Interesting Content and User Discovery without Upvotes

An application of centrality measures to Reddit

Rollen S. D'Souza
rollen.dsouza@uwaterloo.ca

University of Waterloo

2018

Outline

1. Introduction
   - About Reddit
   - Mathematical Preliminaries
2. Analysis
   - Technical Details

What is Reddit?

Purposes

- Social news aggregation

What is Reddit?

Purposes

- Social news aggregation
- Discussion platform

What is Reddit?

Purposes

- Social news aggregation
- Discussion platform
- Community for help or relaxation

What is Reddit?

Basic Structure

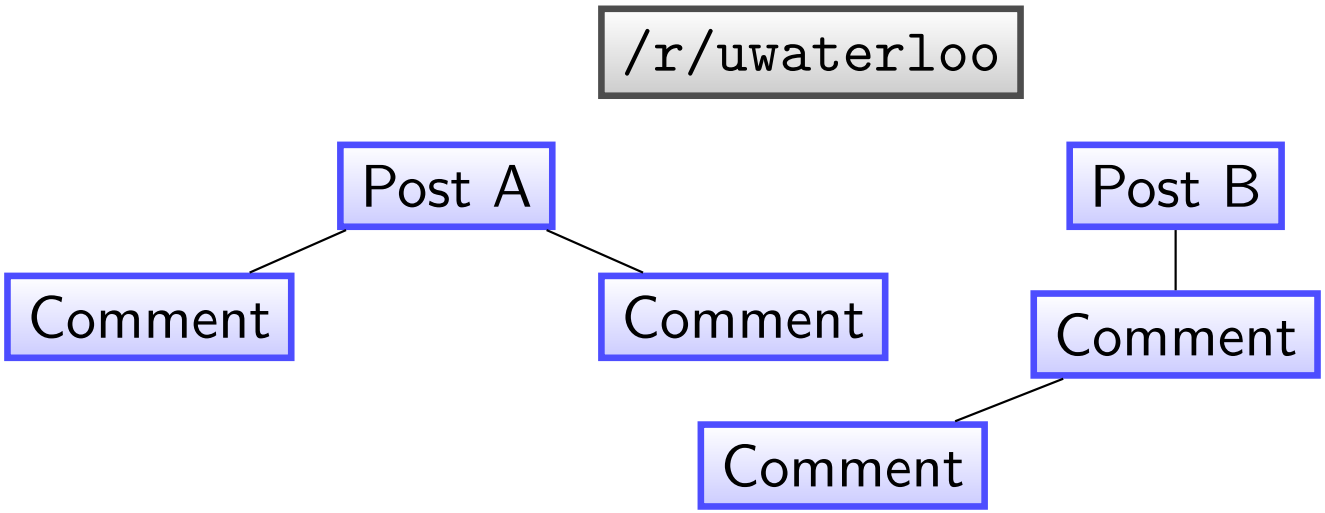
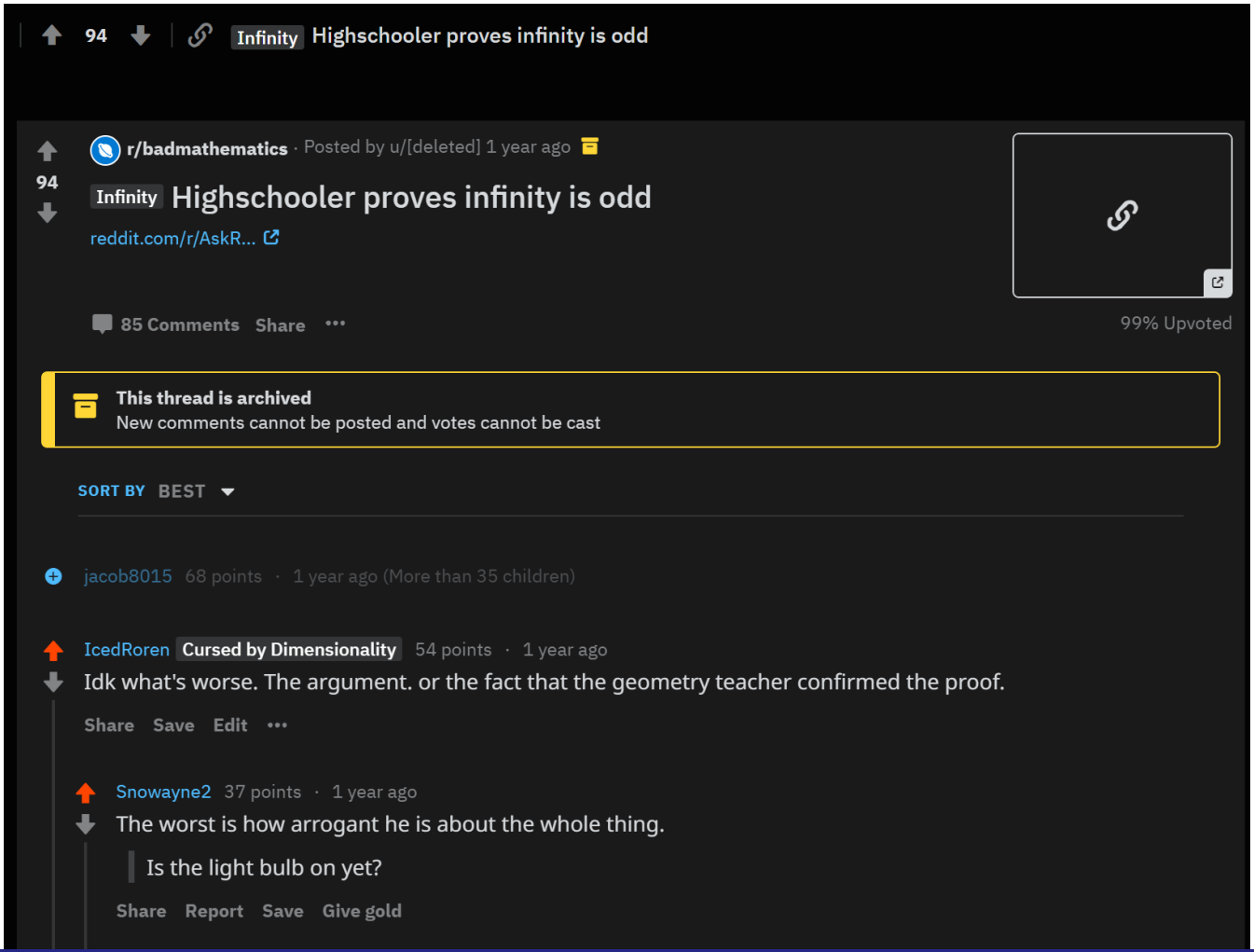


Figure: Graph-like structure of the Reddit platform.

What is Reddit?



Mathematical Preliminaries

What I assume:

- Centrality Measures
- Properties of Laplacians

The former is how I sort through content and users.

Katz Centrality Measure

Definition

The *Katz Centrality Measure* for a node i in graph G with adjacency matrix A is defined as,

$$c_i = \sum_{k=1}^{\infty} \sum_{j=1}^{n} \alpha^k \left(A_{j,i}\right)^k,$$

where $0 < \alpha \leq \rho(A)$.

Katz Centrality Measure

Why this measure?

- Has larger reach in the graph (looks at all paths to a node.)
- Easy to compute.
- Loose requirements on A.
- Intuitive meaning.

On Laplacians

Laplacians can be used for clustering. Recall,

$$L = D_{out} - A.$$

Use eigenvalues near 0 for clustering (eigenvectors associated with nearly disconnected components).

On Laplacians

Laplacians can be used for clustering. Recall,

$$L = D_{out} - A.$$

Use eigenvalues near 0 for clustering (eigenvectors associated with nearly disconnected components).
So...finding eigenvalues is easy right?

On Laplacians

Different Formulation

Problem: The graph can have as many as 10^4 nodes!

On Laplacians

Different Formulation

Problem: The graph can have as many as 10^4 nodes!

Definition

The *normalized Laplacian* of a graph is defined as,

$$L_{rw} = I_{n \times n} - D_{out}^{-1} A$$

On Laplacians

Different Formulation

Problem: The graph can have as many as 10^4 nodes!

Definition

The *normalized Laplacian* of a graph is defined as,

$$L_{rw} = I_{n \times n} - D_{out}^{-1} A$$

Need to ensure the graph has no sinks!

Technical Details

The Code

Code was implemented using `Python 3` and `MATLAB`. Code is available on my Github. Other than that,

- `redis` for storing data in-memory.
- `numpy`,`scipy` for computation outside of `MATLAB`.

Technical Details

The Data

Sourced thanks to Jason Baumgartner who uploaded data dumps of the JSON objects gathered from the Reddit API. Structures contain,

- Submissions with author (id and flair), tags, score, text or link,
- Comments with author (id and flair), text
- and much much more!