

Estimating number of citations using author reputation

Aris Gionis, Carlos Castillo, Debora Donato

Yahoo! Research Barcelona

Complex Networks - Pula
July 5, 2007

Collaborative Content Systems

Social Media

- On-line forums
- Web-logs
- Photo or video sharing communities
- Question-answering portals
- Social bookmarking sites
- Wikis

Goal

Finding *high-quality* items *automatically* in such large systems

Main issues

Main tasks

- Ranking** sorting item and finding the top high-quality or bottom low-quality items
- Similarity** finding users with similar interest, finding related content items, etc.

Main Steps

- 1 Data Modelling
- 2 Data processing
- 3 Characterization
- 4 Feature extraction
- 5 Testing

Data Modelling

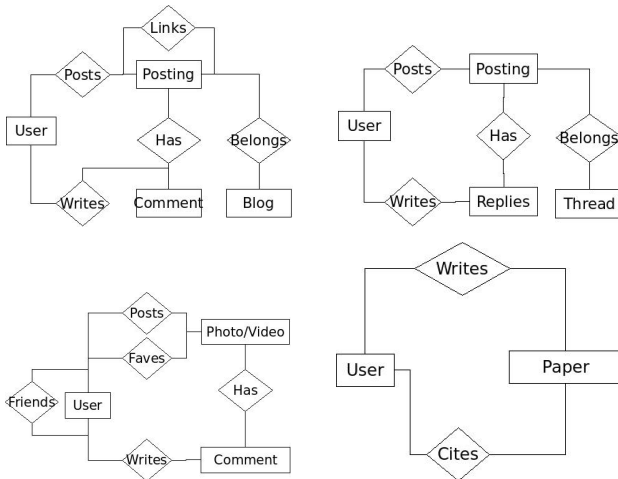


Figure: Simplified entity-relationship diagrams: Weblogs, On-line Forum, Media Sharing, Citations

Predicting popularity

- Dynamic environment in which new items are published
- Items are published by “authors”
- Authors provide feedback to other authors' items
- Feedback can be either explicit or implicit
positive or negative vote, link, citation
- Natural notion of successful items
- Question: Can we predict which items will be successful?

Application I – Photo sharing


Applications Places System 9:55 PM

Casa Batllo - Antoni Gaudi on Flickr - Photo Sharing! - Mozilla Firefox

File Edit View Go Bookmarks Tools Help None

http://www.flickr.com/photos/arutha/277837378/ Go

Casa Batllo - Antoni Gaudi



Uploaded on October 24, 2006 by [arutha](#)

arutha's photostream
845 photos

This photo also belongs to:

My Faves! (Set)
292 photos

Barcelona (Set)
165 photos

HDR (Set)
56 photos

For more photos from Barcelona, check out my [Barcelona Set](#).

Comments

[Annuska Hjärta](#) pro says:
just perfectly beautifull
Posted 7 months ago. ([permalink](#))

[arutha](#) pro says:
Tx Annuska... it's not me, it's Gaudi! :)

Find: Find Next Find Previous Highlight all Match case

Casa Batllo - Antoni Gaudi on Flickr - Pho... [Evolution - INBOX (61 total)] Starting Take Screenshot

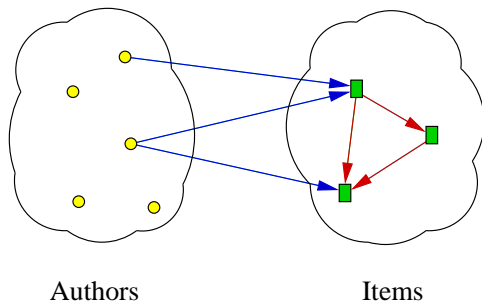
Application I – Photo sharing

- Flickr
- Users (authors)
 - upload photos
 - tag photos
 - comment on photos
 - mark favorites
 - create friendship links
 - form an online community
- Can we predict the popularity of a newly uploaded photo?
- e.g., estimate the number of “favorites” in the next few months

Application II – Academic bibliography

- Database of scientific articles, e.g., CiteSeer
- Authors publish papers
- Existing papers accumulate reputation by citations
- Can we predict the popularity of a newly published paper?
- e.g., estimate the number of citations after a few years

The abstract graph model



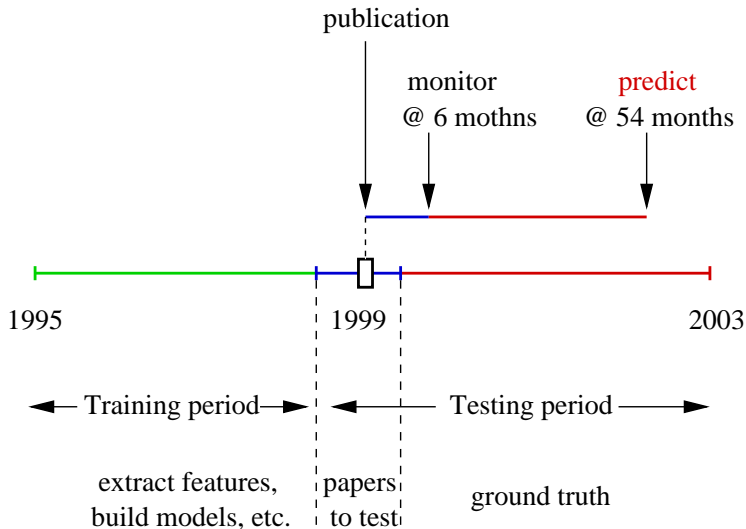
Other information:

- content of items
- a social network on authors

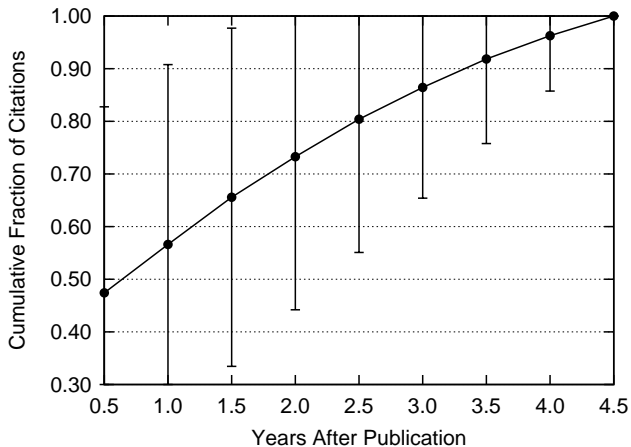
The dataset

- CiteSeer database of scientific articles
- <http://citeseer.ist.psu.edu/>
- 581 866 papers published from 1995 to 2003 (inclusive)
- Keep only papers for which at least one of the authors had three papers or more in the dataset
- Prune 11% of the dataset

The prediction task

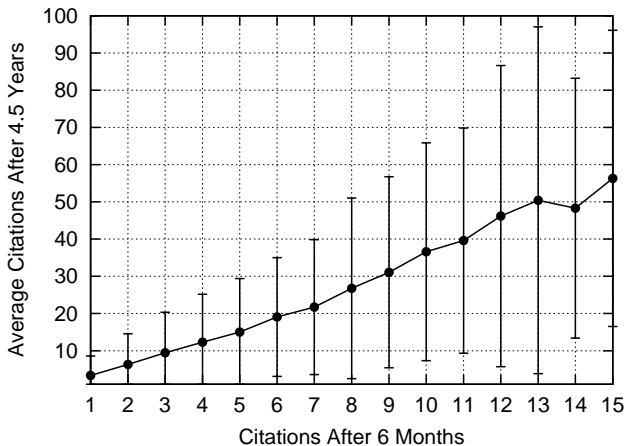


The challenges – Large variance



Cumulative fraction of citations over time

The challenges – Large variance



Citations at 6 months vs. average citations at 54 months

The baseline

- Citations at 6 months and citations at 54 months have correlation coefficient 0.57
- Can be a basis for a prediction, but not so accurate
- How to improve it?

What is missing

- Past information about the authors
- Exploiting the network structure:
- Good authors tend to write good papers
- Good authors tend to cite good papers
- Papers written and cited by good authors tend to be successful

Machine learning approach

- Extract a set of features and use it to build a better model

Author-based features

For each author compute:

- Total number of citations received
- Total number of papers (co)authored
- Average number of citations per paper
- Total number of co-authors
- Average number of co-authors per paper
- ...

For each paper compute:

- aggregate of the features of its authors
(using `sum`, `avg`, `max`)

Link-based features

- EigenRumor algorithm [Fujimura and Tanimoto, 2005]
- Inspired by HITS [Kleinberg, 1999]

Eigenrumor algorithm

- P : *provision matrix* (authors \times papers)
 $P_{ij} = 1$ if author i has provided paper j and 0 otherwise
- E : *evaluation matrix* (authors \times papers)
 $E_{ij} = 1$ if author i has evaluated paper j and 0 otherwise
- \mathbf{r} : *reputation* scores of papers
- \mathbf{a} : *authority* scores of authors
- \mathbf{h} : *hub* scores of authors

Eigenrumor algorithm

- High-reputation papers are written by high-authority authors and cited by high-hub authors
- High-authority authors write high-reputation papers
- High-hub authors cite high-reputation papers
- In equations

$$\mathbf{r} = \alpha P^T \mathbf{a} + (1 - \alpha) E^T \mathbf{h}$$

$$\mathbf{a} = P \mathbf{r}$$

$$\mathbf{h} = E \mathbf{r}$$

Link-based features

For each author compute:

- Authority score
- Hub score

For each paper compute:

- Reputation score
- Aggregate of authority score and hub score of its authors
(using sum, avg, max)

Prediction tasks

- 1 Regression: predict the number of citations of a paper
- 2 Classification: predict if a paper will be *successful*
(defined as being in the top 10%)

The evaluation measures

Regression

- Correlation coefficient

$$r = \frac{E[XY] - E[X]E[Y]}{\sqrt{\text{Var}[X]}\sqrt{\text{Var}[Y]}}$$

Classification

- Recall: R
- False positive rate: P
- F-measure: $F = 2\frac{PR}{P+R}$

Results

Effect of monitoring period

<i>A posteriori</i> citations	Predicting Citations <i>r</i>	Predicting Success <i>F</i>
6 months	0.57	0.15
1.0 year	0.76	0.54
1.5 years	0.87	0.63
2.0 years	0.92	0.71
2.5 years	0.95	0.76
3.0 years	0.97	0.86
3.5 years	0.99	0.91
4.0 years	0.99	0.95

Results

Effect of different type of features

<i>A priori</i> features	<i>A posteriori</i> features			
	First 6 months		First 12 months	
	<i>r</i>	<i>F</i>	<i>r</i>	<i>F</i>
None	0.57	0.15	0.76	0.54
Author-based	0.78	0.47	0.84	0.54
Hubs/Auth	0.69	0.39	0.80	0.54
Host	0.62	0.46	0.77	0.57
EigenRumor	0.74	0.55	0.83	0.64
ALL	0.81	0.55	0.86	0.62

Conclusions

- Predicting reputation as a link-analysis task
- Can we improve performance?
- Can we solve the problem in more “noisy” environments?



Fujimura, K. and Tanimoto, N. (2005).

The EigenRumor Algorithm for Calculating Contributions in Cyberspace Communities.



Kleinberg, J. M. (1999).

Authoritative sources in a hyperlinked environment.

Journal of the ACM, 46(5):604–632.