Structured Collaborative Filtering

UNIVERSIDAD AUTONOMA DE MADRID

Alejandro Bellogín, Pablo Castells
Universidad Autónoma de Madrid, Spain
{alejandro.bellogin, pablo.castells}@uam.es

Jun Wang
University College London, UK
j.wang@cs.ucl.ac.uk

UCL

Vector Space Model

Query

$$Q = \left(w_1, qf_1; \ldots; w_n, qf_n\right)$$

Document

$$D^j = \left(w_1, t_{j1}; \ldots; w_n, t_{jn}\right)$$

Similarity score

$$\mathrm{sim}\left(Q, D_j\right) \propto Q \cdot D_j = \sum_k qf_k \cdot t_{jk}$$

Collaborative Filtering

User

$$Q^u = \left(i_1, r^u_{i_1}; \ldots; i_n, r^u_{i_n}\right)$$

Item

$$I^j = \left(i_1, s^j_{i_1}; \ldots; i_n, s^j_{i_n}\right)$$

Predicted rating

$$rat(u,i) = \mathrm{sim}(u,i) = \mathrm{sim}\left(Q^u, I^j\right) \propto Q^u \cdot I^j = \sum_k r^u_{i_k} \cdot s^j_{i_k}$$

with proper normalization

Item-based CF $rat(u,i) = \dfrac{\sum_k r^u_{i_k} \cdot s^j_{i_k}}{\sum_k s^j_{i_k}}$

Extended Boolean Model

Two connectives (and, or) **are included in the query [Salton et al, 1983]:**

$$\mathbf{Q} := \begin{smallmatrix}\text{and}\\ \text{or}\end{smallmatrix}\left(p_1\right)\left[\begin{smallmatrix}\text{and}\\ \text{or}\end{smallmatrix}\left(p_2\right)[Q]^+, cw\right]$$

$$\mathbf{Q} := (qf_1, \ldots, qf_n)$$

These connectives are weighted by factors p_1 and p_2

The value cw is the clause weight

The document representation is the same:

$$\mathbf{D}^j := \left(t_{j1}, \ldots, t_{jn}\right)$$

The similarity score depends on the connective:

$$\mathrm{sim}\left(\mathbf{Q}_{and(p)}, \mathbf{D}^j\right) = 1 - \left[\frac{(qf_1(1-t_{j1}))^p + \ldots + (qf_n(1-t_{jn}))^p}{(qf_1)^p + \ldots + (qf_n)^p}\right]^{1/p}$$

$$p \in [1, +\infty)$$

$$\mathrm{sim}\left(\mathbf{Q}_{or(p)}, \mathbf{D}^j\right) = \left[\frac{(qf_1 t_{j1})^p + \ldots + (qf_n t_{jn})^p}{(qf_1)^p + \ldots + (qf_n)^p}\right]^{1/p}$$

When p = 1

$$\mathrm{sim}\left(\mathbf{Q}_{or}, \mathbf{D}^j\right) = \mathrm{sim}\left(\mathbf{Q}_{and}, \mathbf{D}^j\right)$$

Extended Vector-Space Representation for CF

Example

(a) using Boolean retrieval.

	Query items		user interest representation	
	a	b	a OR b	a AND b
item 1	1	1	1	1
item 2	1	0	1	0
item 3	0	1	1	0
item 4	0	0	0	0

(b) using extended Boolean retrieval ($p = 2$).

	Query items		user interest representation	
	a	b	a OR b	a AND b
item 1	1	1	1	1
item 2	1	0	$1/\sqrt{2}$	$1 - 1/\sqrt{2}$
item 3	0	1	$1/\sqrt{2}$	$1 - 1/\sqrt{2}$
item 4	0	0	0	0

A user shows interest for two items:

The Boolean model is too rigid: too loose or tight

Extended Boolean model: more discriminative

Representation

Only the user profile needs to change its representation

$$\mathbf{Q}^u := \begin{smallmatrix}\text{and}\\ \text{or}\end{smallmatrix}\left(p_1\right)\left[\begin{smallmatrix}\text{and}\\ \text{or}\end{smallmatrix}\left(p_2\right)[Q^u]^+, cw\right]$$

$$\mathbf{Q}^u := \left(r^u_{i_1}, \ldots, r^u_{i_n}\right)$$

Now, the predicted rating takes into account the structure of the user profile:

$$rat(u,i) = \mathrm{sim}\left(Q^u, I^j\right) \begin{cases} \mathrm{sim}\left(\mathbf{Q}_{and(p)}, I^j\right) \\ \mathrm{sim}\left(\mathbf{Q}_{or(p)}, I^j\right) \end{cases}$$

Semantics of the p-values

- p = +∞ and AND connective: a strict phrase has to be matched
- p = +∞ and OR connective: a strict thesaurus feature is used (all the terms are substitutable)
- low p and AND connective: the presence of every term is worth more (but not compulsory) than the presence of only some of them
- low p and OR connective: the presence of several terms is more important than the presence of one of them
- p = 1: both connectives are equivalent

Different retrieval models obtained:
p=1: Vector Space Model
p = +∞ : Boolean model

Now, item-based CF is simply represented as

$$\mathbf{Q}^u = \mathrm{and}(1)\left[\mathrm{and}(1)\left(\left(i_1, r^u_{i_1}\right), \ldots, \left(i_n, r^u_{i_n}\right), 1\right)\right]$$

$$\mathbf{I}^j = \left(s^j_{i_1}, \ldots, s^j_{i_n}\right)$$

User Profile Expansion

Motivation

Items that tend to occur together: movie series or by the same director

E.g., Lord of the Rings, Star Wars, etc.

These movies could be considered synonyms

Experiments

We expand every user profile with S synonym movies (i.e., the S most similar items).

Expanded terms (synonyms) are included using an inner Boolean OR (i.e., infinite p-value)

Method	P@1	P@3	P@5	P@10	NDCG@3	NDCG@5	NDCG@50	MRR
Baseline	0.002	0.004	0.005	0.007	0.002	0.003	0.009	0.027
$S = 1$, and(1), or(∞)	0.129	0.121	0.120	0.114	0.099	0.099	0.126	0.243
$S = 2$, and(1), or(∞)	0.149	0.138	0.130	0.119	0.113	0.110	0.128	0.260
$S = 5$, and(1), or(∞)	0.190	0.166	0.155	0.141	0.140	0.134	0.146	0.304
$S = 10$, and(1), or(∞)	0.197	0.171	0.161	0.147	0.146	0.140	0.151	0.313
$S = 5$, or(∞), and(1)	0.004	0.005	0.005	0.005	0.002	0.003	0.010	0.028

Not every user needs to be expanded: dynamic user profile expansion

Method	P@1	P@3	P@5	P@10	NDCG@3	NDCG@5	NDCG@50	MRR
Threshold found by median	0.183	0.167	0.157	0.145	0.137	0.132	0.158	0.302
Threshold found by average	0.186	0.165	0.158	0.146	0.137	0.133	0.157	0.303

Discussion

A significant improvement is found, compared to plain profiles (S=0, baseline).

The order of the connectives is important

Profiles built using OR + AND make little sense

Dynamic expansion outperforms static one with larger cutoffs

Future Work

Additional user profile expansion methods to set dynamically the threshold

Combination of this method with recently proposed normalization methods in [Bellogín et al, 2011]

Inferring User Profile Structure

Motivation

Subprofiles in recommendation

E.g., users A and B have similar tastes in movies but different in music

User profiles could thus be decomposed into phrases

Experiments

Each profile is defined as soft OR's of cohesive subprofiles

Each subprofile is also composed of soft OR's of item ratings

Subprofiles are found by performing clustering (using the Weka library) on genre and similarity values among items:

K-means (K=50)

X-means (worse results than K-means)

Uniform p-value for all the subprofiles

Method	P@1	P@3	P@5	P@10	NDCG@3	NDCG@5	NDCG@50	MRR
Baseline	0.002	0.004	0.005	0.007	0.002	0.003	0.009	0.027
K-means genre or(1), or(2)	0.013	0.016	0.017	0.018	0.010	0.011	0.032	0.061
K-means genre or(2), or(2)	0.005	0.006	0.006	0.006	0.003	0.004	0.011	0.030
K-means sim or(1), or(2)	0.008	0.011	0.014	0.016	0.007	0.009	0.023	0.047
K-means sim or(2), or(2)	0.006	0.006	0.006	0.008	0.004	0.004	0.012	0.031

Discussion

Structured profiles outperform plain profiles (baseline)

No significant differences for different inner p-values

Better results with subprofiles induced based on genre information

Future Work

Different p-values for each subprofile, e.g., depending on the intracluster similarity

References

[Bellogín et al, 2011] A. Bellogín, J. Wang, P. Castells. Text retrieval methods for item ranking in collaborative filtering. In ECIR 2011.

[Salton et al, 1983] G. Salton, E.A. Fox, H. Wu. Extended boolean information retrieval. Communications of the ACM 1983.

20th ACM Conference on Information and Knowledge Management (CIKM 2011)
Glasgow, UK, 24-28 October 2011