A Multi-criteria Recommender System Exploiting Aspect-based Sentiment Analysis of Users' Reviews

Cataldo Musto
Dept. of Computer Science-Univerity of Bari

Marco de Gemmis
Dept. of Computer Science-Univerity of Bari

Giovanni Semeraro
Dept. of Computer Science-Univerity of Bari

Pasquale Lops
Dept. of Computer Science-Univerity of Bari

Outline

1. Background and Motivations

2. Methodology

3. Experiment

4. Conclusion

Background and Motivations

- To evaluate different facets of the items
- Multi-criteria ratings can improve the predictive accuracy
- During the training process it is overcharging to ask users to rate the same item many times

A multi-criteria RS:

automatically learns the preferences of the target user by mining the information conveyed by the reviews he/she wrote

Methodology:

- Aspect extraction
- Sentiment analysis
- Collaborative filtering algorithm

Example

[Review]”I loved the food, the best sweet and sour soup ever! Everything was delicious and favorble. Dleivery is pretty good, nice poeople”

Overall: 4. 0/5. 0

Aspect	Sentiment score
food	4. 0
people	4. 0
food-soup	5. 0

Methodology

- Aspect Extraction
- Sentiment Analysis
- Collaborative Filtering algorithm

Aspect Extraction

A frequency-based approaches: a model that grasps the different use of language between a specific domain and a general context, rely on the simple observation that aspects and sub-aspects frequently occur as nouns.

Aspect Extraction: Aspects

Input: reviews $R = \{r_1, r_2, r_3 \ldots r_n\}$

Output: a set of quintuples $\langle r_i, a_{ij}, a_{ijk}, rel(a_{ijk}, r_i), sent(a_{ijk}, r_i) \rangle$

KL-divergence:

$$\delta_t(c_a \parallel c_b) = p(t, c_a) \log \frac{p(t, c_a)}{p(t, c_b)}$$

Aspect Extraction: Aspects

Require: review r_i , general corpus BNC, domain corpus d

Ensure: set of main aspects A and relevence scores

A={} T=nouns(r_i)

for all $t_k \in T$ do

if $\delta_{t_k}(d \,||\, BNC) > \varepsilon$ then

$a_{ij} \leftarrow t_k$

$A = A \cup \{a_{ij}\}$

$rel(a_{ij}, r_i) \leftarrow \delta_{t_k}$

end if

end for

Aspect Extraction: Sub-aspects

- **Phraseness**: the information lost following the adoption of a unigram(LM^1) in place if n-gram(LM^N) language model:

$$\varphi_P = \delta_t (LM_{fg}{}^N \parallel LM_{fg}{}^1)$$

Aspect Extraction: Sub-aspects

- **Informativeness**: the information lost when assuming that t is drawn from LM_{bg} —the background or general—rather than LM_{fg} —the foreground or domain—language model:

$$\varphi_I = \delta_t({LM_{fg}}^N \,\|\, {LM_{bg}}^N)$$

Aspect Extraction: Sub-aspects

Since the quality of the pair (aspect, sub-aspect) is conditioned by both these factors, we define the relevance weight of the pair as :

$$\varphi = (\varphi_P + \varphi_I) * \mathrm{N}(\sigma^2)$$

Aspect Extraction: Sub-aspects

Require: main aspect list A

Ensure: set of B of pairs(aspect, sub-aspect) and relevence score

B={} N=nouns(A)

for all $t_j \in N$ do

for all $t_k \in N$ do

if $\varphi > \gamma$ then

$a_{ijk} \leftarrow \langle t_j, t_k \rangle$

$B = B \cup \{a_{ijk}\}$

$rel(a_{ijk}, r_i) \leftarrow \varphi$

end if

end for

Sentiment Analysis

a sentiment score: $sent(a_{ijk}, r_i)$

A lexicon-based algorithm based on the AFINN wordlist, which contains about 2500 English words that have been manually tagged with a score that can range from positive(+5) to negtive(-5). In this case, the algorithm return a sentiment score associated to each aspect and sub-aspect extracted from the review, according to the sentiment of the piece of text the aspect is mentioned in.

Collaborative Filtering algorithm

- Multi-criteria User-to User CF.

  similarity between users u_j and u_k is calculated as the opposite of the distance $dist(u_j, u_k)$:

$$dist(u_j, u_k) = \frac{1}{\left| I(u_j, u_k) \right|} + \sum_{i \in I(u_j, u_k)} d(R(u_j, i), R(u_k, i))$$

$$d(R(u_j, i), R(u_k, i)) = \sqrt{\sum_{a=1}^{n} \left| R_a(u_j, i) - R_a(u_k, i) \right|^2}$$

Collaborative Filtering algorithm

- Multi-criteria User-to User CF.

  rating prediction of user u on item i is calculated on the top-k neighbors of the user, as:

$$R(u,i) = \sum_{j=1}^{k} \frac{sim(u,u_j) * R(u_j,i)}{\sum_{j=1}^{k} \left| sim(u,u_j) \right|}$$

Collaborative Filtering algorithm

- Multi-criteria Item-to-Item CF.

  similarity between items i_j and i_k is calculated as the opposite of the distance $dist(i_j, i_k)$:

$$dist(i_j, i_k) = \frac{1}{|U(i_j, i_k)|} + \sum_{u \in U(i_j, i_k)} d(R(u, i_j), R(u, i_k))$$

$$d(R(u, i_j), R(u, i_k)) = \sqrt{\sum_{c=1}^{n} \left| R_c(u, i_j) - R_c(u, i_k) \right|^2}$$

Collaborative Filtering algorithm

- Multi-criteria Item-to-Item CF.
  rating prediction of user u on item i is calculated as it happened in the User-to-User scenario:

$$R(u,i) = \frac{\sum sim(i,j) * R(u,i)}{\sum\limits_{u \in U(i,j)} \left| sim(i,j) \right|}$$

Experiment

1. Which combination of the parameters of our framework for aspect extraction and sentiment analysis leads to the best predictive accuracy of our multi-criteria RS?
2. How does our algorithm perform , when compared to a single-criterai CF algorithm as well as to other state-of-art Cf techniques based on matrix factorization?

state-of-the-art datasets

	Yelp	TripAdvisor	Amazon
Users	45981	536952	826773
Items	11537	3945	50210
Ratings/Reviews	229906	796958	1324759
Sparsity	99.95%	99.96%	99.99%

MAE

Table 2: Results of Experiment 1 for *multi-criteria user-based CF*. The best-performing configuration is highlighted in bold.

Configuration			Dataset		
#neigh.	**#asp.**	**sub-asp**	**Yelp**	**TripAdvisor**	**Amazon**
10	10	Y	**0.8362**	**0.7111**	0.6464
10	10	N	0.8410	0.7564	0.6335
10	50	Y	0.8410	0.7269	0.6346
10	50	N	0.8364	0.8007	**0.6276**
30	10	Y	0.8461	0.7677	0.7120
30	10	N	0.8473	0.7722	0.7122
30	50	Y	0.8474	0.7743	0.7101
30	50	N	0.8494	0.8003	0.7140
80	10	Y	0.8579	0.7971	0.7584
80	10	N	0.8592	0.7953	0.7554
80	50	Y	0.8590	0.7907	0.7544
80	50	N	0.8597	0.7995	0.7554

Table 3: Results of Experiment 1 for *multi-criteria item-based CF*. The best-performing configuration is highlighted in bold.

Configuration		Dataset		
#asp.	**sub-asp**	**Yelp**	**TripAdvisor**	**Amazon**
10	Y	**0.8640**	**0.8245**	**0.8110**
10	N	0.8643	0.8252	0.8117
50	Y	0.8641	0.8254	0.8118
50	N	0.8648	0.8260	0.8124

Table 5: Results of Experiment 2. The best-performing configuration is highlighted in bold. Next to each baseline we report the similarity measure used in that specific run. N.A. refers to not available results.

	Dataset		
Configuration	**Yelp**	**Tripadvisor**	**Amazon**
Multi-U2U	**0.8362**	**0.7111**	**0.6276**
U2U-Euclidean	0.8860	0.8337	0.7254
U2U-Pearson	0.9640	1.1222	0.9789
Static-Multi-U2U	N.A.	0.7980	N.A.
Multi-I2I	0.8640	0.8245	0.8110
I2I-Euclidean	0.8745	0.8429	0.8177
I2I-Pearson	1.1794	0.8644	0.9679
Static-Multi-I2I	N.A.	0.8474	N.A
RatingSGD	0.8409	0.7450	0.8859
ParallelSGD	0.8409	0.7449	0.8852
ALSWR	0.9545	0.9053	1.0354

Conclusion

A multi-criteria RS which exploits users reviews to build a multi-faceted representation of users interests. The novelty of this work lies in the fact that the framework is able to unsupervisedly extract relevant aspects from the review, and it is not bound to a fixed and static set of pre-defined aspects.