CSE 8803RS: Recommendation Systems

Lecture 1: Introduction

Hongyuan Zha

School of Computational Science & Engineering
College of Computing
Georgia Institute of Technology

What are Recommender Systems?

- Recommender systems are software applications that aim to support users in their decision-making while interacting with large information spaces
- They recommend items of interest to users based on their preferences
  — preferences: explicitly or implicitly
  — goals: relevance, novelty, surprise
- New paradigm of search: interesting items find the user instead of the user explicitly searching for them

A Bigger Context: Socially Intelligent Computing

Social contexts in computing

- Wikepedia: massive numbers of Internet-based volunteer communities collaboratively write encyclopedia articles of unprecedented scope and scale
- Open-source software
- Collaborative problem solving: perform massive, complex computations that exploit the unused power of millions of computers worldwide
- Online marketplaces: collective behaviors of their participants
  — vast storehouses of consumer-supplied reviews
  — recommend products by matching a consumer's shopping behavior with those other customers with similar behaviors
  — set marketplace prices via computationally mediated auctions

Socially Intelligent Computing: Examples

- Search engines: hyperlinks and user clicks
- Politics: political movements are creating new forms of engagement and collective action in political systems worldwide
- Online games and virtual worlds: millions of people who have never met work together in teams to develop and execute complex activities in online games and virtual worlds

Socially Intelligent Computing: Definition

New, emergent behaviors that arise out of the complex and dynamic interactions among people and computers.

- Social: the interactions among people and increasingly more sophisticated computing technologies
- Intelligent: the emerging intelligence exhibited by such systems as well as their increasing knowledge about people and their interactions with one another and with computers
- Computing: the computation technologies that act as mediators among people, as tools used by people, and as equal or complementary participants with people

Socially Intelligent Computing: Key Questions

Rethink key questions as fundamental as "What is intelligence?" and "What is computable?"

- Can we understand how such systems give rise to emergent behaviors?
- What values do they embody and what affordances do they provide?
- How do we create systems that by design harness the essential characteristics of both people and computers to achieve our ambitions and embody desired behaviors?

Socially Intelligent Computing: Goals

Characterizing, understanding, and eventually designing for desired behaviors arising from computationally mediated groups of people at all scales

- New forms of knowledge creation
- New models of computation
- New forms of culture
- New types of interaction

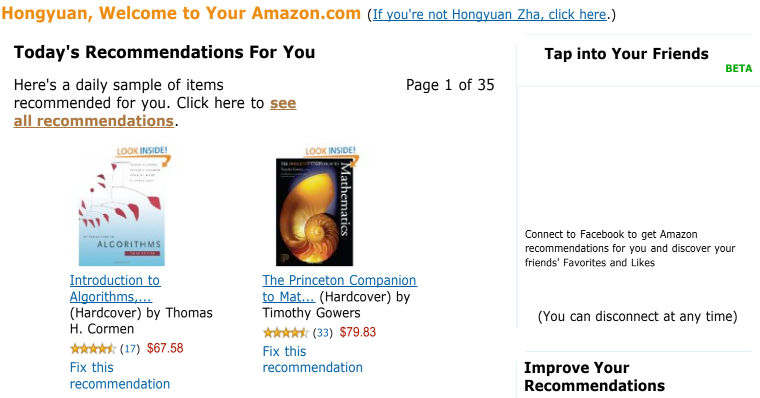
The investigation of such systems and their emergent behaviors and desired properties will inform the design of future systems.

Importance of Recommendation

- Recommendations from other people by spoken words, reference letters, news reports from news media, general surveys, travel guides
- Recommender systems assist and augment this natural social process to help people making choices
- Basic insights: personal tastes/preferences are correlated
  If `A` and `B` both like X, and `A` also likes Y then `B` is more likely to like Y — especially if `B` is friend of `A`

Product Recommendation

- Examples: Amazon, Netflix



- Task: Find a list of products that the user is likely to buy based on past purchase history

The Netflix Challenge

- \$1M prize competition
- Input: large dataset
  — 480K viewers, 18K movies, 1.2% entries observed
- Goal: improve root mean square prediction error rate of 10% compare to Netflix in-house algorithm
- 40000+ teams from 186 countries (5000+ teams with valid submissions)
- Begins October 2006, winners in June 2009

Netflix Cancels the Netflix Prize 2

- Planned: customers' gender, ages, ZIP codes and previously rented movies
- Paul Ohm: gender + ZIP code + birthdate uniquely identifies a significant percentage of Americans
  — 87% according to Latanya Sweeney's study
- Possible to identify users by comparing
  — reviews of obscure movies on Netflix
  — reviews on Imdb.com that were published under screennames.
- Netflix sued 12/2009
  — Closeted lesbian does want it knwon that she had rented a number of "gay-themed" movies from Netflix

Document Recommendation

- Examples: Google Scholar's "Related articles" feature

Google scholar | matrix computations | Search | Advanced Scholar Search | Scholar Preferences

Scholar | Articles and patents | anytime | include citations | Results **1** - **10** of about **1,280,000**. (**0.15** sec)

[BOOK] **Matrix computations**
GH Golub, CF Van Loan - 1996 - books.google.com
©1983, 1989, 1996 The Johns Hopkins University Press All rights reserved. Published 1996 Printed in the United States of America on acid-free paper 9876 First edition 1983 Second edition 1989 Third edition 1996 The Johns Hopkins University Press 2715 North Charles Street **...**
Cited by 29117 - Related articles - Library Search - All 12 versions

[PDF] SPARSKIT: A basic tool kit for sparse **matrix computation** psu.edu [PD
Y Saad - 1994 - Citeseer
Abstract. This paper presents the main features of a tool package for manipulating and working with sparse matrices. One of the goals of the package is to provide basic tools to facilitate exchange of software and data between researchers in sparse **matrix computations**. Our starting **...**
Cited by 576 - Related articles - View as HTML - All 18 versions

[BOOK] Fundamentals of **matrix computations**
DS Watkins - 2002 - books.google.com
This text is printed on acid-free paper. © Copyright © 2002 by John Wiley & Sons, Inc., New York. All rights reserved. Published simultaneously in Canada. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, **...**
Cited by 542 - Related articles - All 15 versions

[CITATION] **Matrix computations**. Johns Hopkins Studies in the Mathematical …
GH Golub, CF Van Loan - Johns Hopkins University Press, 1996
Cited by 445 - Related articles

- Task: Find a list of documents that are *related* to the current

Recommender Systems for Documents

Google scholar [Search] Advanced Scholar Search
Scholar Preferences

Scholar Results **1** - **10** of about **101** related to **Golub: Matrix computations**. (**0.10** sec)

[BOOK] Matrix computations
GH Golub, CF Van Loan - 1996 - books.google.com
©1983, 1989, 1996 The Johns Hopkins University Press All rights reserved. Published 1996 Printed in the United States of America on acid-free paper 9876 First edition 1983 Second edition 1989 Third edition 1996 The Johns Hopkins University Press 2715 North Charles Street **...**
Cited by 29117 - Related articles - Library Search - All 12 versions

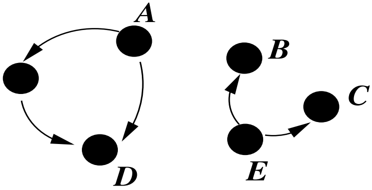
[BOOK] The algebraic eigenvalue problem
JH Wilkinson - 1988 - books.google.com
NUMERICAL MATHEMATICS AND SCIENTIFIC COMPUTATION *P. Dierckx: Curve and surface fittings with splines *H. Wilkinson: The algebraic eigenvalue problem *I. Duff, A. Erisman, and J. Reid: Direct methods for sparse matrices *JD Pryce: Numerical solution of **...**
Cited by 5961 - Related articles - Library Search - All 10 versions

[BOOK] The symmetric eigenvalue problem
BN Parlett - 1998 - books.google.com
SIAM's Classics in Applied Mathematics series consists of books that were previously allowed to go out of print. These books are republished by SIAM as a professional service because they continue to be important resources for mathematical scientists. Editor-in-Chief Robert E. O' **...**
Cited by 2570 - Related articles - All 11 versions

[BOOK] LAPACK Users' guide
E Anderson, Z Bai, C Bischof, S Blackford, J Demmel, J … - 1999 - books.google.com
7/lie 5etie5 inclucles li2siclbOo!<5 2lic^ 5aftwsse guicles 25 well 25 mcoioßtsplis on plActica! implementÄtion c>f computatioiiZ! metnc>6z, environments, Znci tc»c>>z. I'lie focus >5 c»n M2><inß tecent Developments 2V2i!Z,ole in 2 ps2ctic2> fosMÄt tc» seseÄrcliesz **...**
Cited by 3604 - Related articles - Library Search - All 8 versions

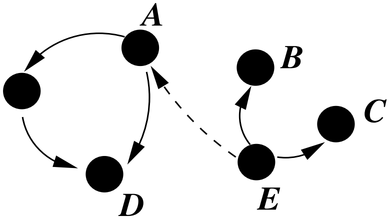
[BOOK] Matrix analysis
RA Horn, CR Johnson - 1990 - books.google.com
Contents Preface Chapter 0 Review and miscellanea 0.0 Introduction 0.1 Vector spaces 0.2 Matrices 0.3 Determinants 0.4 Rank 0.5 Nonsingularity 0.6 The usual inner product 0.7 Partitioned matrices 0.8 Determinants again 0.9 Special types of matrices 0.10 Change of basis **...**
Cited by 10430 - Related articles - Library Search - All 5 versions

Recommender Systems Based on Citation Graphs

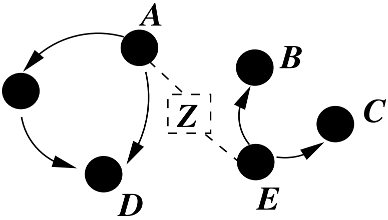


- Citation graph:
  — documents $\Rightarrow$ vertices
  — citations $\Rightarrow$ directed edges
- Document similarity based on co-citations:
  — B and C are similar because they are both cited by E
- Citation graph is *sparse* and *noisy*

Issues Using a Single Citation Graph



(a) Missing citations



(b) Same authors

Exploring Multiple Data Sources

- Real-world problems are complex and involves data from multiple sources
- Consider three relationships:
  — Citation relationship: a directed graph
  — Author-Document relationship: a bipartite graph
  — Document-Venue relationship: a bipartite graph
- A special case of Entity-Relationship Model (ERM)
  — Entity types: documents, authors, and venues
  — Relationships: citation, authorship, document-venue

Experiments

- Data from Citeseer and DBLP
- Evaluations:
  — Randomly remove documents from citations, and predict missing citations
  — Use F_1 measure: $F_1 = pr/(p + r)$, where p is precision and r is recall
- Use label propagation to rank documents

F-score Comparisons

	f \ m	m=t	m=5	m=10
$DS1$	f(lap)	0.013	0.048	0.192
	f(svd)	0.035	0.086	0.138
	f(new)	**0.108**	**0.242**	**0.325**
$DS2$	f(lap)	0.011	0.046	0.156
	f(svd)	0.027	0.072	0.109
	f(new)	**0.083**	**0.158**	**0.229**

Table 1: The f-score calculated on different numbers of top documents, m.

F-score Comparisons

	f \ t	t=1	t=2	t=3	t=4
$DS1$	f(lap)	0.041	0.048	0.075	0.086
	f(svd)	0.062	0.088	0.099	0.103
	f(new)	**0.197**	**0.242**	**0.248**	**0.252**
$DS2$	f(lap)	0.037	0.047	0.068	0.077
	f(svd)	0.049	0.072	0.082	0.086
	f(new)	**0.121**	**0.158**	**0.181**	**0.182**

Table 2: The f-score w.r.t. different numbers of left-out documents, t.

Web Advertisement

Example: Google AdSense, Microsoft adCenter

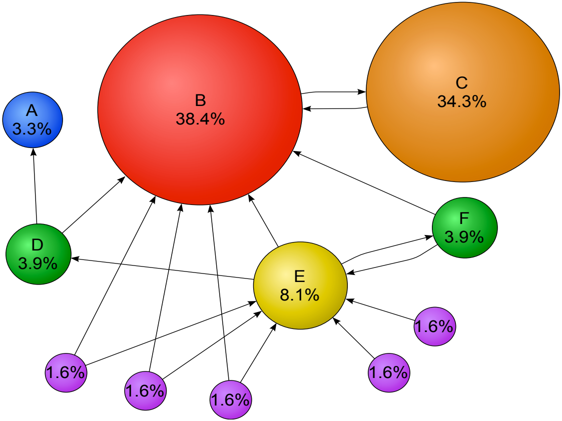
- Task: Fnd a list of ads optimized according to expected income

Example: Bit-Phrase Recommendation

- Task: suggest a list of bit-phrases for a given ad
  — budget constraints
  — maximize CTR
  — specific user populations

Google's PageRank

- Examples: Google's PageRank



- Task: Find Web page popularity
  — inbound links are recommendations
  — recommendations from good site are more valuable

Problem Formulation

- Users: $u, v \in \mathcal{U}$; Items: $i, j \in \mathcal{I}$
- Ratings: r_{ui} indicating degree of preference of user u for item j, higher values $\Rightarrow$ stronger preference
- **Problem.** Ratings are not defined over all $\mathcal{U} \times \mathcal{I}$, need to predict those missing ratings
- Incomplete rating matrix

	Casablanc	God Father	Harry Potter	Lion King
David	5	4	2	?
John	3	2	?	5
Jenny	5	2	5	?

Profiles

- User profile: user characteristics, such as age, gender, income, marital status, etc.
- Item profile: for movies, each movie can be represented by its title, genre, director, year of release, leading actors, etc.
- Explicit elicitation from users: through questionnaires, for example
- Implicit/latent profile: learned from their transactional behavior over time.

Recommendation Systems

- CONTENT-BASED RECOMMENDATIONS: The user will be recommended items similar to the ones the user preferred in the past — information retrieval and information filtering
- COLLABORATIVE RECOMMENDATIONS: The user will be recommended items that people with similar tastes and preferences liked in the past
- HYBRID APPROACHES: Combining collaborative and content-based methods.

Content-Based Methods

- Mostly for items containing textual information: documents, product reviews, news articles
- Items represented by content profile: list of keywords
- User profile: the content of the items previously seen and rated by the user
- Information retrieval techniques, document classification
- Limitations:
  — Limited content analysis
  — Overspecialization
  — New users: cold-start problem

Collaborative Filtering

- Memory-based or heuristic-based:
  — User-based collaborative fltering
  — Item-based collaborative fltering
- Model-based

User-Based Collaborative Filtering

To predict r_{ui}

- For user u, compute similarity with others users
  — Similarity measures $sim(u, v)$: $I_{uv} = \{i | r_{ui} \neq ?, r_{vi} \neq ?\}$, p_u and p_v retrictions of rating vectors to I_{uv}
  - Correlation: correlation between p_u and p_v
  - Cosine: cosine between p_u and p_v
- Aggregate the ratings r_{vi} where v is highly similar to u
  — $S_u = \{v | sim(u, v) \geq t\}$
  - Means on the best users

$$\hat{r}_{ui} = \frac{1}{|S_u|} \sum_{v \in S_u} r_{vi}$$

  - Weighted average on the bests users

$$\hat{r}_{ui} = \frac{1}{\sum_{v \in S_u} sim(u, v)} \sum_{v \in S_u} sim(u, v) r_{vi}$$

Example

Your predictions for user1 on item5, item6 and item7?

	item1	item2	item3	item4	item5	item6	item7
user1	5	3	4	1	?	?	?
user2	5	3	4	1	5	2	5
user3	5	?	4	1	5	3	?
user4	1	3	2	5	1	4	2
user5	4	?	4	4	4	?	4

Item-Based Collaborative Filtering

To predict r_{ui}

- For item i, compute similarity with others items
- Aggregate the ratings r_{uj} where j is highly similar to i

Matrix Factorization Methods

- Users and items are represented by *latent* features: $u \to f_u$, $i \to f_i$, k-dimensional row vectors
- There might be domain-specific meaning attached to each dimension
- Assumption: $r_{ui} \approx f_u f_i^T$, in matrix form

$$R \approx F_U F_I^T$$

- Let S be binary matrix ecoding mssing ratings. We can find F_U and F_I by solving the following optimization problem,

$$\min \|S \odot (R - F_U F_I^T)\|_F^2 = \sum_{(u,i)\in S} (r_{ui} - f_u f_i^T)^2$$

Issues for Collaborative Filtering Methods

- New users, new items
  — cold start problem
- Data sparsity
- Robustness

Hybrid Methods

- Implementing collaborative and content-based methods separately $\Rightarrow$ combining the predictions
- Incorporating some content-based characteristics into a collaborative approach
  — user profiles used to compute user similarity (helps with sparsity)
- Incorporating some collaborative characteristics into a content-based approach
  — matrix factorization $\Rightarrow$ latent features
  — user/item profiles augmented with latent features
- Constructing a unified model incorporating both characteristics

Extensions

- Better understanding of users and items
- Exploring contexts:
  — time sensitivity: the time of the year, such as season or month, or the day of the week
  — the person(s) with whom the product will be consumed or shared and under which circumstances
  — Example, vacation package

  > *the time of the year, with whom the user plans to travel, traveling conditions and restrictions at that time, and other contextual information.*

  — Formulated as tensor factorization

Extensions

- Multcriteria ratings
  — Example: Zagat's Guide: food, decor, and service
- General approaches
  — finding Pareto optimal solutions
  — taking a linear combination of multiple criteria and reducing the problem to a single-criterion optimization problem
  — optimizing the most important criterion and converting other criteria to constraints
- Nonintrusiveness and Cost-sensitive recommendation systems

Other Issues

- Evaluation metrics
- Explainability
  — providing information about each recommendation (eg. ratings, explanation)
- Surprise/Serendipity
- Trustworthiness
  — providing good recommendations with confidence
- Scalability
  — Applications usually need real-time prediction computation
  —The computation time has to scale with number of users and items
- Privacy
  — If the profile is private, the system need to maintain privacy using anonymity techniques.
  — Tricky to do in cross-systems situations

Robust Against Attacks

- Attacks are characterized by number of fake users and manipulation of the system.
- The attacker want to modify the distribution of the ratings without being easy to detect
- Known attacks: sampling attack, random attack, average attack, bandwagon attack...
- Detecting attack : fnd proiles which are unlikely according to the global distribution of profiles, fnd profiles updates which are unlikely according to the global distribution of updates, differential privacy

Key Challenges

RecSys 2009

- Transparency/Explainability
  — Convince a user to accept a recommendation
  — Help a user make a good decision
  — Help a user fit a goal or mood
  — What types of transparency are valuable?
- Exploration versus Exploitation
  — Cold start problems: new items and new users
  — Choosing what questions to ask users
  — How can meta-data on user or item help?

Key Challenges

- Time value/Temporal issues
  — Does value of user input decay with time?
  — Do items change in relevance with time?
  — Do items change in relevance with time?
  — Recurrent vs. transient interests?
  — Short-term (news, trip to Hawaii), Intermediate term (Olympics), Long-term (Chicago Cubs)
- User Action Interpretation
  — Ratings are valuable, but less frequent
  — Positive and negative signal identification: Every action is opportunity to learn about user and content
  — Research challenge to model user intent on behavioral data

Key Challenges

- Evaluating recommender systems
  — Evaluation of entire user experience: RMSE not enough even for predictions; Enjoyment Prediction
  — Best practices for experiments
  — Business metrics and Proxy metrics
- Scalability
  — What are the key scalability features you would value? Large user bases, large user event sets, large item pools
  — What parallelization structure is best?
  – What "hidden" requirements prevent algorithm adoption?

Publication Venues

- ACM Recommender Systems
  — ACM Recommender Systems 2007, Minneapolis, Minnesota, USA
  — ACM Recommender Systems 2008, Lausanne, Switzerland
  — ACM Recommender Systems 2009, New York City, New York, USA
  — ACM Recommender Systems 2010, Barcelona, Spain
  — ACM Recommender Systems 2011, Chicago, Illinois, USA
- WWW, SIGIR, SIGKDD, ICML, NIPS, ICWSM

Final Remarks

- Recommendation systems rely on ML technqiues with broadly visible impacts
- Personalization and consensus information from large comminities
- Set of applications are still being expanded
- Very much interdisciplinary in nature