UNIVERSITÀ
DEGLI STUDI DI BARI
ALDO MORO

Semantic
Web
Access and
Personalization
research group
http://www.di.uniba.it/~swap

Content-based Recommender Systems

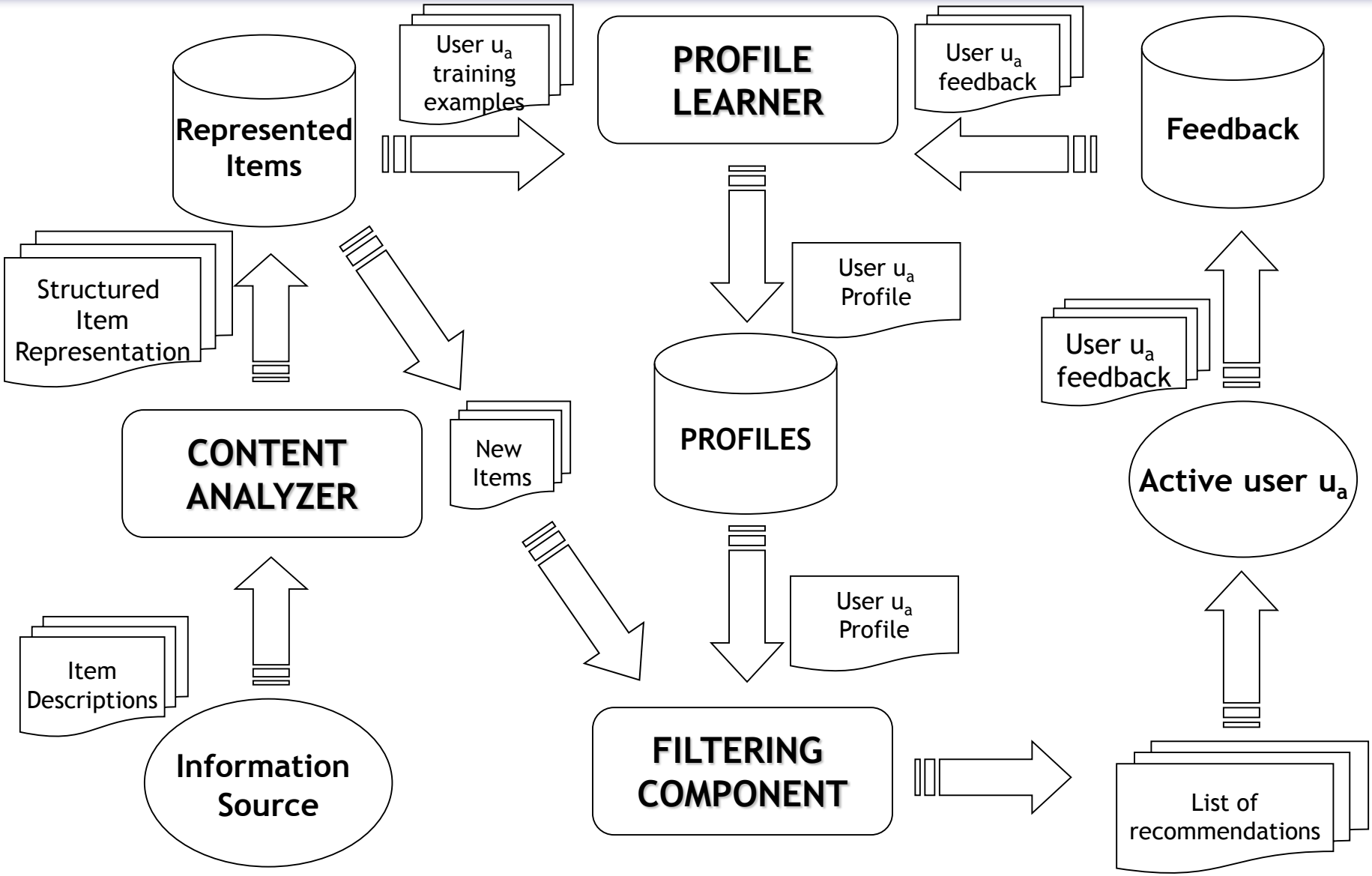
Credits

- Dietmar Jannach
- Markus Zanker
- Alexander Felfernig
- Gerhard Friedrich
- Francesco Ricci
- Marco de Gemmis
- Giovanni Semeraro
- Pasquale Lops

Outline

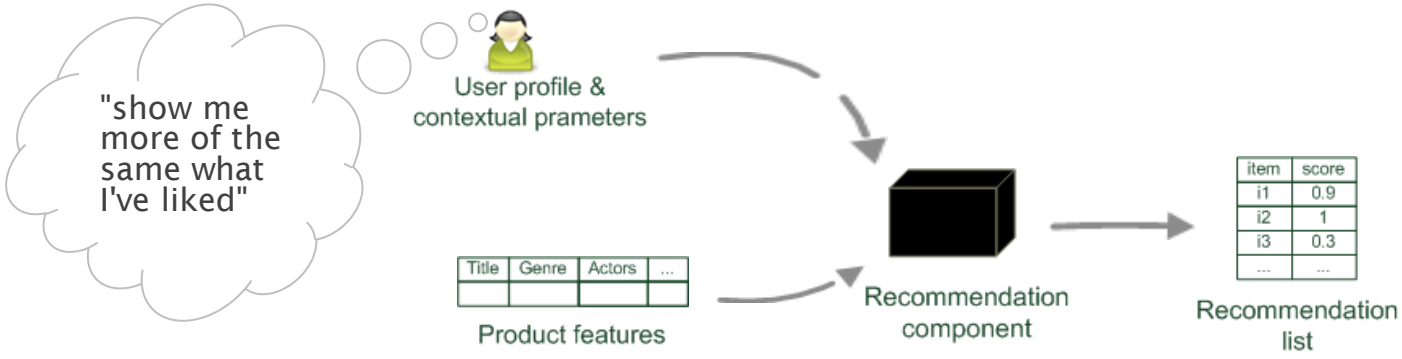
1. Content-based Filtering
   - ✓ Keyword-based item representation
   - ✓ Algorithms
2. Advanced content-based Recommender Systems
   - ✓ Word Sense Disambiguation
   - ✓ Sense-based item representation

General Architecture of CBRS



Content-based recommendation

- **While CF – methods do not require any information about the items,**
  - it might be reasonable to exploit such information; and
  - recommend fantasy novels to people who liked fantasy novels in the past
- **What do we need:**
  - some information about the available items such as the genre ("content")
  - some sort of *user profile* describing what the user likes (the preferences)
- **The task:**
  - learn user preferences
  - locate/recommend items that are "similar" to the user preferences



What is the "content"?

1. Most CB-recommendation techniques were applied to recommending text documents
   - ✓ Like web pages or newsgroup messages for example
2. Content of items can also be represented as text documents
   - ✓ With textual descriptions of their basic characteristics
   - ✓ Structured: Each item is described by the same set of attributes

Title	Genre	Author	Type	Price	Keywords
The Night of the Gun	Memoir	David Carr	Paperback	29.90	Press and journalism, drug addiction, personal memoirs, New York
The Lace Reader	Fiction, Mystery	Brunonia Barry	Hardcover	49.90	American contemporary fiction, detective, historical
Into the Fire	Romance, Suspense	Suzanne Brockmann	Hardcover	45.90	American fiction, murder, neo-Nazism

   - ✓ Unstructured: free-text description

Content representation and item similarities

- **Item representation**

Title	Genre	Author	Type	Price	Keywords
The Night of the Gun	Memoir	David Carr	Paperback	29.90	Press and journalism, drug addiction, personal memoirs, New York
The Lace Reader	Fiction, Mystery	Brunonia Barry	Hardcover	49.90	American contemporary fiction, detective, historical
Into the Fire	Romance, Suspense	Suzanne Brockmann	Hardcover	45.90	American fiction, murder, neo-Nazism

- **User profile**

Title	Genre	Author	Type	Price	Keywords
…	Fiction	Brunonia, Barry, Ken Follett	Paperback	25.65	Detective, murder, New York

$keywords(b_j)$ describes Book b_j with a set of keywords

- **Simple approach**
  - ✓ Compute the similarity of an unseen item with the user profile based on the keyword overlap (e.g. using the Dice coefficient)

$$\frac{2 \times |keywords(b_i) \cap keywords(b_j)|}{|keywords(b_i)| + |keywords(b_j)|}$$

  - ✓ Or use and combine multiple metrics

Item Representation: Term-Frequency - Inverse Document Frequency ($TF - IDF$)

1. Simple keyword representation has its problems
   - in particular when automatically extracted as
     - not every word has similar importance
     - longer documents have a higher chance to have an overlap with the user profile
2. Standard measure: TF-IDF
   - Encodes text documents in multi-dimensional Euclidian space
     - weighted term vector
   - TF: Measures, how often a term appears (density in a document)
     - assuming that important terms appear more often
     - normalization has to be done in order to take document length into account
   - IDF: Aims to reduce the weight of terms that appear in all documents

TF-IDF: a quick reminder

1. Given a keyword i and a document j
2. $TF(i,j)$
   - ✓ term frequency of keyword i in document j
3. $IDF(i)$
   - ✓ inverse document frequency calculated as $\boldsymbol{IDF(i) = log \frac{N}{n(i)}}$
     - ✓ N : number of all recommendable documents
     - ✓ $n(i)$: number of documents from N in which keyword i appears
4. $TF - IDF$
   - ✓ is calculated as: $\boldsymbol{TF\text{-}IDF(i,j) = TF(i,j) * IDF(i)}$

Example TF-IDF representation

1. Each document is now represented by a real-valued vector of $TF\text{-}IDF$ weights $\in \mathbb{R}^{|v|}$

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth
Antony	5.25	3.18	0	0	0	0.35
Brutus	1.21	6.1	0	1	0	0
Caesar	8.59	2.54	0	1.51	0.25	0
Calpurnia	0	1.54	0	0	0	0
Cleopatra	2.85	0	0	0	0	0
mercy	1.51	0	1.9	0.12	5.25	0.88
worser	1.37	0	0.11	4.15	0.25	1.95

Example taken from http://informationretrieval.org

Improving the vector space model

1. Vectors are usually long and sparse
2. Remove stop words
   - ✓ They will appear in nearly all documents.
   - ✓ e.g. "a", "the", "on", …
3. Use stemming
   - ✓ Aims to replace variants of words by their common stem
   - ✓ e.g. "went" $\Longrightarrow$ "go", "stemming" $\Longrightarrow$ "stem", …
4. size cut-offs
   - ✓ only use top n most representative words to remove "noise" from data
   - ✓ e.g. use top 100 words

Improving the vector space model II

1. Use lexical knowledge, use more elaborate methods for feature selection
   - ✓ Remove words that are not relevant in the domain
2. Detection of phrases as terms
   - ✓ More descriptive for a text than single words
   - ✓ e.g. "United Nations"
3. Limitations
   - ✓ semantic meaning remains unknown
   - ✓ example: usage of a word in a negative context
     - "there is nothing on the menu that a vegetarian would like.."
     - The word "vegetarian" will receive a higher weight
     - unintended match with a user interested in vegetarian restaurants

Cosine similarity

1. Usual similarity metric to compare vectors: Cosine similarity (angle)
   - ✓ Cosine similarity is calculated based on the angle between the vectors
     - $sim(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| * |\vec{b}|}$
2. Adjusted cosine similarity
   - ✓ take average user ratings into account ($\bar{r}_u$), transform the original ratings
   - ✓ U: set of users who have rated both items a and b
   - ✓ $sim(\vec{a}, \vec{b}) = \frac{\sum_{u \in U}(r_{u,a} - \bar{r}_u)(r_{u,b} - \bar{r}_u)}{\sqrt{\sum_{u \in U}(r_{u,a} - \bar{r}_u)^2} \sqrt{\sum_{u \in U}(r_{u,b} - \bar{r}_u)^2}}$

Recommending items: k-NN

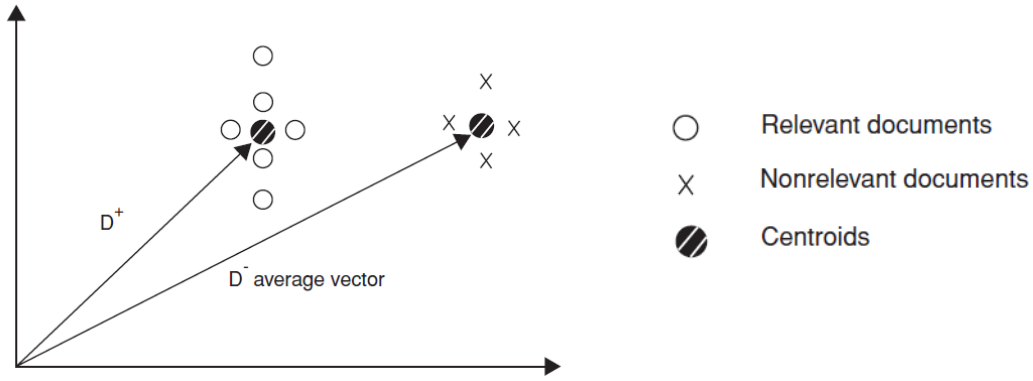
1. Simple method: nearest neighbors
   - ✓ Given a set of items D already rated by the user (like/dislike)
     - ✓ Either explicitly via user interface
     - ✓ Or implicitly by monitoring user's behavior
   - ✓ Find the n nearest neighbors in D of an not-yet-seen item i
     - ✓ Use similarity measures (like cosine similarity) to capture similarity of two items
   - ✓ Take these neighbors to predict a rating for i
     - ✓ e.g. $k = 5$ most similar items to i
       4 of k items were liked by current user ⇨ item i will also be liked by this user
2. Good to model short-term interests / follow-up stories
3. Used in combination with method to model long-term preferences

Rocchio Algorithm

1. Categorization Problem: Items classified in D+ (liked) and D- (disliked)
2. Items represented Vector Space Model
3. TF-IDF word-weighting scheme
4. Learning by combining document vectors (positive and negative examples) into a prototype vector for each class (preference)
   - ✓ Control parameters for setting the relative importance of positive and negative examples
5. Items considered interesting within a certain distance from the prototype
   - ✓ Cosine similarity measure

Rocchio details

1. Item collections D^+ (liked) and D^- (disliked)
   - ✓ Calculate prototype vector for these categories



Probabilistic methods

- **Recommendation as classical text classification problem**
  - long history of using probabilistic methods
- **Simple approach:**
  - 2 classes: liked/disliked
  - simple Boolean document representation
  - calculate probability that document is liked/disliked based on Bayes theorem

Doc-ID	recommender	intelligent	learning	school	Label
1	1	1	1	0	1
2	0	0	1	1	0
3	1	1	0	0	1
4	1	0	1	1	1
5	0	0	0	1	0
6	1	1	0	0	?

$$
\begin{aligned}
&P(Label = 1 | Doc - ID = 6) \\
&= P(recommender = 1 | Label = 1) \\
&\times P(intelligent = 1 | Label = 1) \\
&\times P(learning = 0 | Label = 1) \\
&\times P(school = 0 | Label = 1) \\
&= {}^{3}/_{3} \times {}^{2}/_{3} \times {}^{1}/_{3} \times {}^{2}/_{3} \approx 0.149
\end{aligned}
$$

$$
\begin{aligned}
&P(Label = 0 | Doc - ID = 6) \\
&= P(recommender = 1 | Label = 0) \\
&\times P(intelligent = 1 | Label = 0) \\
&\times P(learning = 0 | Label = 0) \\
&\times P(school = 0 | Label = 0) \\
&= {}^{0}/_{2} \times {}^{0}/_{2} \times {}^{1}/_{2} \times {}^{1}/_{2} = 0
\end{aligned}
$$

Improvements

1. Side note: Conditional independence of events does in fact not hold
   - ✓ "New York", "Hong Kong"
   - ✓ Still, good accuracy can be achieved
2. Boolean representation simplistic
   - ✓ positional independence assumed
   - ✓ keyword counts lost
3. More elaborate probabilistic methods
   - ✓ e.g., estimate probability of term v occurring in a document of class C by relative frequency of v in all documents of the class

Limitations of content-based recommendation methods

1. Keywords alone may not be sufficient 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 extractable (multimedia)
2. Ramp-up phase required
   - ✓ Some training data is still required
   - ✓ Web 2.0: Use other sources to learn the user preferences
3. Overspecialization
   - ✓ Algorithms tend to propose "more of the same"

Possible solutions

PROBLEMS	CHALLENGES	RESEARCH DIRECTIONS
Limited Content Analysis	Beyond keywords: novel strategies for the representation of items and profiles	▪ Semantic analysis of items by means of external knowledge sources (WordNet, Wikipedia,…)
Overspecialization	Defeating homophily: diversification of results	▪ Knowledge Infusion from open knowledge sources

Semantic Analysis: beyond keywords

Semantic Analysis =

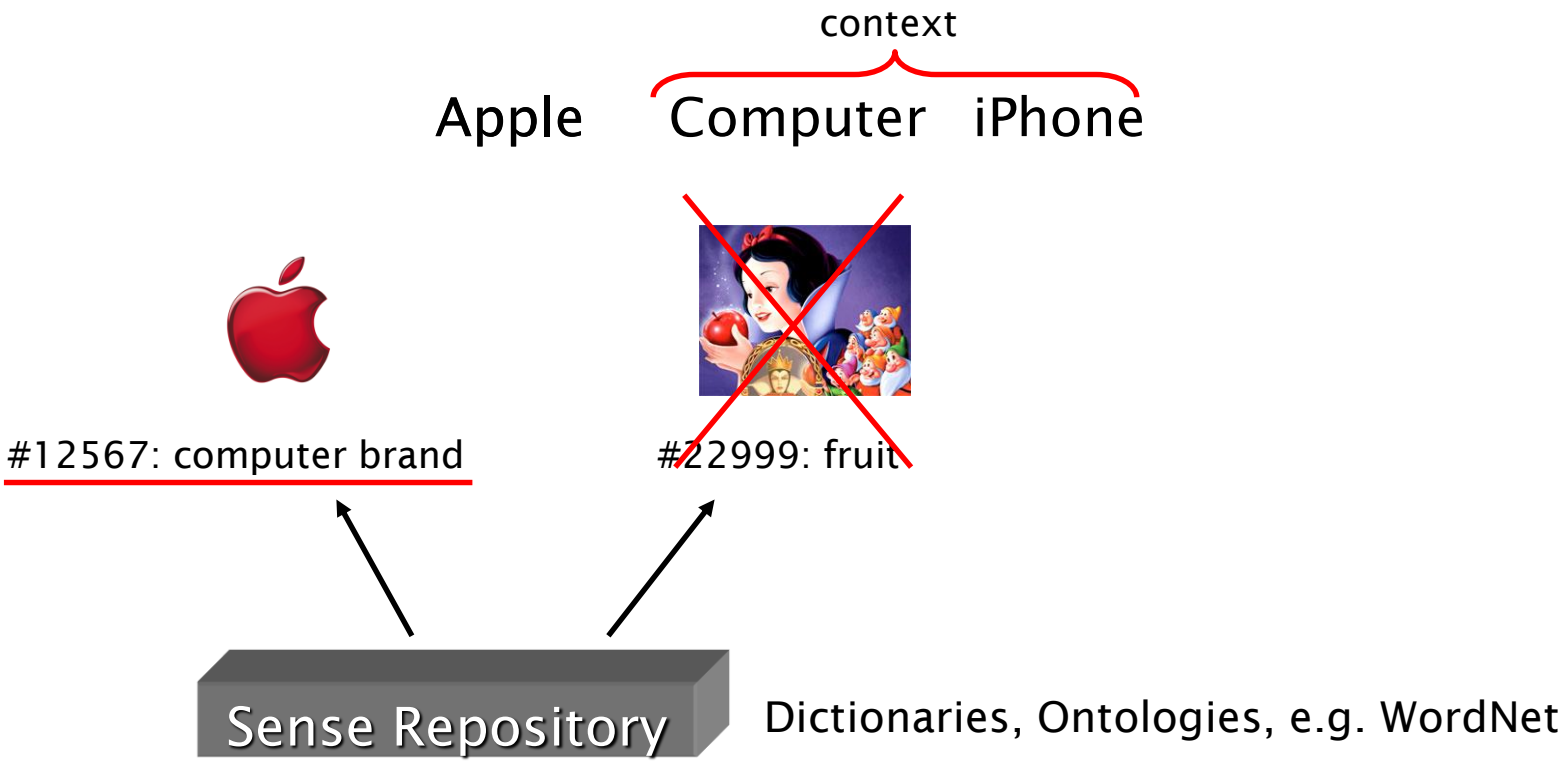
1. Semantics: concept identification in text-based representations through advanced NLP techniques → "*beyond keywords*"

+

2. Personalization: representation of user information needs in an effective way → "*deep (high-accuracy) user profiles*"

Beyond keywords - Word Sense Disambiguation (WSD): from words to meanings

1. WSD selects the proper meaning (*sense*) for a word in a text by taking into account the context in which that word occurs



[Basile07] P. Basile, M. Degemmis, A. Gentile, P. Lops, and G. Semeraro. UNIBA: JIGSAW algorithm for Word Sense Disambiguation. In *Proceedings of the 4th ACL 2007 International Workshop on Semantic Evaluations (SemEval-2007), Prague, Czech Republic*, pages 398–401. Association for Computational Linguistics, June 23-24, 2007.

Word Sense Disambiguation (WSD)

1. The different meanings of polisemous words are known as *senses*
2. Only one sense of a polisemous word is used in a specific linguistic context. The context determines the correct sense
3. The process of deciding which sense is used in a specific context is called Word Sense Disambiguation

Approaches to WSD

- Knowledge-based: uses *Machine Readable Dictionaries*
- Corpus-based: uses *sense-tagged corpus*

WordNet

1. Lexical reference database whose design is inspired by current psycholinguistic theories of human lexical memory
   - ✓ The work started in 1985 by a group of psychologists and linguists at Princeton University
2. English *nouns*, *verbs*, *adverbs* and *adjectives* are organized into SYNonym SETs, each representing one underlying lexical concept
3. Relations among synsets can be used to engineer a change of representation in text data by transforming vectors of words into vectors of word meanings
   - ✓ The synonymy relation can be used to map words with similar meanings together
   - ✓ Hypernymy (corresponding to the IS-A relation) can be used to generalize noun and verb meanings to a higher level of abstraction

wordnet.princeton.edu/

The Lexical Matrix

WordNet

a lexical database for the English language

cognitive science laboratory | princeton university | 221 nassau st. | princeton, nj 08542

Synonym word forms: SYNSET

Word Meanings	***Word Forms***					
	F_1	F_2	F_3	…	…	F_n
M_1	E(1,1)	E(2,1)				
M_2		E(2,2)	E(3,2)			
M_3						
$M_{…}$						
M_m						E(m,n)

Mapping between word forms e word meanings

The Lexical Matrix

WordNet — a lexical database for the English language

cognitive science laboratory | princeton university | 221 nassau st. | princeton, nj 08542

the word form is polysemous: WSD needed

Word Meanings	***Word Forms***					
	F_1	F_2	F_3	…	…	F_n
M_1	E(1,1)	E(2,1)				
M_2		E(2,2)	E(3,2)			
M_3						
$M_{...}$						
M_m						E(m,n)

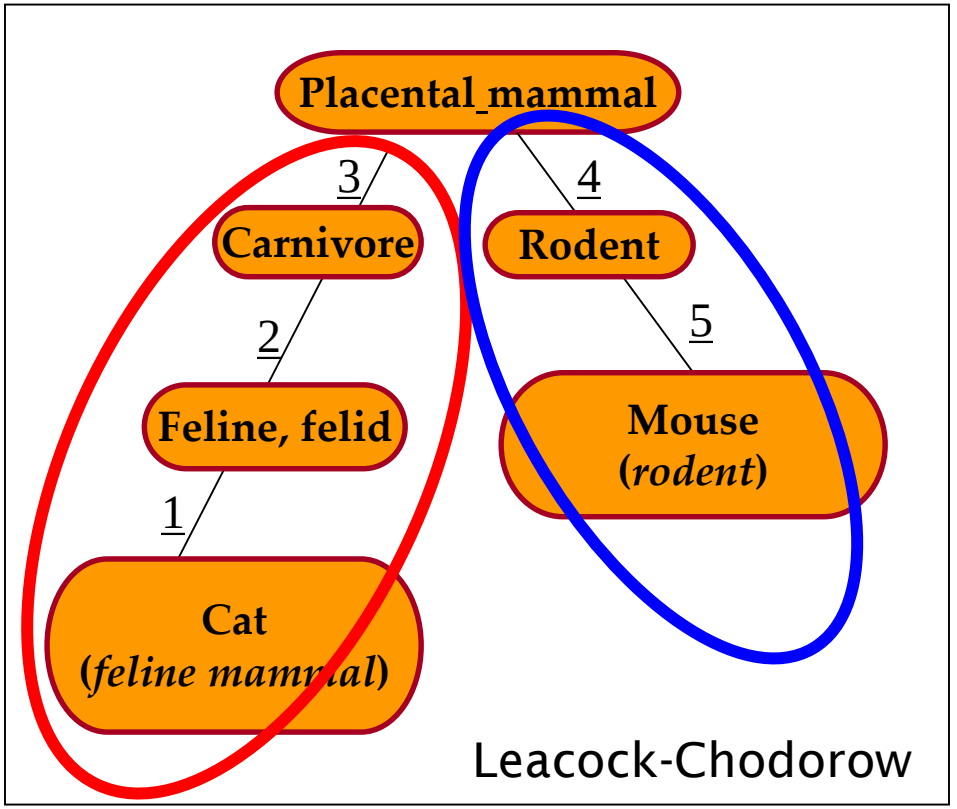
Mapping between word forms e word meanings

Synset Semantic Similarity

```
24: function SINSIM(a, b)          ▷ The similarity of the synsets a and b
25:     N_p ← the number of nodes in path p from a to b
26:     D ← maximum depth of the taxonomy          ▷ In WordNet 1.7.1 D = 16
27:     r ← −log(N_p/2D)
28:     return r
29: end function
```

SINSIM(cat,mouse) =

-log(5/32)=0.806

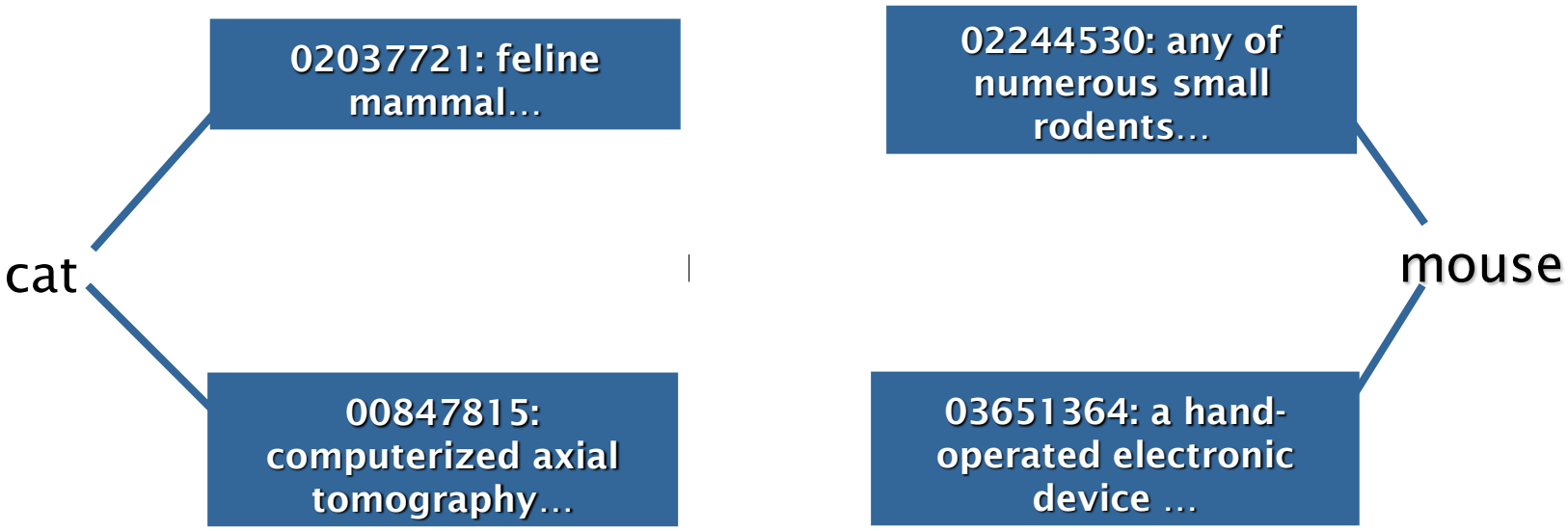


Cat-Mouse Disambiguation

"The white cat is hunting the mouse"

w = cat

C = {mouse}

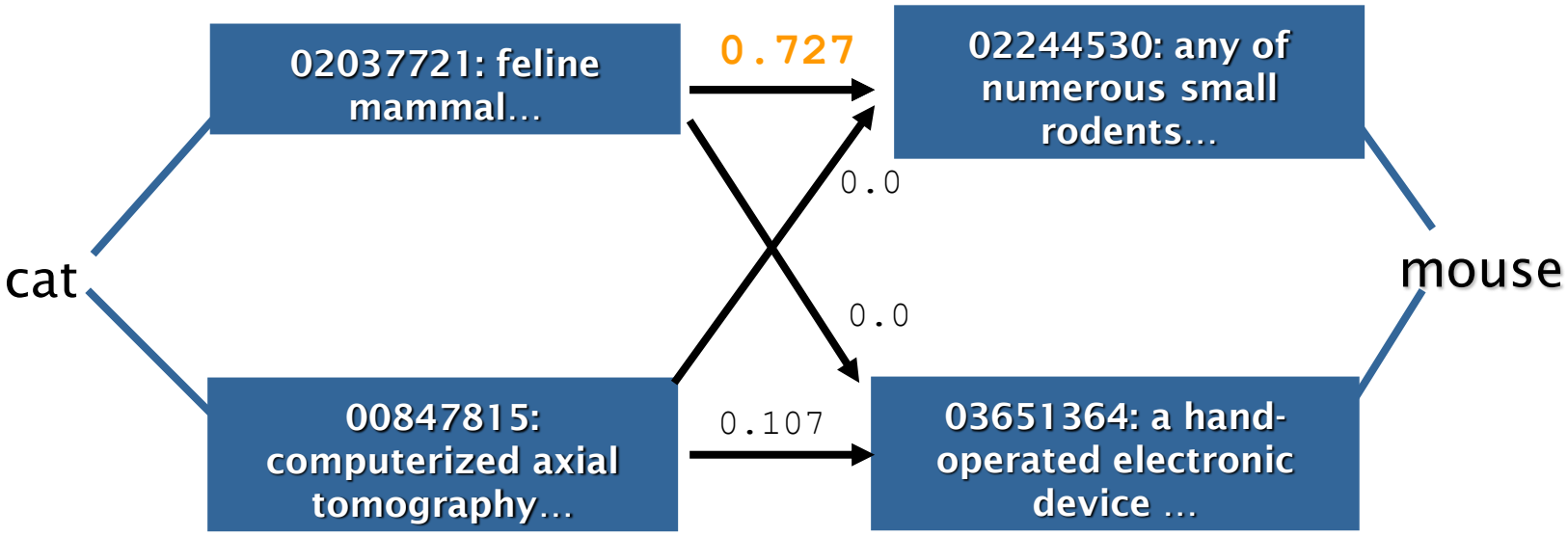


Cat-Mouse Disambiguation

"The white cat is hunting the mouse"

w = cat

C = {mouse}



Sense-based Profiles

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…

USER PROFILE

#12387	0.03
apple	0.13
AI	0.15
…	

MULTI-WORD CONCEPTS

Sense-based Profiles

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…

USER PROFILE

#12387	0.18
apple	0.13
#12387	0.15
…	

SYNONYMY

Sense-based Profiles

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…

SEMANTIC USER PROFILE
sense identifiers rather than keywords

USER PROFILE

#12387	0.18
#12567	0.13
…	

POLYSEMY

[Degemmis07] M. Degemmis, P. Lops, and G. Semeraro. A Content-collaborative Recommender that Exploits WordNet-based User Profiles for Neighborhood Formation. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research (UMUAI)*, 17(3):217–255, Springer Science + Business Media B.V., 2007.

[Semeraro07] G. Semeraro, M. Degemmis, P. Lops, and P. Basile. Combining Learning and Word Sense Disambiguation for Intelligent User Profiling. In M. M. Veloso, editor, *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007* , pages 2856–2861. Morgan Kaufmann, 2007.

Advantages of Sense-based Representations

1. Semantic matching
   - ✓ computing semantic relatedness rather than string matching (e.g., by using similarity measures between WordNet synsets) [Pedersen04]
2. Senses are inherently multilingual
   - ✓ Concepts remain the same across different languages, while terms used for describing them in each specific language change
3. Improving transparency
   - ✓ matched concepts can be used to justify suggestions
4. Collaborative Filtering could benefit too
   - ✓ finding better neighbors: similar users discovered by looking at profile overlap even if they did not rate exactly the same items
   - ✓ semantic profiles succeed where Pearson's correlation coefficient fail

[Pedersen04] Pedersen, Ted and Patwardhan, Siddharth, and Michelizzi, Jason. WordNet:Similarity - Measuring the Relatedness of Concepts. In *Proceedings of the Nineteenth National Conference on Artificial Intelligence* (AAAI-2004), pp. 1024-1025, San Jose, CA, July, 2004.

A Sense-based Recommender System

- Content-based recommender developed at Univ. of Bari [Semeraro07]
  - ✓ learns a probabilistic model of the interests of the user from textual descriptions of items
  - ✓ user profile = binary text classifier able to categorize items as interesting (LIKES) or not (DISLIKES)
  - ✓ a-posteriori probabilities as classification scores for LIKES and DISLIKES

[Semeraro07] G. Semeraro, M. Degemmis, P. Lops, and P. Basile. Combining Learning and Word Sense Disambiguation for Intelligent User Profiling. In M. M. Veloso, editor, *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 2856–2861, Morgan Kaufmann, 2007.

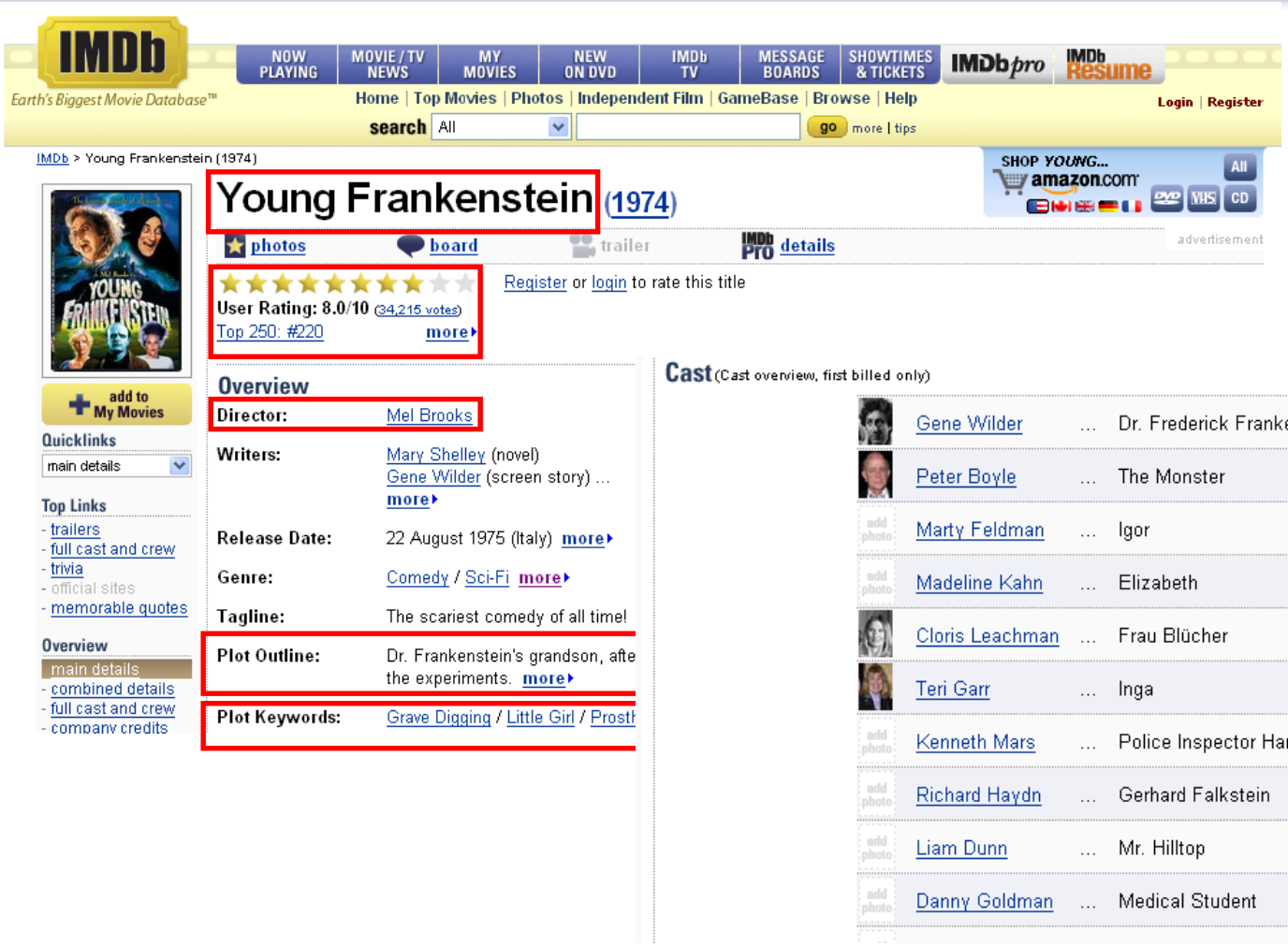
Movie Recommending

(1/2)

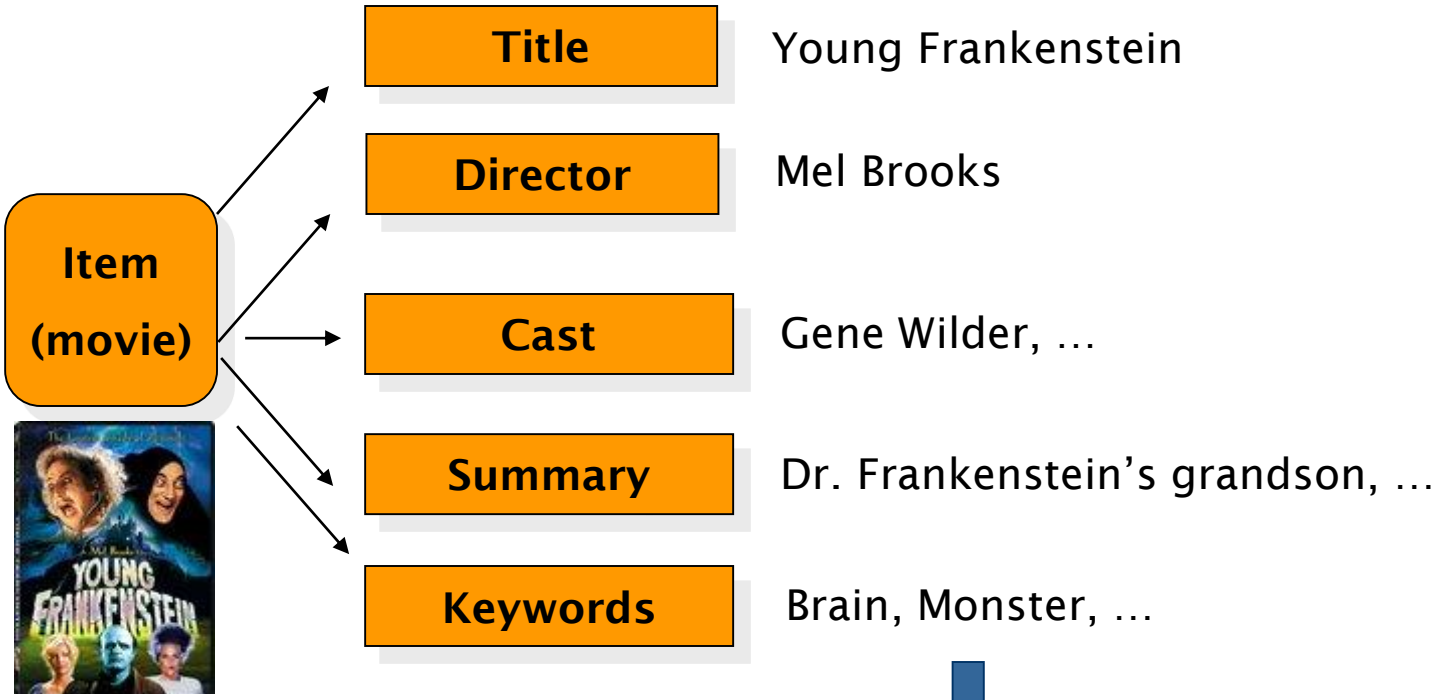
Movie Recommending (1/2)

Movie Recommending (1/2)

Movie Recommending (1/2)



Movie Recommending (2/2)



User Rating: ★★★

Tokenization +
Stopword +
(Stemming)

→ **BOW representation**

POS +
WSD

→ **BOS representation**

Example of Keyword-based User Profile

User ID: 6	Category: dummy	Class Priors:	P(YES)= 0.5333333	P(NO)= 0.4666666

Slot: **abstractContent**

Feature	Strength
entiti	3.6138782
repositori	2.8947555
easi	2.8947555
reusabl	2.8947555
upper-level	2.8947555
way	2.8947555
avail	2.8947555
area	2.5893739
kim	2.4626222
seamless	2.1475411
subtasks:	2.1475411

$$strength(t_k, s_m) = \log \frac{P(t_k \mid c_+, s_m)}{P(t_k \mid c_-, s_m)}$$

Features are keywords

Example of Sense-based User Profile

User ID: 6	Category: dummy	Class Priors:	P(YES)= 0.5333333	P(NO)= 0.4666666

Slot: **abstractContent**

Feature	Strength
1742	3.6145387
2268652	2.8954161
2766412	2.8954161
2223910	2.8954161
4415376	2.8954161
5655492	2.5376664
5552847	2.3181007
2311478	2.1482017
5309075	2.1482017
1636312	2.1482017

$$strength(t_k, s_m) = \log \frac{P(t_k \mid c_+, s_m)}{P(t_k \mid c_-, s_m)}$$

is computed on synsets instead of keywords

Features are WordNet synsets

Recommendations based on User Profiles

User Profile

User ID: 6 | Category: dummy | Class Priors: P(YES)= 0.5333333 | P(NO)= 0.4666666

Slot: **abstractContent**

Feature	Strength
1742	3.6145387
2268652	2.8954161
2766412	2.8954161
2223910	2.8954161
4415376	2.8954161
5655492	2.5376664
5552847	2.3181007
2311478	2.1482017
5309075	2.1482017
1636312	2.1482017

Naive Bayes

Text Classifier

0.78

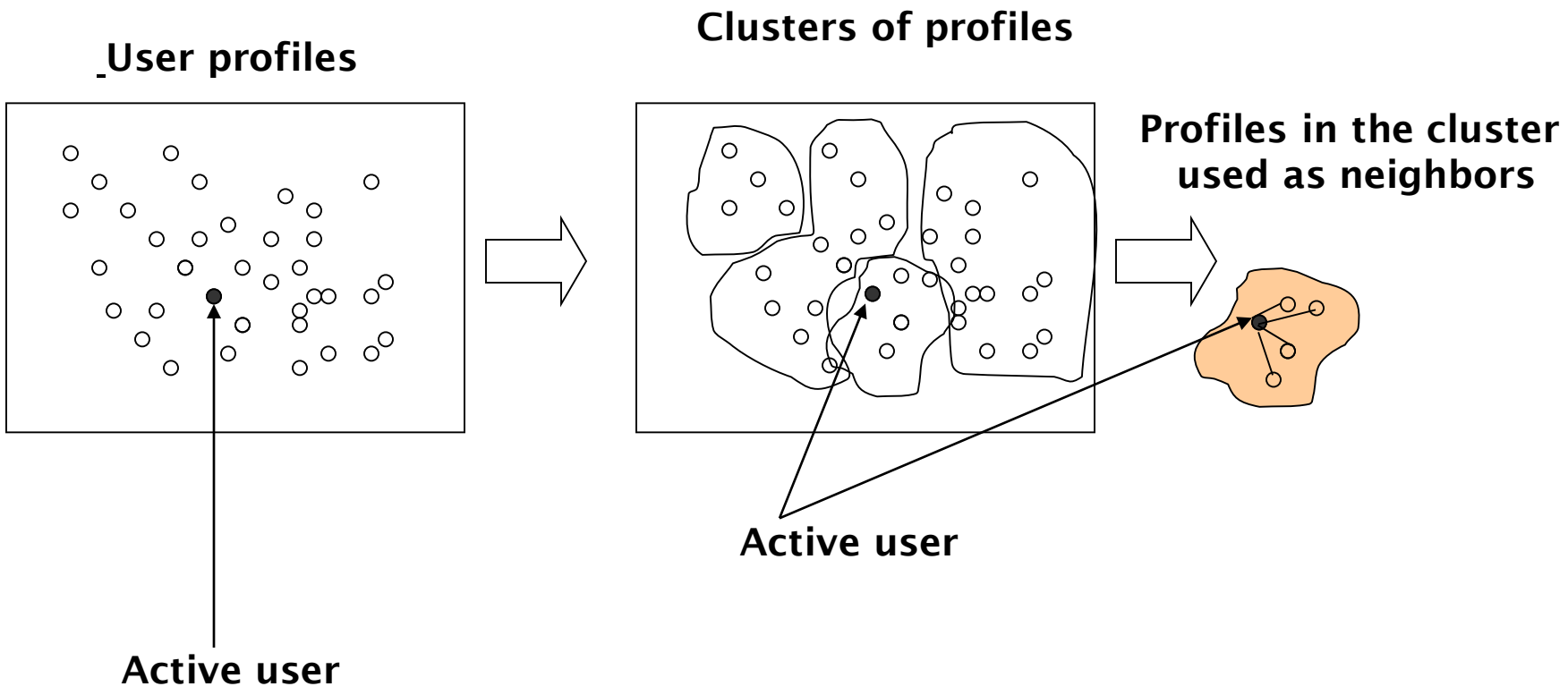
Classification Score

P(user-likes|E.T.)

Probability that the owner of the profile will like the document

Example of application of Sense-based profiles in a hybrid CB-CF recommender

Clustering of sense-based profiles



How to find overlap between users?

	Spiderman	Superman	Batman	X-Men	Shining	The Ring
Joe	5		4		1	
Howard		4		5		1
Tom					4	4
John				5	4	

- Who are the Joe's neighbors?
  - ✓ The strategy based on the co-rated items fails
  - ✓ Low accuracy with sparse user-item matrices
- Neighbors selected by means of synset-based profiles
  - ✓ Clusters of user profiles for the neighborhood formation
  - ✓ Overcoming the sparsity problem

How to find overlap between users?

	Spiderman	Superman	Batman	X-Men	Shining	The Ring
Joe	5		4		1	
Howard		4		5		1
Tom					4	4
John				5	4	

Joe's profile	Howard's profile	Tom's profile	John's profile
superhero	superhero	drama	sci-fi
action	action	horror	horror
fantasy	fantasy	mistery	mistery
adventure	sci-fi	ghost	fantasy

- ✓ Clusters of user profiles for the neighborhood formation
- ✓ Overcoming the sparsity problem

How to find overlap between users?

	Spiderman	Superman	Batman	X-Men	Shining	The Ring
Joe	5		4		1	
Howard		4		5		1
Tom					4	4
John				5	4	

Joe's profile	Howard's profile	Tom's profile	John's profile
superhero	superhero	drama	sci-fi
action	action	horror	horror
fantasy	fantasy	mistery	mistery
adventure	sci-fi	ghost	fantasy

- ✓ Clusters of user profiles for the neighborhood formation
- ✓ Overcoming the sparsity problem

Discussion & summary

1. In contrast to collaborative approaches, content-based techniques do not require user community in order to work
2. Presented approaches aim to learn a model of user's interest preferences based on explicit or implicit feedback
   - ✓ Deriving implicit feedback from user behavior can be problematic
3. Evaluations show that a good recommendation accuracy can be achieved with help of machine learning techniques
   - ✓ These techniques do not require a user community
4. Danger exists that recommendation lists contain too many similar items
   - ✓ All learning techniques require a certain amount of training data
   - ✓ Some learning methods tend to overfit the training data
5. Pure content-based systems are rarely found in commercial environments

Literature

1. [Michael Pazzani and Daniel Billsus 1997] Learning and revising user profiles: The identification of interesting web sites, Machine Learning 27 (1997), no. 3, 313-331.
1. [Soumen Chakrabarti 2002] Mining the web: Discovering knowledge from hypertext data, Science & Technology Books, 2002.