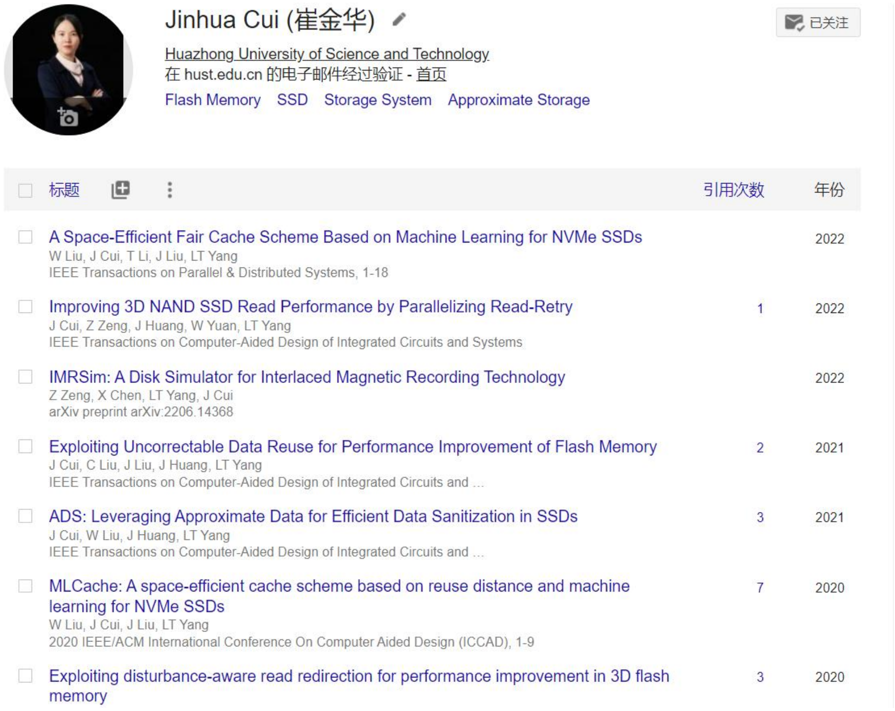
# Chapter 6: Recommender Systems

崔金华
电子邮箱: jhcui@hust.edu.cn
个人主页: https://csjhcui.github.io/

致谢:课件主要参考《 Mining of Massive Datasets》(Second Edition) J. Leskovec, A. Rajaraman, J. Ullman的相关课件, 特此致谢!!!

# Example: Recommender System

华中科技大学 计算机科学与技术学院 School of Computer Science &Technology, HUST





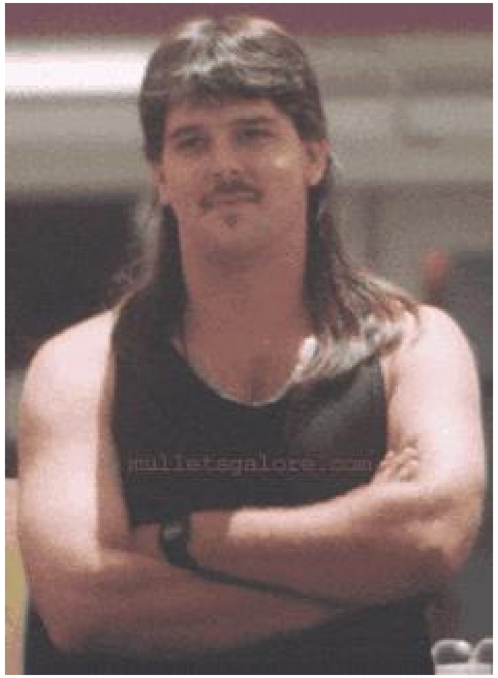
- **Customer X**
  - Published papers
  - Saved papers in personal library

- **Customer X**
  - Recommend related papers

# Another Example: Recommender System

华中科技大学 计算机科学与技术学院 School of Computer Science &Technology, HUST

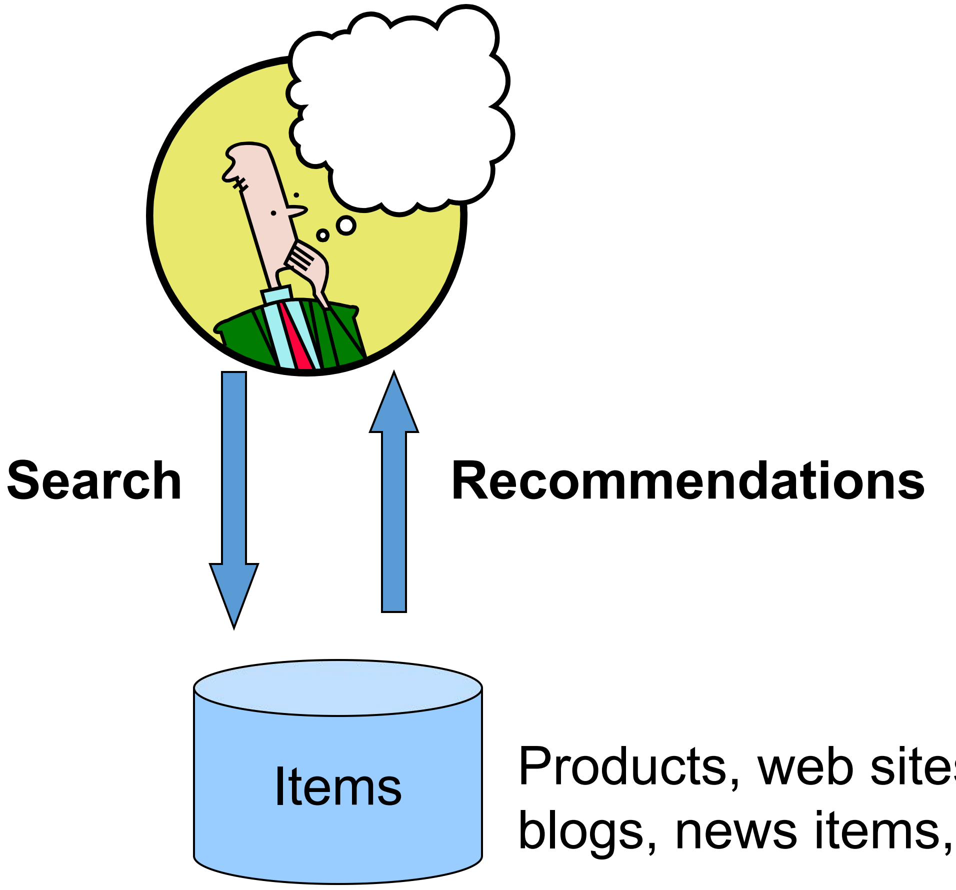
- **Customer X**
  - Buys Metallica(金属乐队) CD
  - Buys Megadeth(麦格德斯) CD




- **Customer Y**
  - Does search on Metallica
  - Recommender system suggests Megadeth from data collected about customer **X**

# Recommendations






Products, web sites, blogs, news items, …

**Examples:**

新浪微博

Baidu百度

淘宝网 Taobao.com

抖音

豆瓣 douban

知乎 有问题 上知乎

iQIYI 爱奇艺

bilibili

腾讯视频

芒果tv

YOUKU

…

# From Scarcity to Abundance

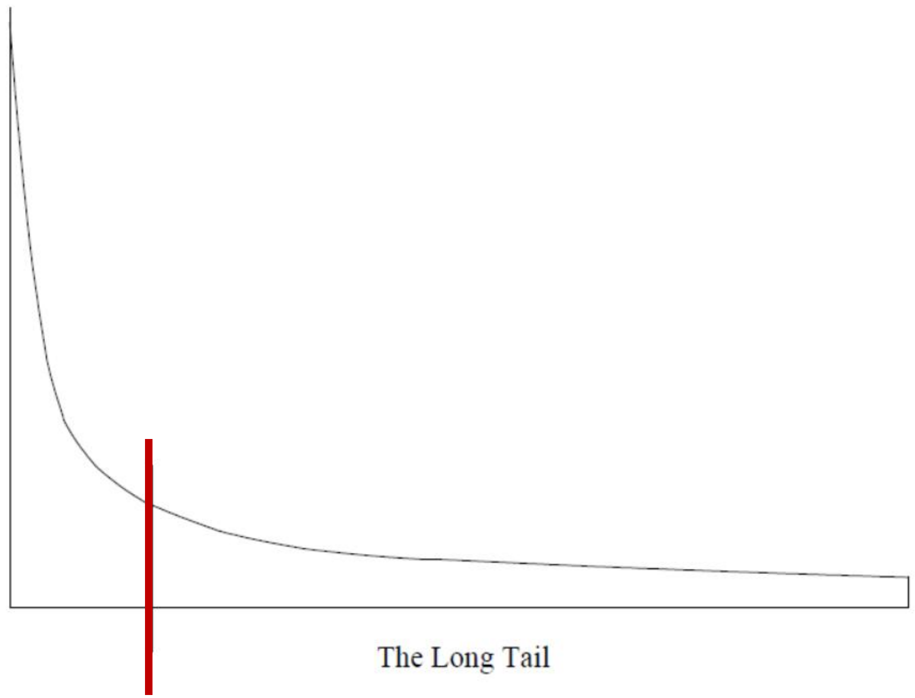


- **Shelf space is a scarce commodity for traditional retailers**
  - Also: TV networks, movie theaters,…

- **Web enables near-zero-cost dissemination of information about products**
  - From scarcity to abundance
  - **Long tail phenomenon (长尾现象)**

# The Long Tail

- The distinction between the physical and on-line worlds has been called the **long tail phenomenon (长尾现象)**.




- 纵坐标代表流行度(the number of times an item is chosen).
- 所有项按照流行度在横坐标上排序.
- 实体机构只列出红色竖线左边的流行项; 在线机构能提供包括流行项和尾部项在内的全范围的项.

# From Scarcity to Abundance

- **Shelf space is a scarce commodity for traditional retailers**
  - Also: TV networks, movie theaters,…

- **Web enables near-zero-cost dissemination of information about products**
  - From scarcity to abundance
  - **Long tail phenomenon (长尾现象)**
- **More choice necessitates better filters**
  - Recommendation engines
  - How **Into Thin Air 《巅峰》** made **Touching the Void 《攀越冰峰》** a bestseller: http://www.wired.com/wired/archive/12.10/tail.html

# Types of Recommendations



## ❑ Editorial and hand curated
- List of favorites
- Lists of "essential" items

## ❑ Simple aggregates
- Top 10, Most Popular, Recent Uploads

## ❑ Tailored to individual users
- Amazon, Netflix, …

# Formal Model



- ***X*** = set of **Customers**
- ***S*** = set of **Items**
- **Utility matrix** (效用矩阵) $\boldsymbol{u}$: $\boldsymbol{X} \times \boldsymbol{S} \to \boldsymbol{R}$
  - ***R*** = set of ratings
  - ***R*** is a totally ordered set
  - e.g., **0-5** stars, real number in **[0,1]**

# Utility Matrix

| | Avatar (阿凡达) | LOTR (指环王) | Matrix (黑客帝国) | Pirates (加勒比海盗) |
|---|---|---|---|---|
| Alice | 1 | | 2 | |
| Bob | | 5 | | 3 |
| Carol | 2 | | 1 | |
| David | | | | 4 |

# Key Problems



- **(1) Gathering "known" ratings for matrix**
  - How to collect the data in the utility matrix

- **(2) Extrapolate unknown ratings from the known ones**
  - Mainly interested in high unknown ratings
    - We are not interested in knowing what you don't like but what you like

- **(3) Evaluating extrapolation methods**
  - How to measure success/performance of recommendation methods

# (1) Gathering Ratings

- **Explicit**
  - Ask people to rate items
  - Doesn’ t work well in practice – people can’ t be bothered

- **Implicit**
  - Learn ratings from user actions
    - E.g., purchase implies high rating
  - What about low ratings?

- **Hybrid: both explicit and implicit**

# (2) Extrapolating Utilities



- **Key problem:** Utility matrix ***U*** is **sparse**
  - Most people have not rated most items
  - **Cold start:**
    - New items have no ratings
    - New users have no history

- **Approaches to recommender systems:**
  - **1)** Content-based
  - **2)** Collaborative
  - **3)** Latent factor based
  - ……