UNIVERSITAT DE BARCELONA

DATA SCIENCE @ UNIVERSITAT DE BARCELONA

Master on Foundations of Data Science

Recommender Systems

Content Based Recommendations

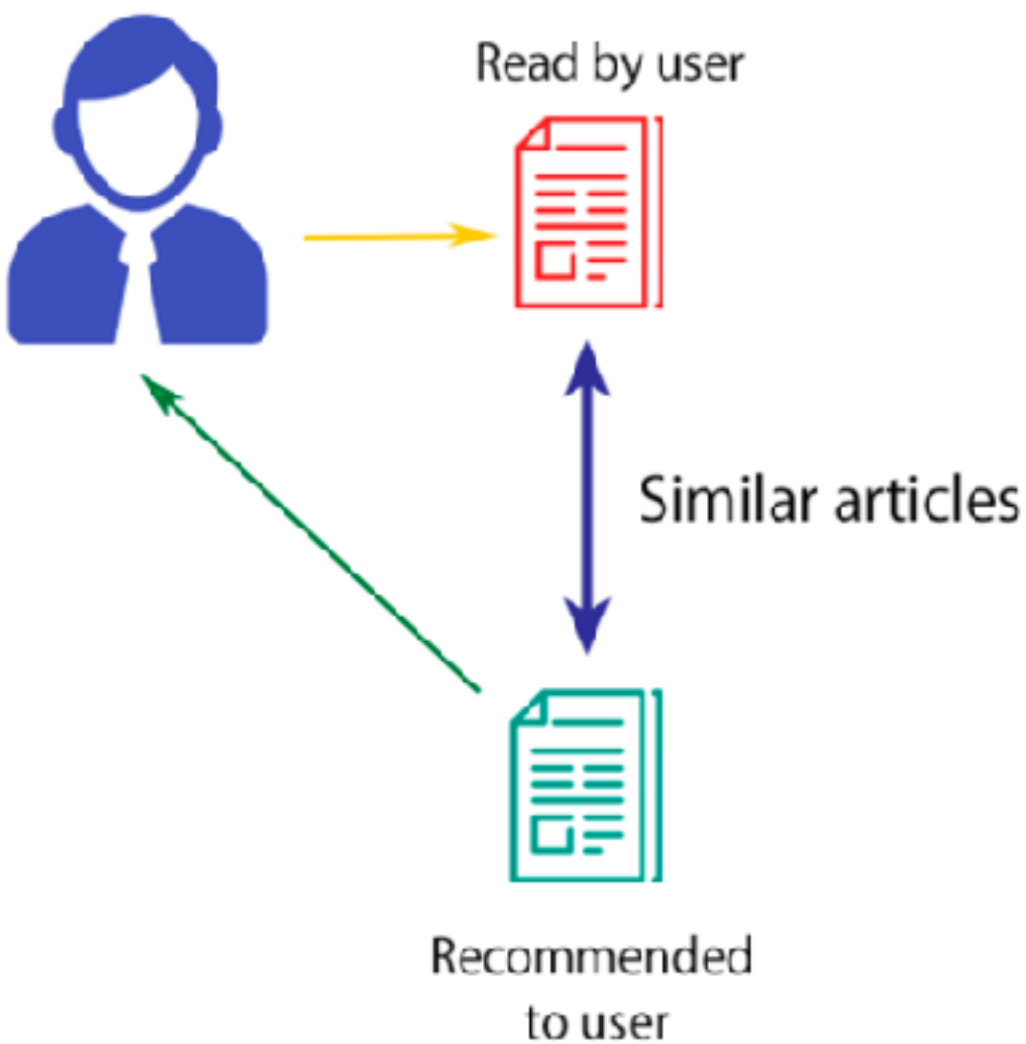
Santi Seguí | 2017-2018

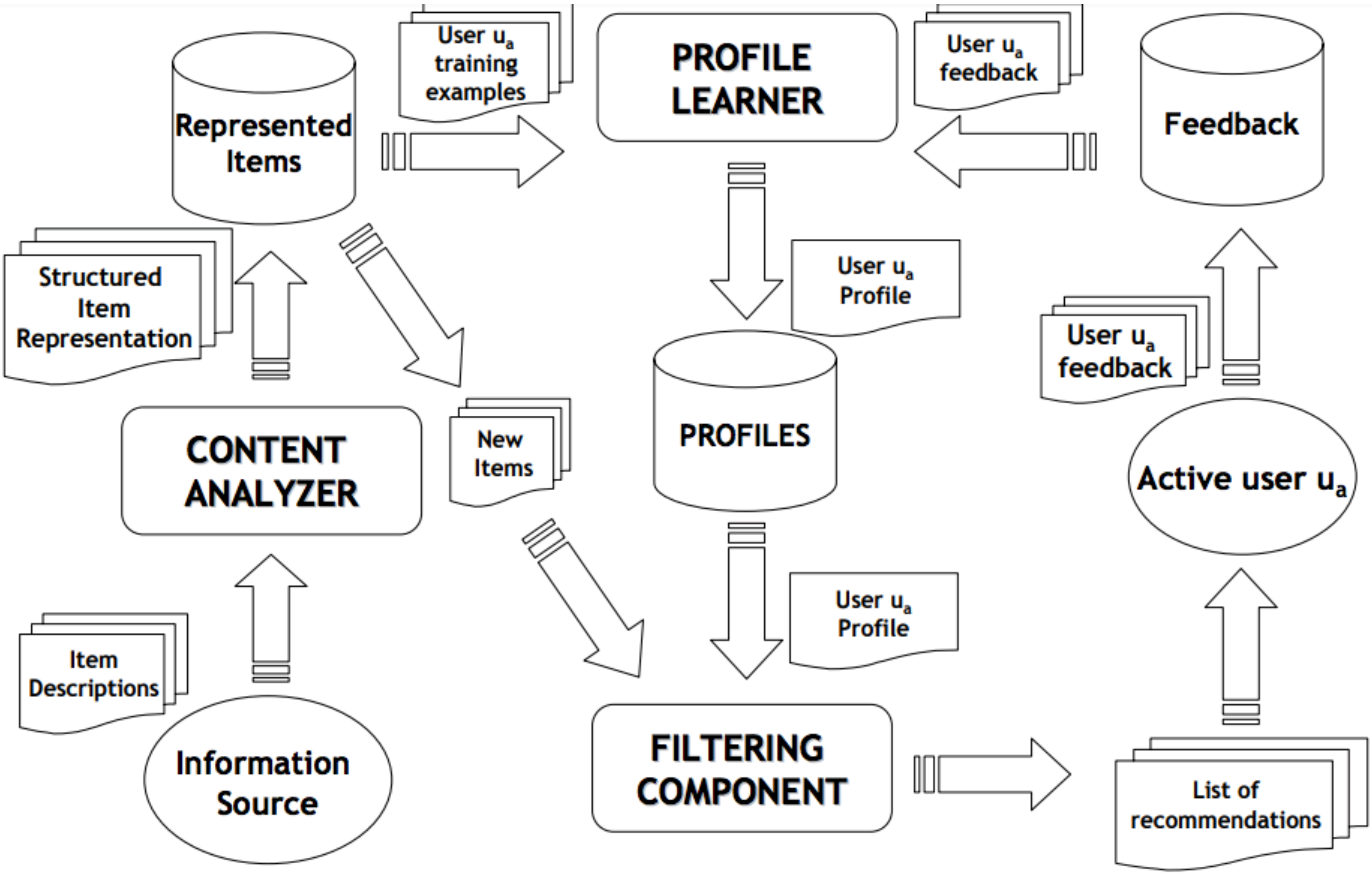
Content-Based Methods

Conceptual goal:
Give me recommendation based on the content (attributes) I liked before

Input:
User ratings (user profile) + item attributes

What is a content based approach?




User u_a training examples
PROFILE LEARNER
User u_a feedback
Represented Items
Feedback
Structured Item Representation
User u_a Profile
User u_a feedback
PROFILES
CONTENT ANALYZER
New Items
Active user u_a
User u_a Profile
Item Descriptions
Information Source
FILTERING COMPONENT
List of recommendations

Content-based Filtering

- Requires **content** (from the items) that can be encoded as meaningful **features**.
  - Item title, description, price, image, etc…
- Need to compute a **similarity between items** based on the content of the items.
- **Users' tastes** must be represented as a **learnable function** of these content features.
- Does **not** to exploit quality judgments of **other users**.
  - Unless these are somehow included in the content features.

Advantages of CBRS

- **User independence**
  - CBRS exploit solely ratings provided by the active user to build the recommendation
  - No need for data on other users
- **Transparency**
  - Can provide explanations for recommended items by listing content-features that caused an item to be recommended
- **New Item (Cold Start on items)**
  - Can recommend new and unknown items

When Content Based?

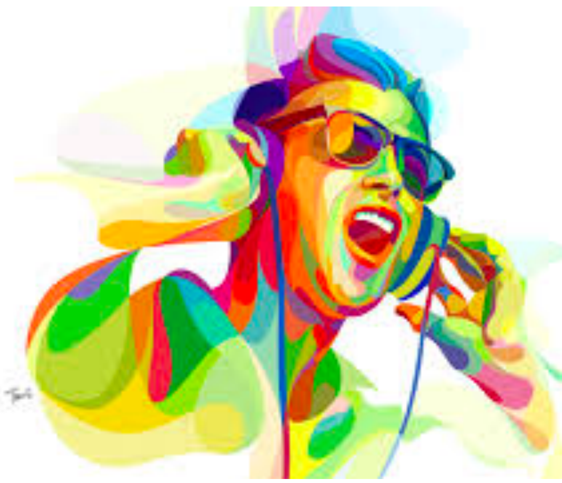
Really popular for cold-start problems.

Popular in domain like:
news recommendation or music recommendation

Famousness CB Recommender Systems

Item profile

- For each item, create an item profile
- Profile is a set of features.
    - Which features??

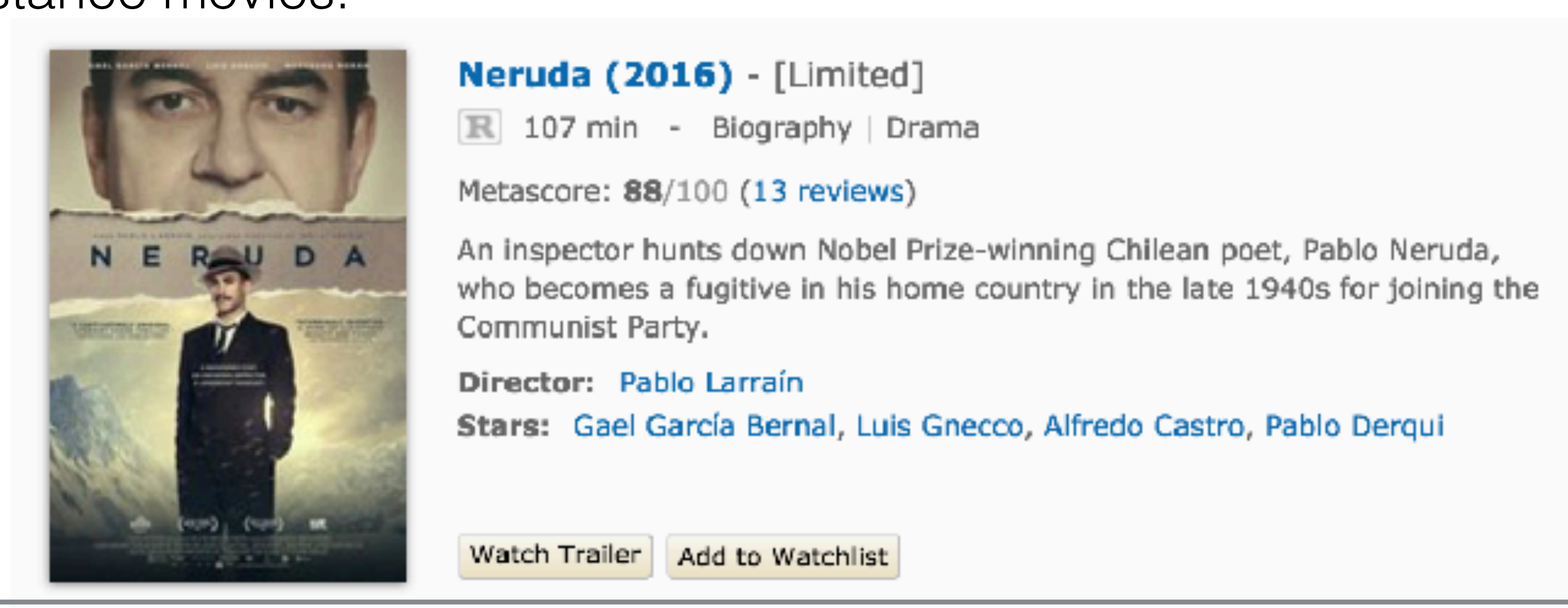
Item profile

- For each item, create an item profile
- Profile is a set of features.
  - **Movies**: author, title, director, actor,…
  - **Images**, videos: metadata and tags
  - **People**: set of friends
  - **News**: keywords,..
- Convenient to consider the item profile as a vector:
  - One entry per feature (e.g., each actor, director, ..)
  - Vector might be boolean or real-valued

What is "content"?

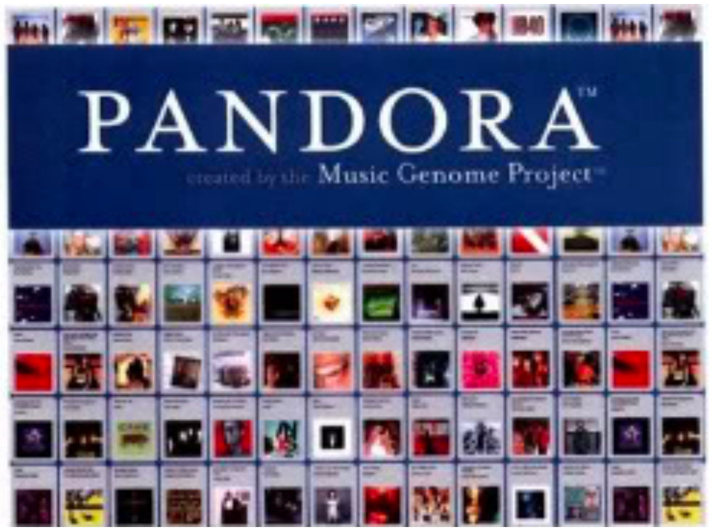
- Content Based recommenders systems have been applied mosty on text document
- However, content of items, items such as movies or songs, can be represented as text documents
  - With textual description of their basics characteristics
  - Structured: Each item is described with the same set of attributes
  - Unstructured: Free-text document

As for instance movies:



UNIVERSITAT DE BARCELONA

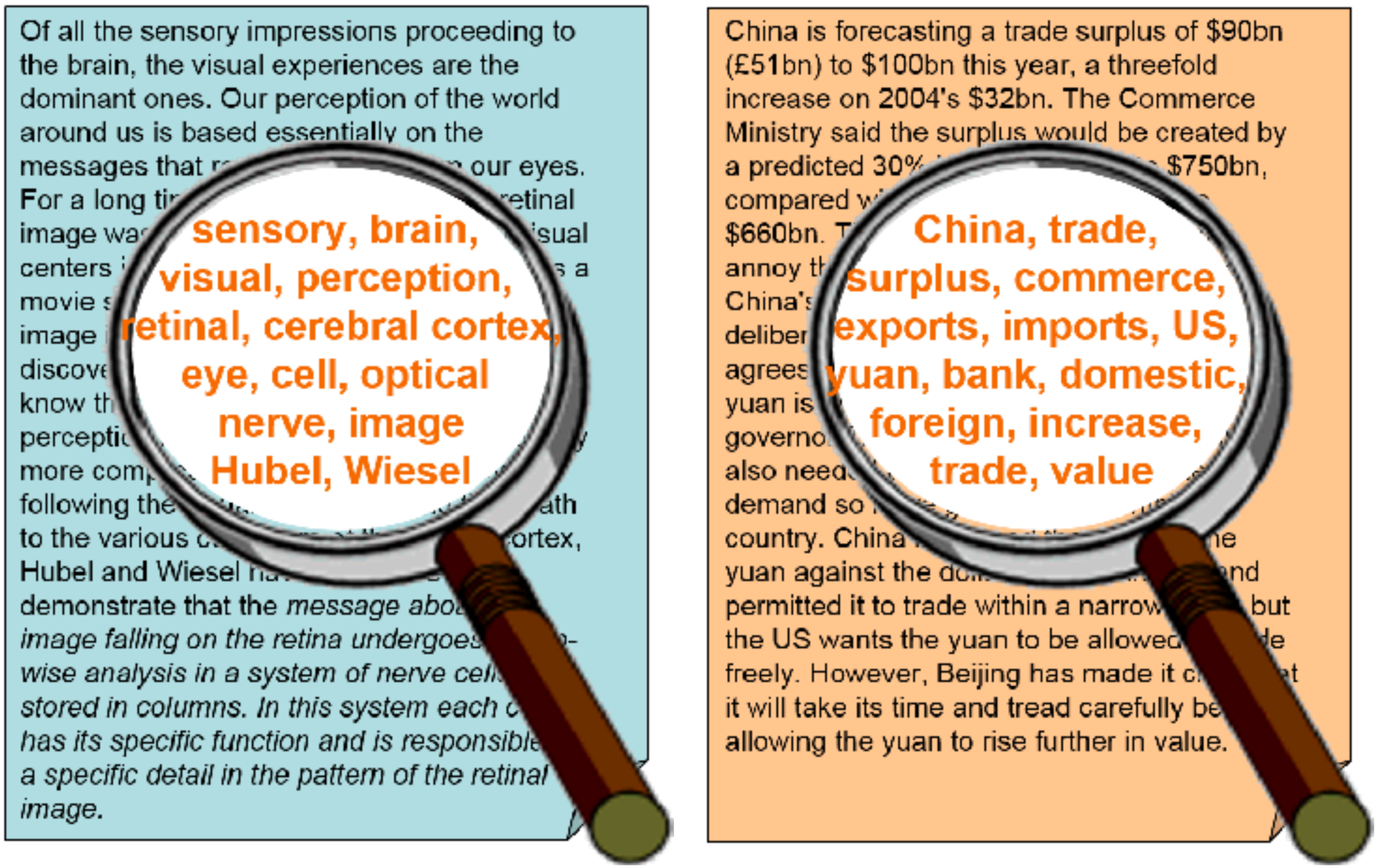
Pandora



- How it works:
  - Base it recommendation on data

    from Music Genome Project
  - Assigns 400 attributs for each song, done by musicians.
    - Some reports says that takes half an hour per second
  - Use this method to find sons which are silimar to the users's favorite sogs

How to describe textual information?



Feature Representation and Cleaning

- Extremly important when the unstructured format is used for representation.
- Bag of Words (BOW) from the unstructured description of the products or Web Pages used to be used, however, these reprensentrations needs to be cleaned and represented in a suitable format for processing.
- Several important steps:
  - **Stop-word removal**: Words such “a”, “an,”, “the”, does not provide important information
  - **Stemming**. Variations of the same words are consolidated. For example, words such “hope” and “hoping” are consolidated into the common root “hop”
  - **Phrase extraction:** The idea is to detect words that occur together in documents on a frequent basis.

TD-IDF

In information retrieval, **tf–idf**, short for term frequency–inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining.

- **Term Frequency** x **Inverse Document Frequency**
- **Term Frequency** =
    - Number of occurrences of a term in a document (can be a simple count)
- **Inverse Document Frequency** =
    - How few documents contain this term
    - Typically : Log(#documents / #documents with term)

So, items that appears rarely or appears everywhere are note important

TD-IDF

Document 1

Term	Term Count
this	1
is	1
a	2
sample	1

Document 2

Term	Term Count
this	1
is	1
another	2
example	3

$$\mathrm{tf}(''\mathsf{this}'', d_1) = \frac{1}{5} = 0.2 \qquad \mathrm{tf}(''\mathsf{this}'', d_2) = \frac{1}{7} \approx 0.14$$

idf is constant per corpus, and **accounts** for the ratio of documents that include the word "this". In this case, we have a corpus of two documents and all of them include the word "this".

$$\mathrm{idf}(''\mathsf{this}'', D) = \log\left(\frac{2}{2}\right) = 0$$

tf–idf is zero for the word "this", which implies that the word is not very informativ as it appears in all documents.

$$\mathrm{tfidf}(''\mathsf{this}'', d_1) = 0.2 \times 0 = 0 \qquad \mathrm{tfidf}(''\mathsf{this}'', d_2) = 0.14 \times 0 = 0$$

What does TFIDF do?

- Automatic find of stop words, common terms
- Promote core terms over accidental terms
- When it fails?
  - If core term is not used frequently in a document (e.g., legal contracts)

Variants and Alternatives

- Some applications use variants on TF
  - Binary
  - Logarithmic frequencies
  - Normalized frequencies (log(tf + 1))

Relevance and Problems

- Significance in Documents
  - Titles, heading,… (different weight?)
- Phrases and n-grams
  - “recommender system” != “recommender” and “system”
  - Adjacency
- General score
- Implied Contet
  - Links, usage,…

Keyword Vector

- The universe of possible keywords defines a content space
  - Each "keyword" is a dimension
  - Each item has a position in that space; that position defines a vector
  - Each user has a taste profile that is also a vector in that space
  - The match between user preferences and items is measured by how closely the two vectors aling
  - May want to limit/collapse keyword space

Vector representation

- Simple 0/1 (keyword applies or does not)
- Simple occurrence count
- TFIDF
- Other variants include factors such as document length
- Eventually, this vector is ofter normalized

Other terms?

- Clothing attributes (color, size, etc..)
- Terms used in hotel reviews (pool, front desk, friendly)
- Terms used in news articles (elections, football, economy)

How to build preferences?

- Set of “keyword” a user  or 
- count the number of times the user chooses item with each keyword
- or more sophisticated methods

User Preferences

- My preferences:
  - **Movies** - I like SeVeN, American History X, Gladiator
  - **Hotels** - I prefer 24-hour front desk, internet, spa
  - **Music** - I like Blur, Pulp, The Verve,…

User Preferences

- **Vector Space Model:**
    - single scalar value for each dimension
    - same dimensions than item Vector Space

User Preferences

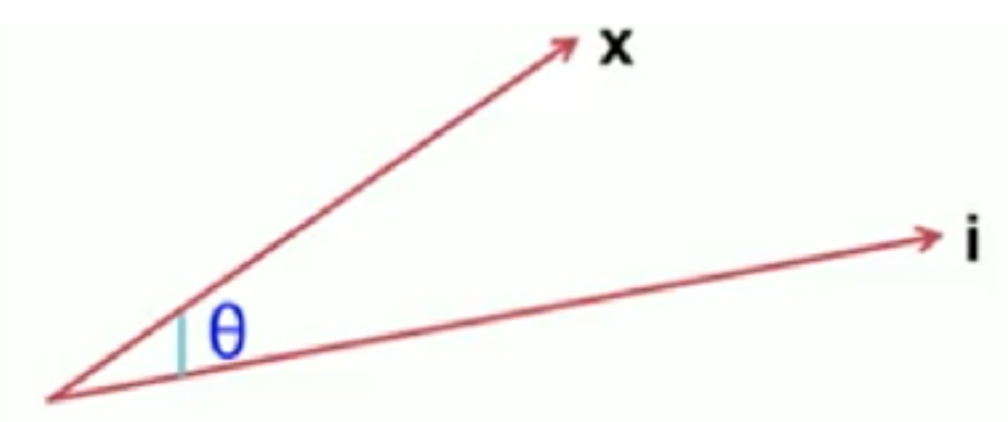
- How to accumulate features from the profiles?
  - Add together the item vectors?
    - Should we normalize first?
      - Should all items have the same weight?
    - Do we weight the vectors somehow?
      - We can use ratings..
      - Confidence?

User Preferences

- What about **new items?**
  - Update the vector taking into account number of items used
  - Update with some decay?

Making Predictions

- User profile x, Item profile i
- Estimate U(x,i) = cos(ø) = (x . i) / (|x| |i|)



- Technically, the cosine distance is actually the angle ø. And the cosine similarity is the angle 180-ø

How to improve it?

How to improve it?

- Better classifier/Regressor
  - Liner regression to XGboost models are used
  - Each feature will have a different weight on the recommendation
- **Richer representations**

Keywords are **not appropiate** for representing content, due to **polysemy**, **synonymy**, **multi-word concepts**,….

Keyword-based Models

doc1
AI is a branch of computer science

doc2
the 2011 International Joint Conference on **Artificial Intelligence** will be held in Spain

doc3
apple launches a new product...

Items

Multi-words Concepts

Synonyms

User Profile

artificial	0.02
intelligence	0.01
apple	0.13
AI	0.15
…	…

apple

Polysemy

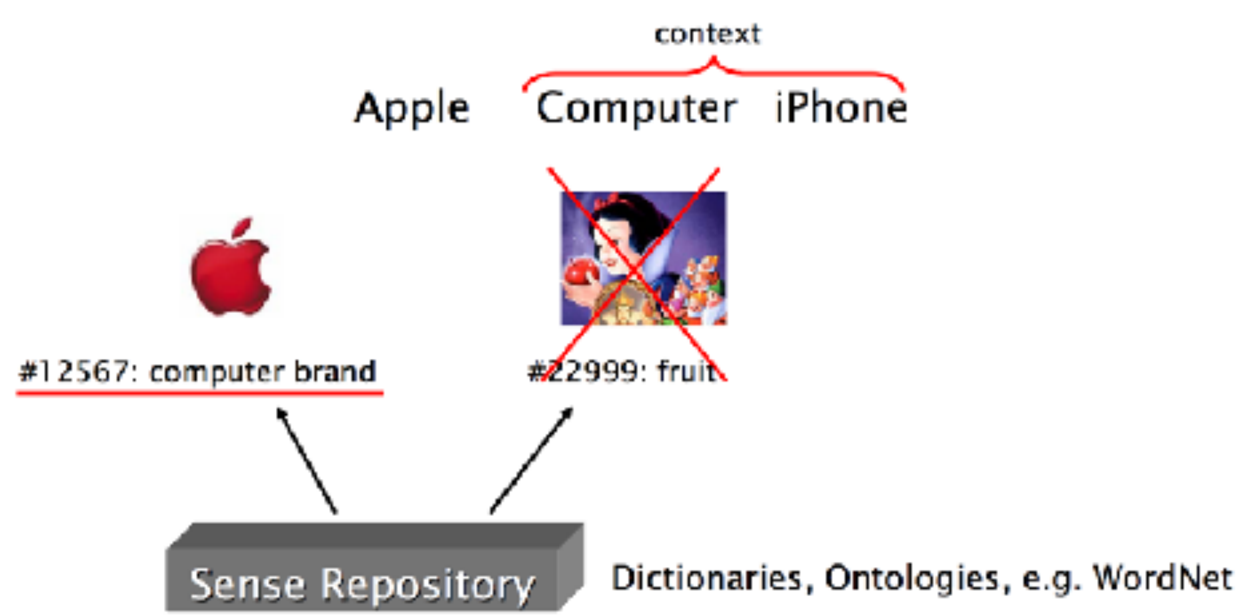
NLP methods are needed for the elicitation of user interests

Richer representations

- Semantic Analysis
  - **Semantics:** concept identification in text-based representations through advanced NLP techniques ->“beyond keywords”
  - **Personalization:** representation of user information needs in an effective way -> “deep user profiles”

Sematic Analysis using Ontologies

- Word sense Disambiguation (WSD) -> From words to meanings
    - WSD selects the proper meaning (sense) for a word in a text by taking into account the context in which that word occurs

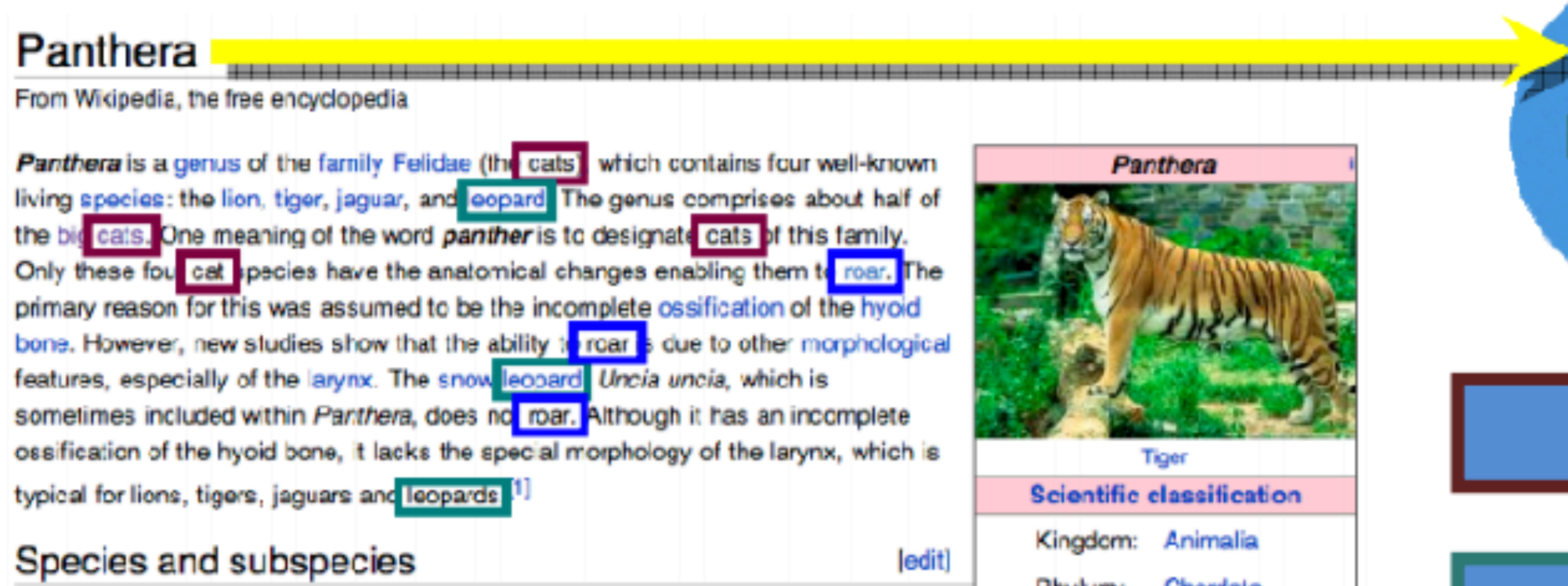


A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation M Degemmis, P Lops, G Semeraro User Modeling and User-Adapted Interaction 17 (3), 217-255	181	2007

Sematic Analysis using Enclyclopedic Knowledge Sources

Wikipedia is viewed as an <u>ontology</u> - a collection of ~<u>1M</u> concepts

Every Wikipedia article represents a **concept**

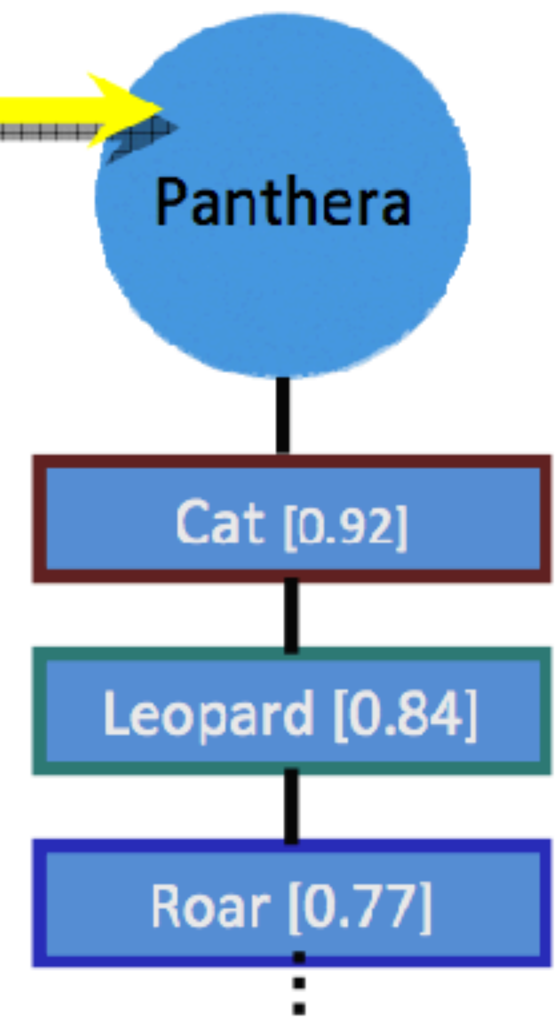

Panthera

From Wikipedia, the free encyclopedia

Panthera is a genus of the family Felidae (the cats) which contains four well-known living species: the lion, tiger, jaguar, and leopard. The genus comprises about half of the big cats. One meaning of the word *panther* is to designate cats of this family. Only these four cat species have the anatomical changes enabling them to roar. The primary reason for this was assumed to be the incomplete ossification of the hyoid bone. However, new studies show that the ability to roar is due to other morphological features, especially of the larynx. The snow leopard *Uncia uncia*, which is sometimes included within *Panthera*, does not roar. Although it has an incomplete ossification of the hyoid bone, it lacks the special morphology of the larynx, which is typical for lions, tigers, jaguars and leopards.[1]

Species and subspecies [edit]

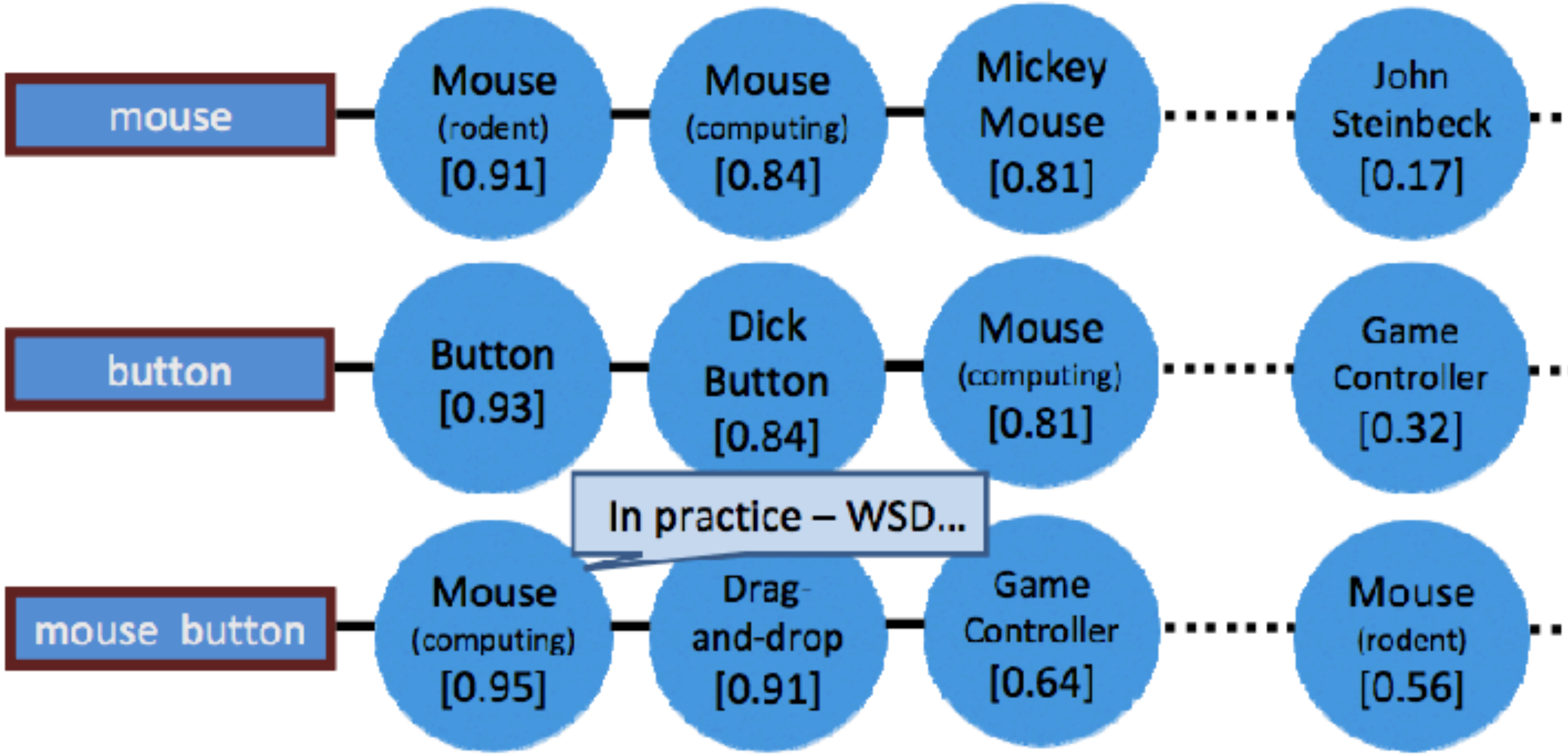
Panthera	
Tiger	
Scientific classification	
Kingdom:	Animalia
Phylum:	Chordata



Article words are associated with the concept (TF-IDF)

Sematic Analysis using Enclyclopedic Knowledge Sources

The **semantics** of a **text fragment** is the **average** vector (centroid) of the semantics **of its words**



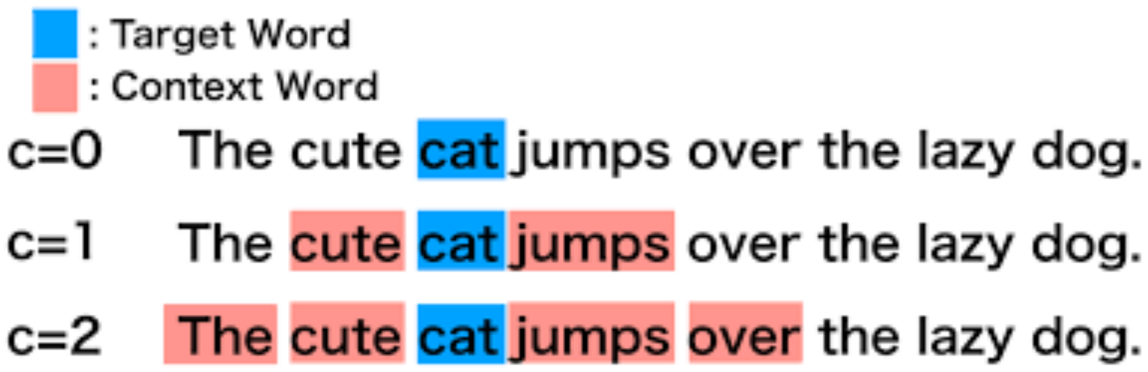
word2vec

(WATER - WET) + FIRE = FLAMES

(PARIS - FRANCE) + ITALY = ROME

(WINTER - COLD) + SUMMER = WARM

(MINOTAUR - MAZE) + DRAGON = SIMCITY



Semantics Aware CB

Richer representations allows to

overcome limited content analysis
provide better explanations
foster unexpectedness and serendipity

Matrix Facorization

- Latent Semantic Analysis
- Latent Dirichlet Allocation

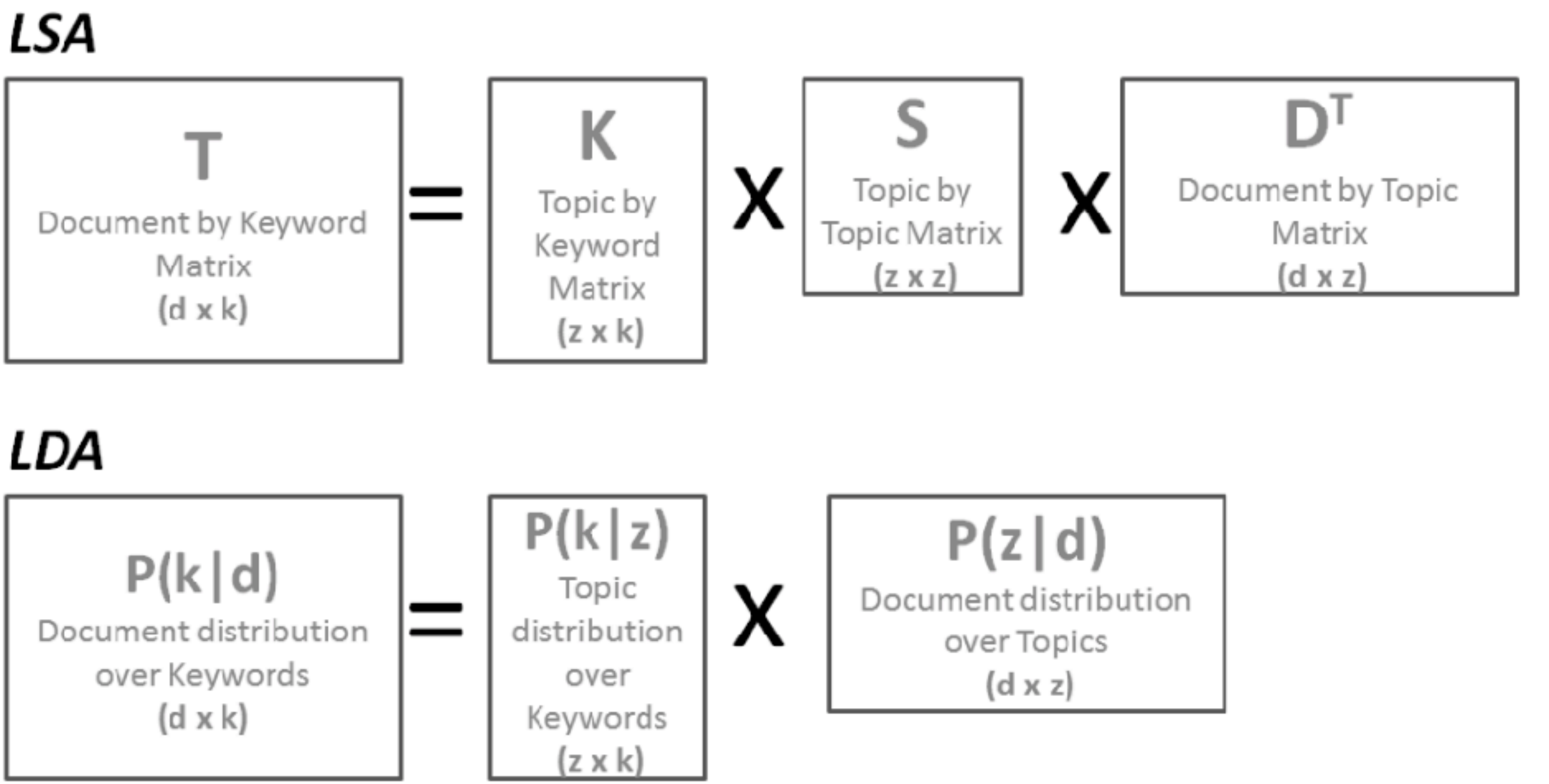


Fig. 2: Matrix decomposition for LSA and LDA.

Topic Modeling

A simple way to analyze topics of large text collections (corpus).

Latent dirichlet allocation
DM Blei, AY Ng, MI Jordan
Journal of machine Learning research 3 (Jan), 993-1022

Topic 1		Topic 2		Topic 3	
term	**weight**	**term**	**weight**	**term**	**weight**
game	0.014	space	0.021	drive	0.021
team	0.011	nasa	0.006	card	0.015
hockey	0.009	earth	0.006	system	0.013
play	0.008	henry	0.005	scsi	0.012
games	0.007	launch	0.004	hard	0.011

Latent dirichlet allocation
DM Blei, AY Ng, MI Jordan
Journal of machine Learning research 3 (Jan), 993-1022
17706 2003

Hands on time!

Drawbacks

- Content must be encoded in **meaningful features**
- No suitable suggestion if the analyzed content does not contains enough information to discriminate items the user likes from items the user does no like
- Keywords alone may not be suffient to judge quality/relevance of a document or web page
  - up-to-date-ness, usability, aesthetics, writing style
  - content may also be limited / too short
  - content may not be automatically extracted (multimedia)

Drawbacks

- suggest items whose scores are high when matched against the user profile
- No inherent method for finiding something unexpected
- Obviusness in recommendations
    - Suggesting “Start Treck” to a science-fiction fan: Accurate but not useful
    - users don’t want systems that produce better ratings, but sensible recommendation
- **Serendipity problem**

OVER SPECIALIZATION

Being accurate is not enough: how accuracy metrics have hurt recommender systems SM McNee, J Riedl, JA Konstan CHI'06 extended abstracts on Human factors in computing systems, 1097-1101	775	2006

Discussion & Summary

- Pure content-based filters are rarely found in commercial environments
- Content-based techniques does not requiere a user community
- Aim to learn a model user's interest preferences based on explicit or implicit information
- Good recommendation accuracy can be obtained using machine learning techniques
- Danger to recommend too many similar items