Recommender Systems: Content-based, Knowledge-based, Hybrid

Radek Pelánek

2014

Today

- lecture, basic principles:
  - content-based
  - knowledge-based
- discussion – projects
  - brief presentation of your projects
  - application of notions to projects

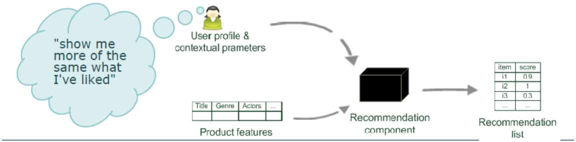
Content-based vs Collaborative Filtering

- collaborative filtering: *“recommend items that similar users liked”*
- content based: *“recommend items that are similar to those the user liked in the past”*

Content-based Recommendations

we need explicit (cf latent factors in CF):

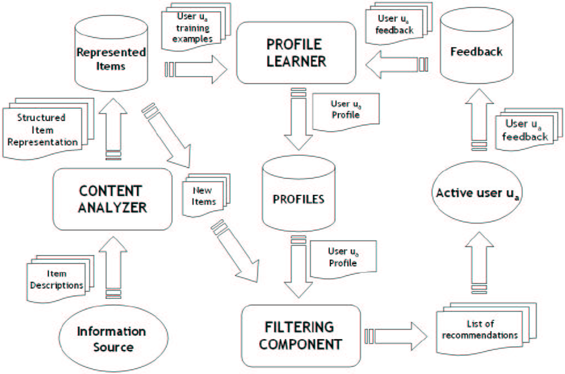
- information about items (e.g., genre, author)
- user profile (preferences)


"show me more of the same what I've liked"
User profile & contextual prameters
Title | Genre | Actors | ...
Product features
Recommendation component
Recommendation list

item	score
i1	0.9
i2	1
i3	0.3
...	...

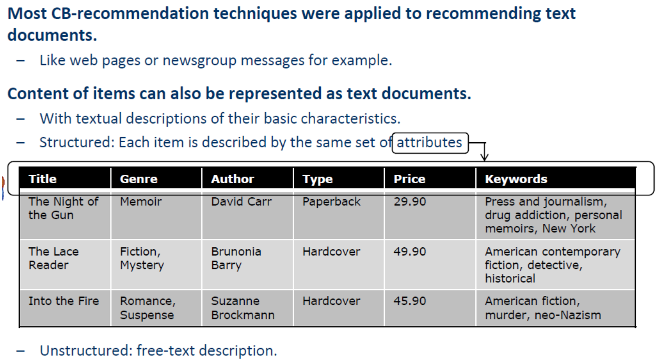
Recommender Systems: An Introduction (slides)

Architecture of a Content-Based Recommender



Handbook of Recommender Systems

Content



Most CB-recommendation techniques were applied to recommending text documents.

- Like web pages or newsgroup messages for example.

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.

Recommender Systems: An Introduction (slides)

Content: Multimedia

- manual anotation
  - songs, hundreds of features
  - Pandora, http://www.pandora.com
  - Music Genome Project
  - experts, 20-30 minutes per song
- automatic techniques – signal processing

User Profile

- explicitly specified by user
- automatically learned

Similarity: Keywords

sets of keywords A, B

- Dice coefficient: $\frac{2\cdot|A\cap B|}{|A|+|B|}$
- Jaccard coefficient: $\frac{|A\cap B|}{|A\cup B|}$

Term Frequency – Inverse Document Frequency

- keywords (particularly automatically extracted) – disadvantages:
    - importance of words ("course" vs "recommender")
    - length of documents
- TF-IDF – standard technique in information retrieval
    - Term Frequency – how often term appears in a particular document (normalized)
    - Inverse Document Frequency – how often term appears in all documents

Term Frequency – Inverse Document Frequency

keyword (term) t, document d

- $TF(t,d) =$ frequency of t in d / maximal frequency of a term in d
- $IDF(t) = \log(N/n_t)$
  - N – number of all documents
  - n_t – number of documents containing t
- $TFIDF(t,d) = TF(t,d) \cdot IDF(t)$

Improvements

all words – long, sparse vectors

- common words, stop words (e.g., “a”, “the”, “on”)
- stemming (e.g., “went” → “go”, “university” → “univers”)
- cut-offs (e.g., *n* most informative words)
- phrases (e.g., “United Nations”)

Limitations

- semantic meaning unknown
- example – use of words in negative context

steakhouse description: "there is nothing on the menu that a vegetarian would like..." ⇒ *keyword "vegetarian"* ⇒ *recommended to vegetarians*

Similarity

- cosine similarity – angle between vectors
  - $sim(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|}$
- (adjusted) cosine similarity
  - normalization by subtracting average values
  - closely related to Pearson correlation coefficient

Recommendations

- nearest neighbors
- Rocchio's relevance feedback method (interactivity)

Nearest Neighbors

- k-nearest neighbors (kNN)
- predicting rating for not-yet-seen item i:
    - find k most similar items, already rated
    - predict rating based on these
- good for modeling short-term interest, “follow-up” stories

Other Methods

- probabilistic methods – Naive Bayes
- linear classifiers

Limitations of Content-Based Recommendations

- limited content analysis – content may not be automatically extractable (multimedia), missing domain knowledge, ...
- keywords may not be sufficient
- overspecialization – “more of the same”, too similar items

Content-Based vs Collaborative Filtering

- paper "Recommending new movies: even a few ratings are more valuable than metadata" (context: Netflix)
- our experience in educational domain – difficulty rating (Sokoban, countries)

Knowledge-based Recommendations

application domains:

- expensive items, not frequently purchased, few ratings
- time span important (e.g., technological products)
- explicit requirements of user

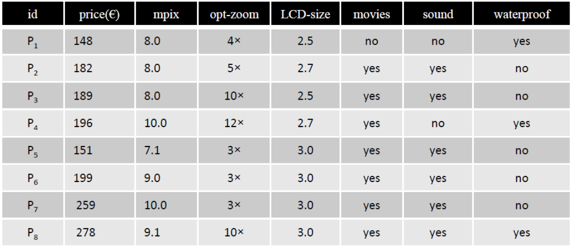
- collaborative filtering unusable – not enought data
- content based – “similarity” not sufficient

Knowledge-based Recommendations

- constraint-based
  - explicitly defined conditions
- case-based
  - similarity to specified requirements

“conversational” recommendations

Constraint-Based Recommmendations – Example



id	price(€)	mpix	opt-zoom	LCD-size	movies	sound	waterproof
P_1	148	8.0	4×	2.5	no	no	yes
P_2	182	8.0	5×	2.7	yes	yes	no
P_3	189	8.0	10×	2.5	yes	yes	no
P_4	196	10.0	12×	2.7	yes	no	yes
P_5	151	7.1	3×	3.0	yes	yes	no
P_6	199	9.0	3×	3.0	yes	yes	no
P_7	259	10.0	3×	3.0	yes	yes	no
P_8	278	9.1	10×	3.0	yes	yes	yes

Recommender Systems: An Introduction (slides)

Constraint Satisfaction Problem

- V is a set of variables
- D is a set of finite domains of these variables
- C is a set of constraints

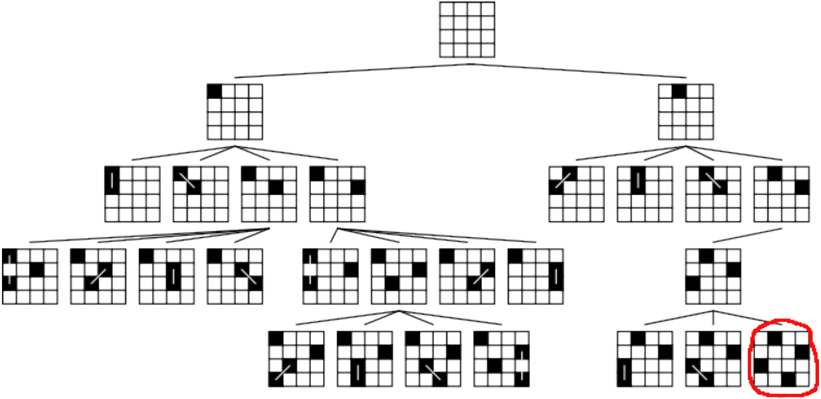
Typical problems: logic puzzles (Sudoku, N-queen), scheduling

CSP: N-queens

problem: place N queens on an $N \times N$ chess-board, no two queens threaten each other

- V – N variables (locations of queens)
- D – each domain is $\{1, \ldots, N\}$
- C – threatening

CSP Example: N-queens Problem

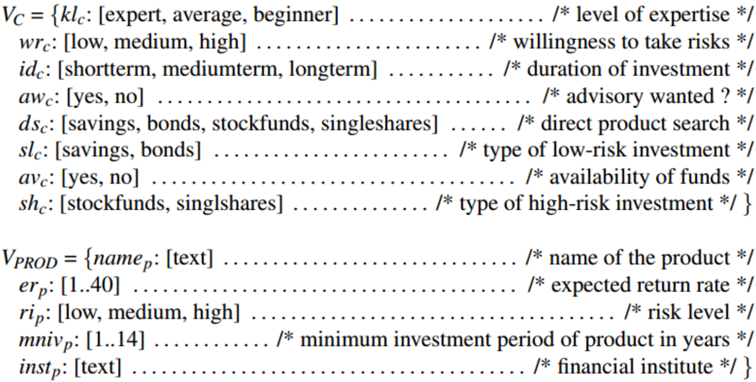


CSP Algorithms

- basic algorithm – backtracking
- heuristics
  - preference for some branches
  - pruning
  - ... many others

Recommender Knowledge Base

- customer properties V_C
- product properties V_{PROD}
- constraints C_R (on customer properties)
- filter conditions C_F – relationship between customer and product
- products C_{PROD} – possible instantiations



$V_C = \{kl_c$: [expert, average, beginner] .... /* level of expertise */

wr_c: [low, medium, high] .... /* willingness to take risks */

id_c: [shortterm, mediumterm, longterm] .... /* duration of investment */

aw_c: [yes, no] .... /* advisory wanted ? */

ds_c: [savings, bonds, stockfunds, singleshares] ...... /* direct product search */

sl_c: [savings, bonds] .... /* type of low-risk investment */

av_c: [yes, no] .... /* availability of funds */

sh_c: [stockfunds, singlshares] .... /* type of high-risk investment */ $\}$

$V_{PROD} = \{name_p$: [text] .... /* name of the product */

er_p: [1..40] .... /* expected return rate */

ri_p: [low, medium, high] .... /* risk level */

$mniv_p$: [1..14] .... /* minimum investment period of product in years */

$inst_p$: [text] .... /* financial institute */ $\}$

Recommender Systems Handbook; Developing Constraint-based Recommenders

$$
\begin{aligned}
C_R = \{&CR_1\colon wr_c = high \rightarrow id_c \neq shortterm,\\
&CR_2\colon kl_c = beginner \rightarrow wr_c \neq high\}
\end{aligned}
$$

$$
\begin{aligned}
C_F = \{&CF_1\colon id_c = shortterm \rightarrow mniv_p < 3,\\
&CF_2\colon id_c = mediumterm \rightarrow mniv_p \geq 3 \wedge mniv_p < 6,\\
&CF_3\colon id_c = longterm \rightarrow mniv_p \geq 6,\\
&CF_4\colon wr_c = low \rightarrow ri_p = low,\\
&CF_5\colon wr_c = medium \rightarrow ri_p = low \vee ri_p = medium,\\
&CF_6\colon wr_c = high \rightarrow ri_p = low \vee ri_p = medium \vee ri_p = high,\\
&CF_7\colon kl_c = beginner \rightarrow ri_p \neq high,\\
&CF_8\colon sl_c = savings \rightarrow name_p = savings,\\
&CF_9\colon sl_c = bonds \rightarrow name_p = bonds\ \}
\end{aligned}
$$

$$
\begin{aligned}
C_{PROD} = \{&CPROD_1\colon name_p = savings \wedge er_p = 3 \wedge ri_p = low \wedge mniv_p = 1 \wedge inst_p = A;\\
&CPROD_2\colon name_p = bonds \wedge er_p = 5 \wedge ri_p = medium \wedge mniv_p = 5 \wedge inst_p = B;\\
&CPROD_3\colon name_p = equity \wedge er_p = 9 \wedge ri_p = high \wedge mniv_p = 10 \wedge inst_p = B\}
\end{aligned}
$$

Recommender Systems Handbook; Developing Constraint-based Recommenders

Development of Knowledge Bases

- difficult, expensive
- specilized graphical tools
- methodology (rapid prototyping, detection of faulty constraints, ...)

Unsatisfied Requirements

no solution to provided constraints

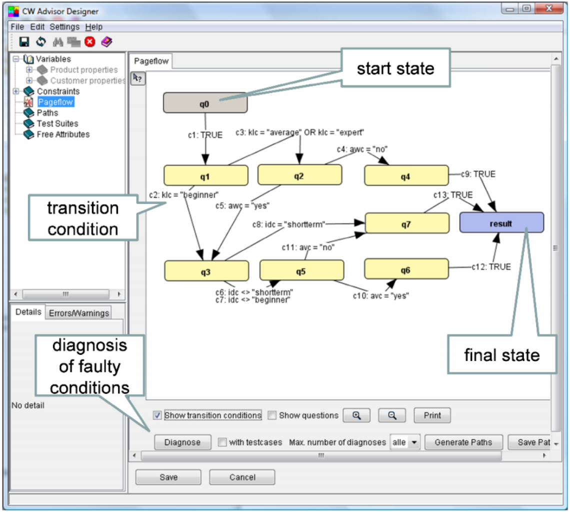
- we want to provide user at least something
- constraint relaxation
- proposing “repairs”
- minimal set of requirements to be changed

User Guidance

requirements elicitation process

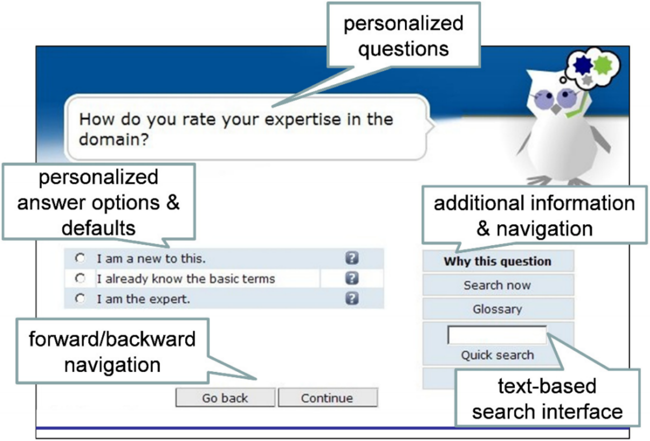
- session independent user profile (e.g., social networking sites)
- static fill-out forms
- conversational dialogs

User Guidance



Recommender Systems Handbook; Developing Constraint-based Recommenders

User Guidance



How do you rate your expertise in the domain?

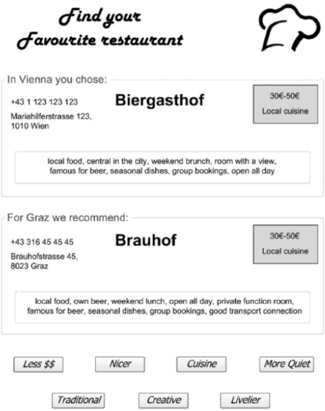
- ○ I am a new to this.
- ○ I already know the basic terms
- ○ I am the expert.

personalized questions
personalized answer options & defaults
additional information & navigation
forward/backward navigation
text-based search interface
?
Why this question
Search now
Glossary
Quick search
Go back
Continue

Fig. 6.4: Interactive and personalized preference elicitation example. Customers specify their preferences by answering questions.

Recommender Systems Handbook; Developing Constraint-based Recommenders

Critiquing



Find your Favourite restaurant

In Vienna you chose:

+43 1 123 123 123
Mariahilferstrasse 123,
1010 Wien

Biergasthof

30€-50€
Local cuisine

local food, central in the city, weekend brunch, room with a view, famous for beer, seasonal dishes, group bookings, open all day

For Graz we recommend:

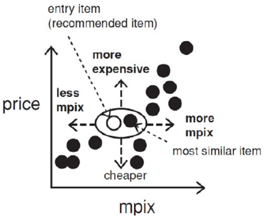
+43 316 45 45 45
Brauhofstrasse 45,
8023 Graz

Brauhof

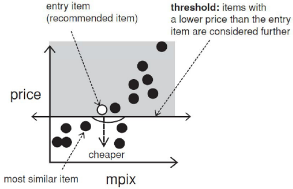
30€-50€
Local cuisine

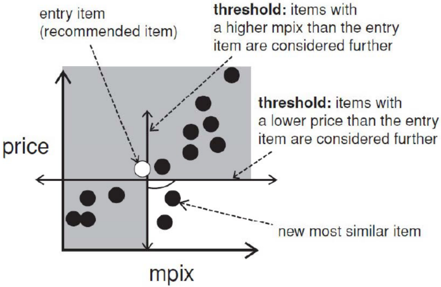
local food, own beer, weekend lunch, open all day, private function room, famous for beer, seasonal dishes, group bookings, good transport connection

Critiquing



Critique on price




entry item
(recommended item)
threshold: items with
a higher mpix than the entry
item are considered further
threshold: items with
a lower price than the entry
item are considered further
price
new most similar item
mpix

Limitations

- cost of knowledge acquisition (consider project proposals)
- accuracy of models
- indepenendence assumption for preferences

Hybrid Methods

collaborative filtering: *“what is popular among my peers”*
content-based: *“more of the same”*
knowledge-based: *“what fits my needs”*

- each has advantages and disadvantages
- hybridization – combine more techniques, avoid some shortcomings
- simple example: CF with content-based (or simple “popularity recommendation”) to overcome “cold start problem”

Hybridization Designs

- monolitic desing, combining different features
- parallel use of several systems, weighting
- pipelined invocation of different systems

Your Projects: Useful Questions

- How will the user interact with the system?
- Where/how will you obtain (meta)data about items?
- Do you already have some data about user preferences (ratings)?
- How will you collect ratings? (explicit/implicit)
- Which techniques are relevant/suitable for you project?

Project Topics – Short Text

- blog posts
- funny quotes
- recipes

Project Topics – Short Text

- blog posts
- funny quotes
- recipes

- content-based aspects: manual labels, TF-IDF
- ratings: implicit?, explicit?
- recipes – critiquing?, knowledge-based aspects ("quick preparation", "cheap ingredients", ...)

Project Topics – Products

- (board) games
- wine
- PC components

Project Topics – Products

- (board) games
- wine
- PC components

- content-based similarity
- knowledge-based aspects?, critiquing?

Project Topics – Educational

- vocabulary

Project Topics – Educational

- vocabulary

- frequencies of words
- tags?: verbs, animals, travel, ...
- rating $\sim$ testing ?