Web Mining: Blogspace and Folksonomies

A Guide to Web Research: Lecture 3

Yury Lifshits

Steklov Institute of Mathematics at St.Petersburg

Stuttgart, Spring 2007

Talk Objective

Today:

- Short description of technology
- Technological challenges
- Algorithmic problems

To do:

- Adding assumptions to the problems
- Constructing (approximate) algorithms

Outline

1. Introduction to Blogspace
2. Introduction to Folksonomies
3. Algorithmic Challenges
   - Personal News Aggregation
   - Structure Discovery in Folksonomies

Part I Blogspace

What is blogspace?

What related technologies are supposed to appear in nearest future?

Blogspace: Overview

Blogspace (Blogosphere) is a set of all weblogs

Every blog consists of:

- Profile
- Posts: title, content, time-stamp, comments, tags
- Subscribers

Prominent technologies in the field:
Blogger, Livejournal, Wordpress, Technorati

Technological Challenges in Blogspace

- Blog search and blog ranking
- Personal newspaper: every user every day receives personal digest of all posts in the world
- Advertising in blogspace, in particular, understanding information propagation in blogspace
- Tracking blogspace reflections of real life events

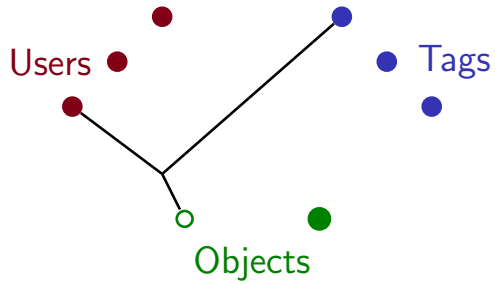
Part II
Folksonomies

What is folksonomy?

What related technologies are supposed to appear in nearest future?

Folksonomy: Overview

Folksonomy is a set of triples $< user, object, tag >$
Primary purpose: memory assistance



Prominent technologies in the field:
Del.icio.us, Flickr.com, tags in blogspace, GMail labels

Technological Challenges in Folksonomies

- Tag-based file system
- Utilizing folksonomies in web search
- Tag subscriptions and other folksonomy-based recommendations
- Second layer challenge: discover and visualize relations between tags
- Automatic labelling

Part III
Algorithmic Challenges

Personal news aggregation

Structure discovery in folksonomies

Personal News Aggregation: Informally

Personal news aggregation:

Every user has a preference profile:
specified information sources, keywords,
tags(topics), popularity,
references to the preferences of others

Every news item has its own description:
text, votes and recommendations, tags,
author reputation, comments

All-to-all filtering:

To find, say, ten most appropriate news items for every user

Personal News Aggregation: Solutions

Personalized news delivery:

Google News
Google Reader
Bloglines
Livejournal Friends
Feedburner

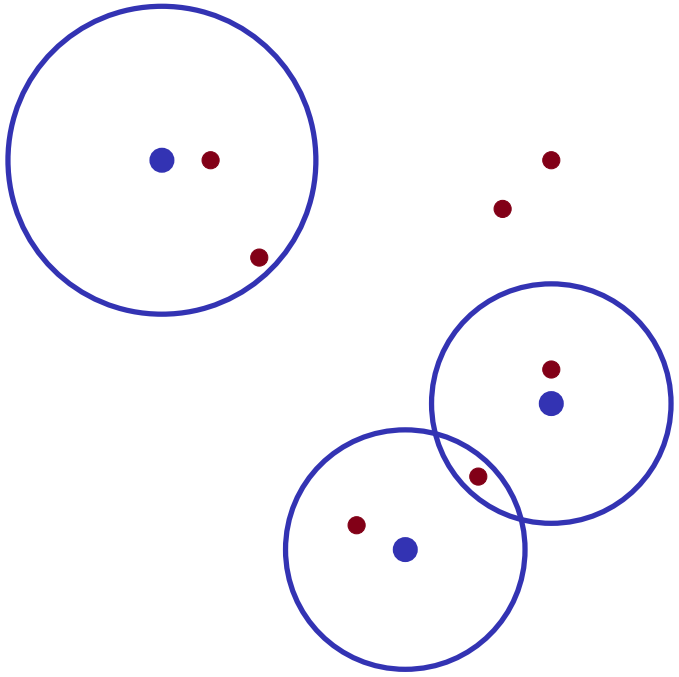
Formalization

- Every news is represented by a sparse vector
- Every user profile is represented by a sparse vector
- Similarity is defined as cosine between two vectors
- **Simplification:** 0/1 vectors,
  similarity proportional to the size of intersection

Large Scale All-to-All Nearest Neighbors

- N **blue** vectors in d-dimensional space, every vector has at most k nonzero components
- M **red** vectors in d-dimensional space, every vector has at most k nonzero components
- To find 10 nearest (according to cosine similarity) **red** vectors to every **blue** vector
- Desired time complexity

$$(N + M)\, polylog(N + M)\, poly(k)$$



All-to-All Nearest Neighbors in Set Notation

- N **blue** sets, each of size at most k
- M **red** sets, each of size at most k
- To find 10 nearest (according to intersection-size similarity) **red** sets to every **blue** one in time

$$(N + M)\, polylog(N + M)\, poly(k)$$

Structure Discovery in Folksonomies

Problem: finding similar tags in folksonomy

Evidences of similarity:

- Inner co-occurrence: some user applied both tags to some object
- Outer co-occurrence: one user applied the first tag, another user applied the second tag to the same object

Tag similarity

Projection to bipartite graph:

Removing users from folksonomy

Notation: $Q(t)$ is the set of all objects tagged by t

Three formulas for tag similarity:

$$Sim(t_1, t_2) = |Q(t_1) \cap Q(t_2)|$$

$$Sim(t_1, t_2) = \frac{|Q(t_1) \cap Q(t_2)|}{|Q(t_1)| + |Q(t_2)|}$$

$$Sim(t_1, t_2) = \frac{|Q(t_1) \cap Q(t_2)|}{\min\{|Q(t_1)|, |Q(t_2)|\}}$$

Discovering Related Tags

- Bipartite graph tags-objects, F edges
- Task 1: for every tag find 10 nearest tags
- Task 2: for a given α find all tag pairs that have similarity above α
- Desired time complexity: $F \cdot polylog(F)$

Discussion

Which of these two problems do you like more?

Changes to presented formalization?

Ideas and approaches?

Relevant work?

Call for participation

Know a relevant reference?
Have an idea?
Find a mistake?
Solved one of these problems?

- Knock to my office 1.156
- Write to me `yura@logic.pdmi.ras.ru`
- Join our informal discussions
- Participate in writing a follow-up paper

Highlights

Three problems to think about:

- All-to-all nearest neighbors in sparse vector model
- All-to-all nearest neighbors in set notation with intersection-size similarity
- Finding all over-threshold similarities between tags in folksonomy

Vielen Dank für Ihre Aufmerksamkeit!
Fragen?

References (1/2)

Course homepage

http://logic.pdmi.ras.ru/~yura/webguide.html

R. Kumar, J. Novak, P. Raghavan, A. Tomkins

On the Bursty Evolution of Blogspace

http://cui.unige.ch/tcs/cours/algoweb/2005/articles/p568-kumar.pdf

D. Gruhl, R. Guha, D. Liben-Nowell, A. Tomkins

Information diffusion through blogspace

http://wwwconf.ecs.soton.ac.uk/archive/00000597/01/p491-gruhl.pdf

E. Adar, L.A. Adamic

Tracking Information Epidemics in Blogspace

http://www.hpl.hp.com/research/idl/papers/blogs2/trackingblogepidemics.pdf

References (2/2)

A. Mathes

Folksonomies-Cooperative Classification and Communication Through Shared Metadata

http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.pdf

A. Hotho, R. Jaschke, C. Schmitz, G. Stumme

Information retrieval in folksonomies: Search and ranking

http://www.kde.cs.uni-kassel.de/stumme/papers/2006/hotho2006information.pdf