

Graph Visualization With Neo4j Using Neovis.js.

Leveraging Graph Algorithms For Data Visualization

William Lyon

@lyonwj

lyonwj.com



Agenda

- *Neo4j Sandbox*
- *Run some graph algorithms*
 - Using *neo4j-graph-algorithms*
- *Visualize the results*
 - Using *neovis.js*
 - Embed visualization in a web page

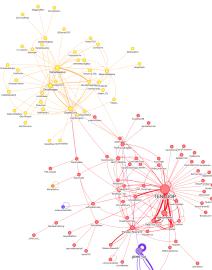
Russian Twitter Trolls Get Started Details Data Model Code Advanced -

Neo4j Browser: <https://10-0-1-179-34898.neo4jsandbox.com/>
Direct Neo4j HTTP: <http://34.224.3.24:34898/browser/>
Username: neo4j
Password: plastics-comparisons-gunner
IP Address: 34.224.3.24
HTTP Port: 34898
Bolt Port: 34897
Expires: 2 days, 23 hours, 59 minutes

+

```
function draw() {
    var config = {
        container_id: "viz",
        server_url: "bolt://localhost:7687",
        server_user: "neo4j",
        server_password: "letmein",
        directed: true,
        arrows: true,
        hierarchical: true,
        layout_type: "tree",
        method: "directed",
        labels: {
            "Character": {
                "name": "name",
                "size": "degree",
                "community": "community",
                "sizecypher": "MATCH (n) WHERE id(n) = {id} MATCH (n)-[r]-() RETURN sum(r.weight) AS c"
            }
        },
        relationships: {
            "INTERACTS": {
                "thickness": "weight",
                "caption": true
            }
        },
        initial_cypher: "MATCH (n:Character)-[r:INTERACTS]-(m:Character) WHERE EXISTS(n.degree) AND EXISTS(m.degree) RETURN n,r,m LIMIT 200"
    };
    viz = new Neovis(config);
    viz.render();
    console.log(viz);
}
```

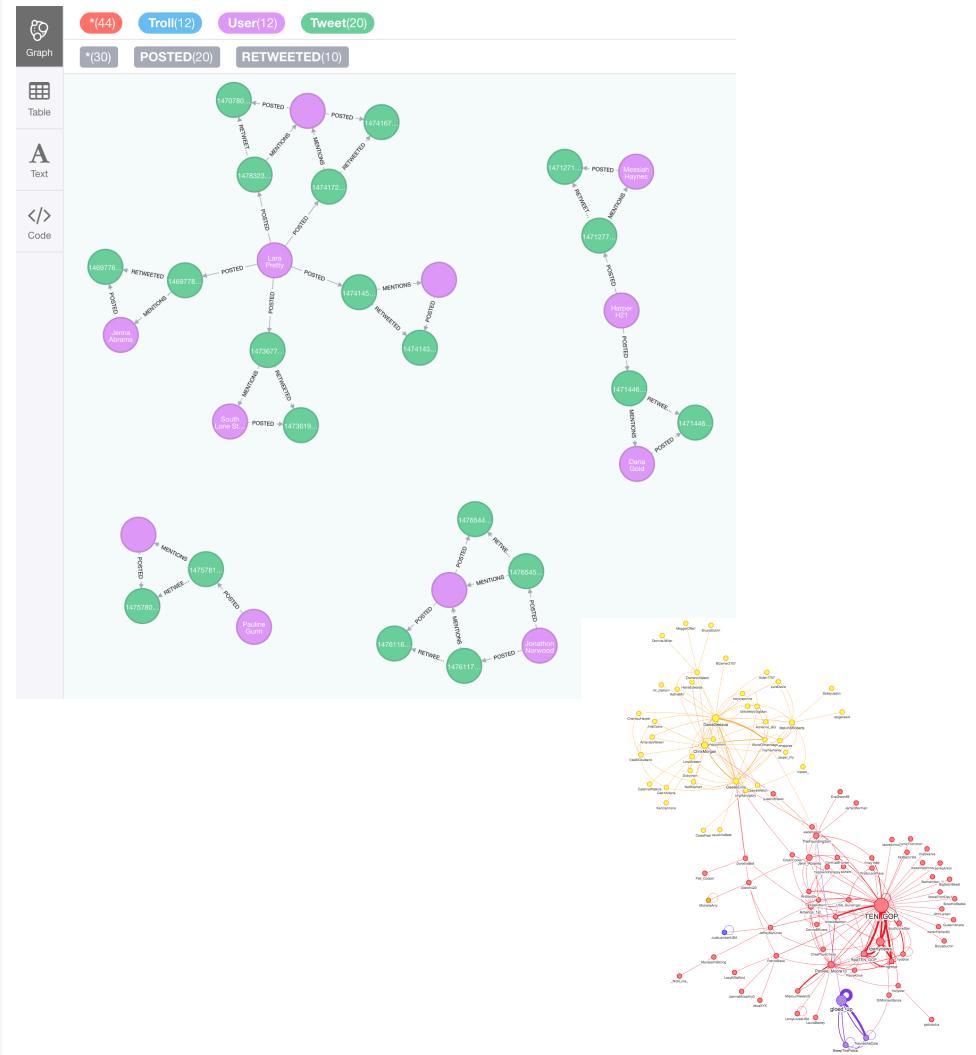
=



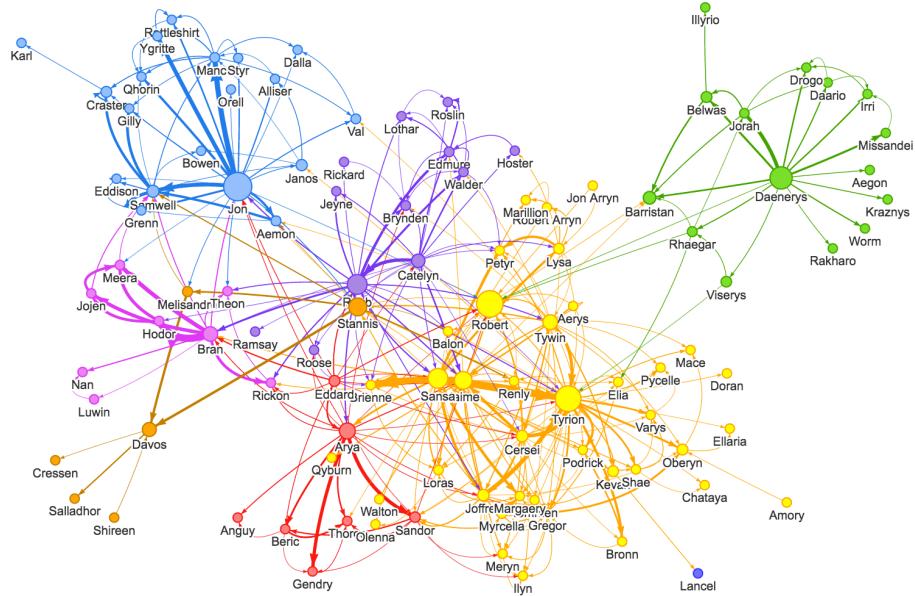
Goals Of Graph Visualization

- Explore the graph
- Summarize the data
- Show results of analysis
- Interactive
- Static
- Embed in (web) application
- Standalone application

```
1 MATCH p=(r1:Troll)-[:POSTED]->(:Tweet)<-[:RETWEETED]-(:Tweet)<-[:POSTED]-(r2:Troll)
2 RETURN p LIMIT 10
```



Neovis.js



```
function draw() {
  var config = {
    container_id: "viz",
    server_url: "bolt://localhost:7687",
    server_user: "neo4j",
    server_password: "letmein",
    arrows: true,
    hierarchical: true,
    //hierarchical_sort_method: "directed",
    tables: {
      "Character": {
        "caption": "name",
        "size": "degree",
        "community": "community",
        "sizeCypher": "MATCH (n) WHERE id(n) = {id} MATCH (n)-[r]-() RETURN sum(r.weight) AS c"
      }
    },
    relationships: {
      "INTERACTS": {
        "thickness": "weight",
        "caption": true
      }
    },
    initial_cypher: "MATCH (n:Character)-[r:INTERACTS]-(m:Character) WHERE EXISTS(n.degree) AND EXISTS(m.degree) RETURN n,r,m LIMIT 200"
  };
  viz = new NeoVis.default(config);
  viz.render();
  console.log(viz);
}
```

github.com/neo4j-contrib/neovis.js

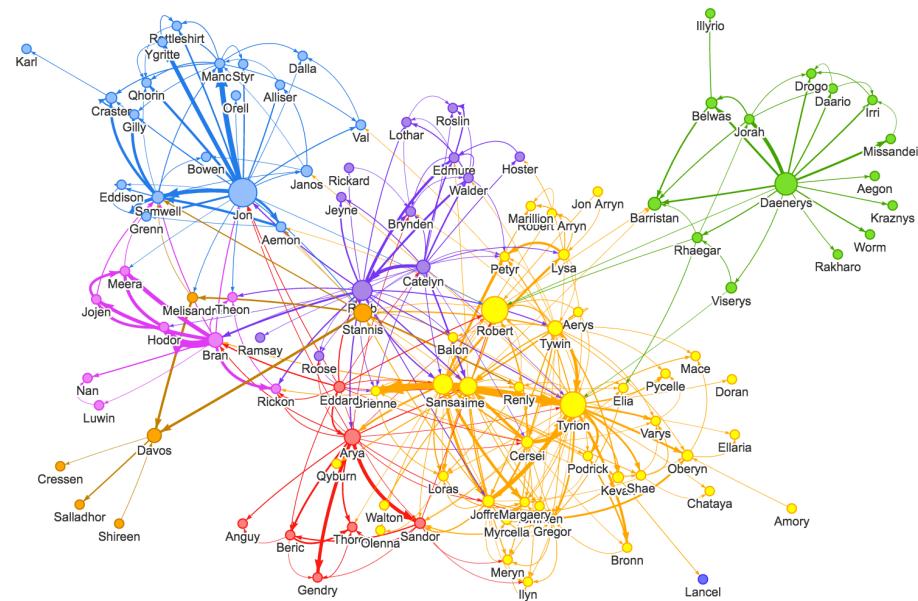


Graph Visualization + Graph Algorithms

Node size → Centrality
(degree, *PageRank*, *betweenness*)

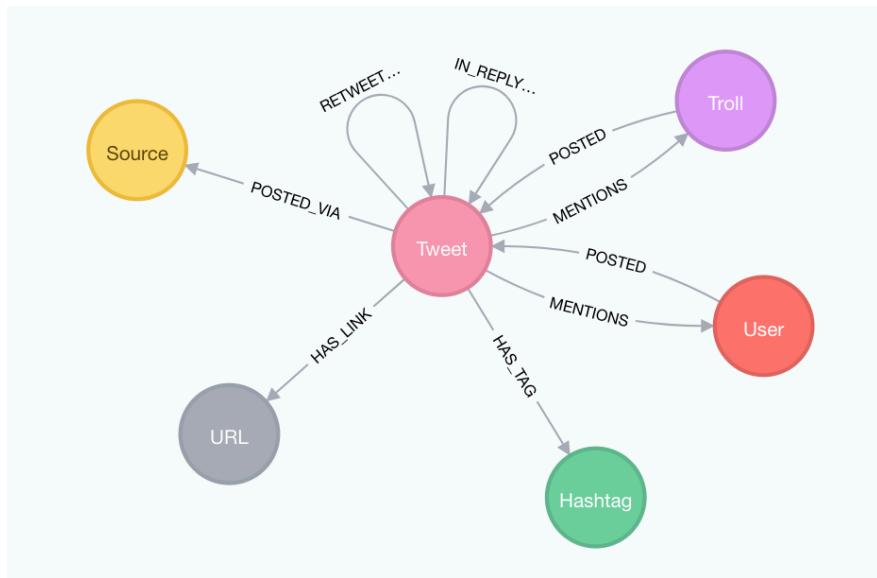
Color → Community
(*label propagation*, *union find*, *Louvain*)

Relationship Thickness → Weight
(*pathfinding*, A^* , *Dijkstra*)



Neo4j Sandbox

Russian Twitter Trolls



Russian Twitter Trolls Get Started Details Data Model Code Advanced ▾

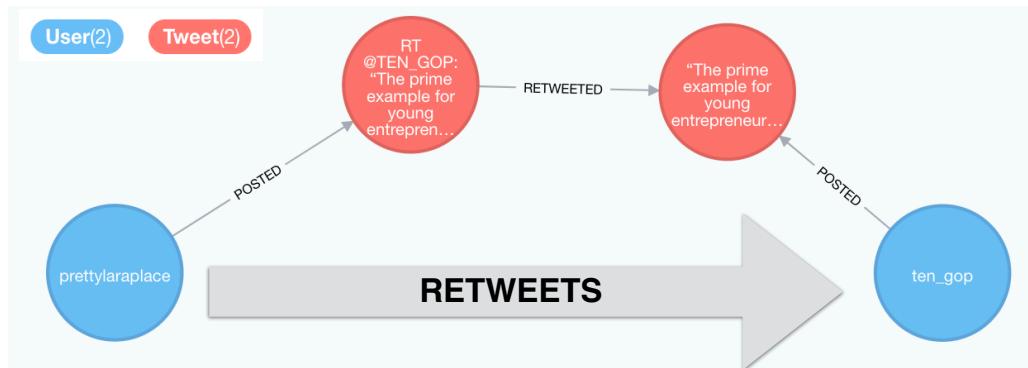
Neo4j Browser: <https://10-0-1-179-34898.neo4jsandbox.com/>
Direct Neo4j HTTP: <http://34.224.3.24:34898/browser/>

Username: neo4j
Password: plastics-comparisons-gunnyery

IP Address: 34.224.3.24
HTTP Port: 34898
Bolt Port: 34897

Expires: 2 days, 23 hours, 59 minutes

Twitter Troll Retweet Graph



(:User)-[:RETWEETS]->(:User)

Resources

Neo4j Sandbox

neo4jsandbox.com/

Neovis.js GitHub page

github.com/neo4j-contrib/neovis.js

Neo4j Graph Visualization Developer Page

neo4j.com/developer/guide-data-visualization/

Neo4j Graph Algorithms

neo4j.com/developer/graph-algorithms/





(you)-[:HAVE]->(questions)<-[:ANSWERS]-(will)

