

# **SUBGRAPH EXPLORER**

Big Data Class — 2015 Contest

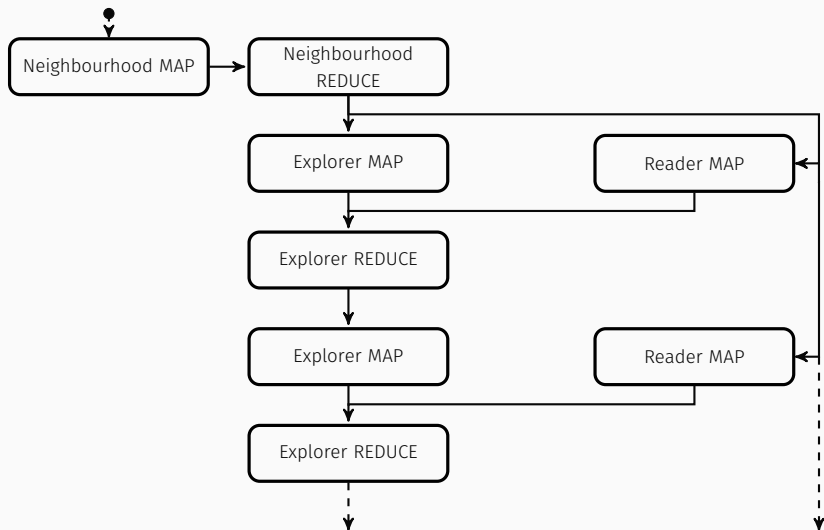
---

Adriano Di Luzio    Danilo Francati

May 25, 2015

Computer Science Department  
Sapienza University of Rome

# OUR JOBS' ARCHITECTURE



This is the first step of our algorithm.

---

### Neighbourhood builder Map-Reduce algorithm

---

```
function MAP(Node u, Node v)
    emit( $u, v$ )
    emit( $v, u$ )

function REDUCE(Node u, Node[]  $\Gamma(u)$ )
    emit ( $\{u : \Gamma(u)\}$ )
```

---

We simply build the neighbourhood of each node.

We defined  $\{u : \Gamma(u)\}$  as a special Hadoop data-type.

---

**Subgraph explorer** Map-Reduce algorithm

---

```
function MAPexplorer(Neighbourhood map)
   $v \leftarrow \text{mostPromisingNode}(\text{map})$ 
  emit( $v$ ,  $\text{map}$ )

function MAPreader( $\{u : \Gamma(u)\}$ )
  emit( $u$ ,  $\Gamma(u)$ )

function REDUCE(Node  $u$ , Neighbourhood[ ] neighbourhoods)
   $\text{newMap} \leftarrow \text{merge}(\text{neighbourhoods})$ 
  if  $\rho(\text{newMap}) \geq \hat{\rho}$  then
    output( $\text{newMap}$ )
  else
    emit( $\text{newMap}$ )
```

---

- Heuristic factor
- Neighbourhood filtering
- $K$ -Promising Nodes
- Best result selection
- Input pruning

## EXPERIMENTAL RESULTS

Graph	Maximum clique	$\rho$	$\hat{\rho}$	$t$
loc-gowalla	29	14	$\leq 14$	27''
web-BerkStan	201	100	$\leq 100$	8'15''
as-skitter	67	33	$\leq 33$	5'

Graph	Min. subgraph	$\rho$	$\hat{\rho}$	$t$
loc-gowalla	32	15.31	15	1'30''
loc-gowalla	51	20.50	20	2'
loc-gowalla	73	25.15	25	2'30''
web-BerkStan	392	103.40	100.5	8'40''
as-skitter	70	34.34	34	6'

Thanks for your attention!



<https://github.com/AldurD392/SubgraphExplorer>