique che appaiono o scompaiono
- clusterizzare le clique? cioè cercare cliques simili per zona coperta, dimensione, fascia oraria di appartenenza, stesso giorno della settimana etc.
- tutte queste cose insieme?

3 Distributed graphs frameworks

Apache Giraph From `https://giraph.apache.org`: "Apache Giraph is an iterative graph processing system built for high scalability. For example, it is currently used at Facebook to analyze the social graph formed by users and their connections. Giraph originated as the open-source counterpart to Pregel, the graph processing architecture developed at Google and described in a 2010 paper. Both systems are inspired by the **Bulk Synchronous Parallel** model of distributed computation introduced by Leslie Valiant. Giraph adds several features beyond the basic Pregel model, including master computation, sharded aggregators, edge-oriented input, out-of-core computation, and more. With a steady development cycle and a growing community of users worldwide, Giraph is a natural choice for unleashing the potential of structured datasets at a massive scale. To learn more, consult the User Docs section above."

GraphX From `http://amplab.github.io/graphx/`: "GraphX extends the distributed fault-tolerant collections API and interactive console of Spark with a new graph API which leverages recent advances in graph systems (e.g., GraphLab) to enable users to easily and interactively build, transform, and reason about graph structured data at scale."

Apache Spark Website `http://spark.incubator.apache.org/`: Runs 100x faster then Hadoop, on top of Yarn. "Apache Spark™ is a fast and general engine for large-scale data processing."

X-RIME Website `http://xrime.sourceforge.net/`. "Hadoop based large scale social network analysis". Currently Supported SNA Metrics and Structures:

- vertex degree statistics
- weakly connected components (WCC)
- strongly connected components (SCC)
- bi-connected components (BCC)
- ego-centric network density
- bread first search / single source shortest path (BFS/SSSP)
- K-core
- maximal cliques
- pagerank
- hyperlink-induced topic search (HITS)
- minimal spanning tree (MST)
- grid variant of Fruchterman-Reingold network layout algorithm

Commento: qui è praticamente tutto già fatto :/

[PAPER] Efficient graphs analysis with map reduce `http://www.umiacs.umd.edu/~jimmylin/publications/Lin_Schatz_MLG2010.pdf`