

# Motori di Ricerca

presente e futuro prossimo

## Recommendation Systems

Paolo Ferragina, Università di Pisa

## Recommendations Systems

Il loro uso principale è nei siti di e-commerce (p.e. [Amazon](#)), ma prenderanno sempre più piede nei KMS, come strumento per personalizzare siti e ricerche di utenti in base ai loro interessi!

- Solitamente si avvalgono di:
  - Item comprati o visti
  - Dati demografici
  - Interessi personali su soggetti, artisti, scrittori,...
- [Amazon](#) ha dimostrato che il [Click-through](#) e [Buied](#) items superano di molto ciò che è ottenibile con sistemi *untargeted*
- Problema interessante:
  - Milioni di utenti e item, ma risultati in real-time
  - Informazione limitata per nuovi utenti (la maggioranza)
  - Informazione sovrabbondante per vecchi utenti
  - Informazione volatile e puntuale

Paolo Ferragina, Università di Pisa

## Recommendations Systems

Esistono quattro approcci per risolvere il problema

- User-to-User Collaborative filtering
- Clustering model
- Search-based method
- Item-to-item collaborative filtering (Amazon, brevetto US)
- Primi due approcci cercano "utenti simili"
  - Sfruttano item comprati e valutati
- Gli ultimi due approcci cercano "item simili"
  - Stesso autore, artista, tipo, keywords, ...
  - Se essi vengono "comprati insieme" dagli utenti
- L'ultimo approccio scala molto bene ed è real-time

Paolo Ferragina, Università di Pisa

## Traditional Collaborative Filtering

Sia  $N$  il numero di item nel catalogo:

- Un utente viene visto come un vettore  $N$ -dimensionale
- Componenti positive per item comprati o valutati bene
- Componenti negative per item valutati male
- Componente  $j$  moltiplicata per  $(1/\#\text{compratori\_item}[j])$
- I vettori sono molto sparsi
- Similarità è il classico  $\text{coseno}(\text{utente1}, \text{utente2})$
- Approccio computazionalmente costoso
- Speed-up possibile ma impatta sulla qualità dei suggerimenti
  - Sampling sugli utenti
  - Filtraggio utenti che hanno comprato pochi item
  - Partizionando gli item in gruppi per categoria o soggetto, usiamo questi per trovare più velocemente l'item simile

Paolo Ferragina, Università di Pisa

## Cluster Models

### Utente visto ancora come vettore N-dimensionale

- Metrica di similarità, p.e. *Coseno*
- Si formano *cluster* di utenti simili in modo greedy (efficienza)
- Ogni *cluster* ha un rappresentante
- Utente classificato nel *cluster* il cui rappresentante è più simile
  
- Migliori prestazioni e scalabilità
  - Confrontiamo soltanto i rappresentanti
  
- Qualità dei suggerimenti è bassa
  - N.ro rappresentanti impatta sull'efficienza on-line

Paolo Ferragina, Università di Pisa

## Search-based method

### Si serve di un motore di ricerca sulla descrizione degli item

- Deriva dagli item comprati o valutati dall'utente, a cui il suggerimento si rivolge, un insieme di *keywords*
- Costruisce un insieme di query e le esegue
  
- Prestazioni e scalabilità sono molto buone se il numero di item comprati o valutati dall'utente sono pochi
  - Poche query generate
  
- Qualità dei suggerimenti è bassa
  - Troppo specifici o troppo generali

Paolo Ferragina, Università di Pisa

## Item-to-Item Collaborative Filtering

Amazon offre due tipi di suggerimenti:

- Link "Your Recommendations" è generale
- "Shopping chart recommendations" è corrente

Definisce due item *simili* se comprati insieme da qualche utente

- Non si può iterare su tutte le coppie, molte non comprate insieme
- Suggerimento su I
  - Per ogni utente U che ha comprato I, si trovano gli altri item I' comprati da U
  - Si calcola la similarità della coppia (I,I') come il *coseno* tra i loro vettori M-dimensionali formati sugli utenti

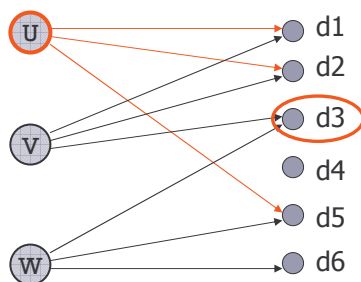
Suggerimento per utente V generato prendendo le migliori coppie che hanno uno degli item comprati da V come 1° componente

Qualità dei suggerimenti e scalabilità molto alta

- Pressoché indipendente dalla dimensione dell'archivio

Paolo Ferragina, Università di Pisa

## Una semplificazione grafica



Il sistema cerca gli altri utenti che hanno visto d1, d2 o d5 (danno I). I documenti visti da questi utenti formano la seconda componente della coppia (I,I').

Raccomanda a U il documento, o i documenti, che sono molto popolari tra quegli utenti

Paolo Ferragina, Università di Pisa