

# Similarity Search: a Web Perspective

Yury Lifshits

Caltech

<http://yury.name>

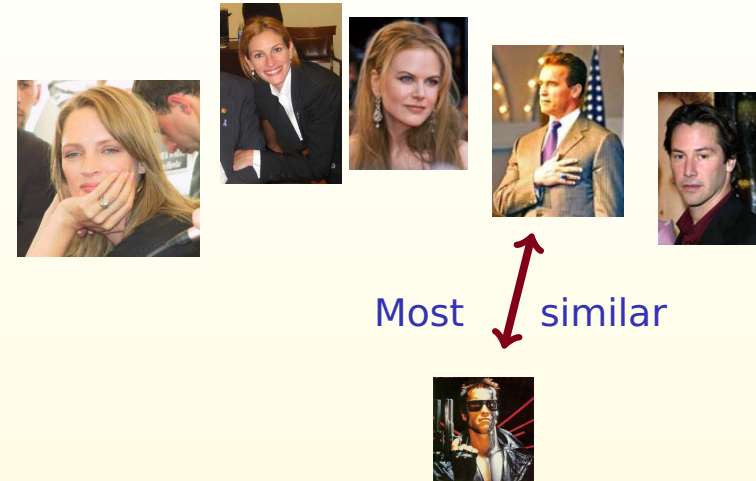


CMI Retreat

9 October 2007

1 / 15

## Similarity Search: An Example



2 / 15

## Outline



New math problems, algorithms and experiments

**1**

Applications

**2**

Current State

**3**

Problem List

3 / 15

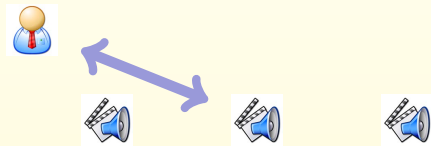
**1**

Similarity Search  
in Web Technologies

4 / 15

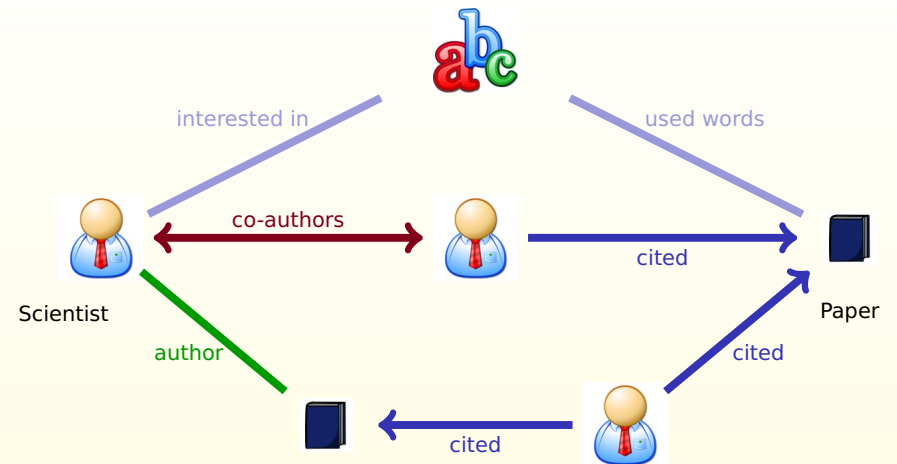
## Similarity Search vs. Web

- Recommendations (movies, books...)
- Item-item recommendations
- News aggregation
- Ad targeting
- “Best match” search: resume, job, BF/GF, car, apartment



5 / 15

## Similarity Chart

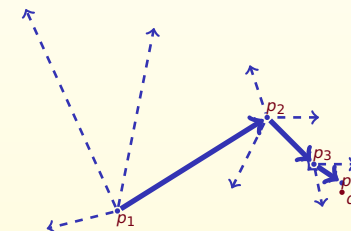


**High similarity:** many chains, short chains, heavy chains

6 / 15

## Recent Results

- Similarity search without triangle inequality  
joint work with Navin Goyal and Hinrich Schütze
- Similarity search for “random texts”  
joint work with Benjamin Hoffmann and Dirk Nowotka
- Least squares for sparse matrices  
joint work with Dirk Nowotka
- Improving Viterbi algorithm for HMM  
joint work Shay Mozes, Oren Weimann, and Michal Ziv-Ukelson



8 / 15

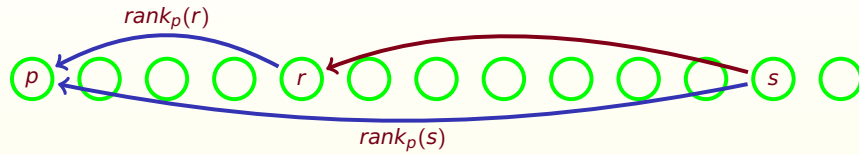
## 2

Current State of  
My Research

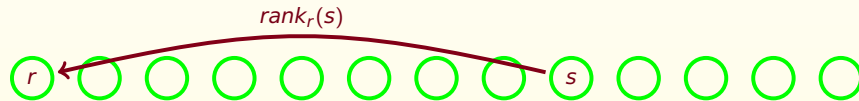
7 / 15

# Concept of Disorder

Sort all objects by their similarity to  $p$ :



Then by similarity to  $r$ :



Dataset has disorder  $D$  if

$$\forall p, r, s: \text{rank}_r(s) \leq D(\text{rank}_p(r) + \text{rank}_p(s))$$

There is similarity search solution with roughly  $\mathcal{O}(Dn \log n)$  data structure and  $\mathcal{O}(D \log n)$  search time

# Other Related Stuff

- **Yandex datasets**: on-line advertising logs, friendship graph
- <http://simsearch.yury.name>  
Bibliography, researchers, links, open problems

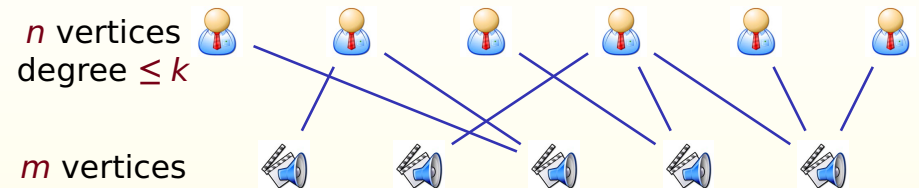


- **Algorithmic Problems Around the Web**  
CS101.2, MW 11:00-11:55, Jorgensen 287
- **Nearest Neighbors** Tutorial
- Mini-course **A Guide to Web Research**

# 3

My Problem List

# Similarity Search in Bipartite Graphs

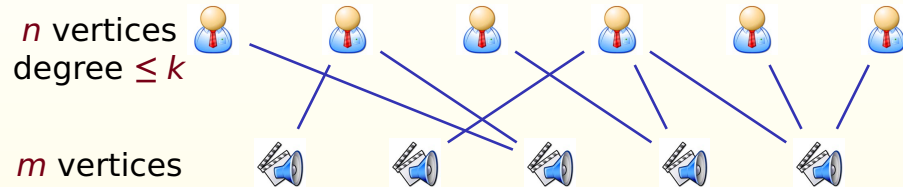


Person-person similarity: # 2-step paths  
Person-movie similarity: # 3-step paths

## Constraints:

$\text{poly}(m, n)$  for preprocessing  
 $\text{poly}(k, \log n, \log m)$  for query processing

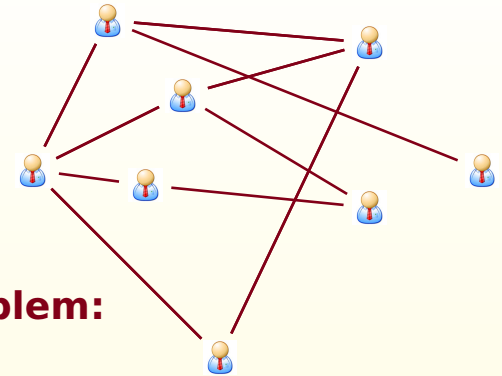
## Clustering in Bipartite Graphs



**$(\alpha, \beta)$ -clustering** for movies:  
Every cluster has size at most  $\alpha$   
For every user all his choices are covered  
by at most  $\beta$  clusters

13/15

## Visualizing Social Networks



### Optimization problem:

To map people  
(collisions forbidden)  
to 2-dimensional grid  
minimizing the sum

$$\sum_{p,q \text{ are friends}} |M(p) - M(q)|^2$$

14/15

Thanks for your attention!

Questions?

15/15