Item-Based CF

IIC 3633 - Sistemas Recomendadores

Denis Parra

TOC

En esta clase

1. Resumen Clase anterior
2. Comentario de Jonathan Lee
3. Por qué otra versión de CF?
4. Filtrado Colaborativo Basado Items (Sarwar et al. 2010)
5. Referencias

Resumen última clase

Recommender Systems aim to help a user or a group of users in a system to select items from a crowded item or information space.

- Ranking no personalizado: Varias opciones. Si consideramos que los ítems a rankear tienen valoraciones positivas y negativas, el ranking ideal debería considerar la proporción de positivas y la cantidad de muestras consideradas: una opción es el límite inferior del Intervalo de Confianza del Wilson Score, para un parámetro Bernoulli.

$$\left(\hat{p} + \frac{z^2_{\alpha/2}}{2n} \pm z_{\alpha/2}\sqrt{[\hat{p}(1-\hat{p}) + z^2_{\alpha/2}/4n]/n}\right)/(1 + z^2_{\alpha/2}/n).$$

- Filtrado Colaborativo (Basado en el usuario): Buscamos los K usuarios más parecidos a nuestro "active" o "center" user (K-NN). Luego, hacemos predicción de items que los vecinos han consumido, pero que el "active user" no ha consumido aún.

$$Similaridad(u, v) = w(u, v), v \in K$$

$$\hat{p}_{u,i} = \bar{r}_u + \alpha \sum_{v \in N(u)} w(u, v)(r_{v,i} - \bar{r}_v)$$

Por qué otra versión de Filtrado Colaborativo?

Balance entre Escalabilidad y Exactitud

- **Exactitud**: Mientras más vecinos K consideramos (bajo cierto umbral), mejor debería ser mi clasificación (Lathia et al. 2008)
- **Escalabilidad**: Pero mientras más usuarios n existen en el sistema, mayor es el costo de encontrar los K vecinos más cercanos, ya que K-NN es $O(dnk)$. Considerando un sitio con millones de usuarios, calcular las recomendaciones usando este método $memory - based$ se hace poco sustentable.

Más aún, hay que lidiar con otros problemas

- **Dispersión (Sparsity)**: La baja densidad de los datos hace que el Filtrado Colaborativo basado en el usuario sufra de "Cold-start" (usuarios con pocos ratings o historial de acciones) y también del "new item problem" (items nuevos que nadie los ha consumido)

Opciones

- **Model-based methods**: Redes Bayesianas (ideales en casos en que las preferencias del usuario no cambian tan a menudo), Reducción de dimensionalidad (estado del arte, pero tiene algunos costos de implementación, especialmente en "tunear" los parámetros)
- **Clustering**, aunque tienen como efecto producir recomendaciones "no tan personalizadas" y, disminuir la exactitud de las predicciones en algunos casos (Breese et al. 1998)
- **Graph-based methods**: Horting, Random Walks, Spread of activation. Son menos precisos, pero contribuyen a dar mayor diversidad a las recomendaciones
- **Item-base recommendation**: Revisar user-based (precisión + simpleza) y escalarlo :-)

Alternativa UB-CF con Clustering

- Ungar, L. H., & Foster, D. P. (1998). Clustering methods for collaborative filtering. In AAAI workshop on recommendation systems. ~ EM.
- O'Connor, M., & Herlocker, J. (1999). Clustering items for collaborative filtering. In Proceedings of the ACM SIGIR workshop on recommender systems. ~ Hierarchical.
- Xue, G. R., Lin, C., Yang, Q., Xi, W., Zeng, H. J., Yu, Y., & Chen, Z. (2005). Scalable collaborative filtering using cluster-based smoothing. SIGIR ~ K-means.

Item-Based Collaborative Filtering

USER-BASED

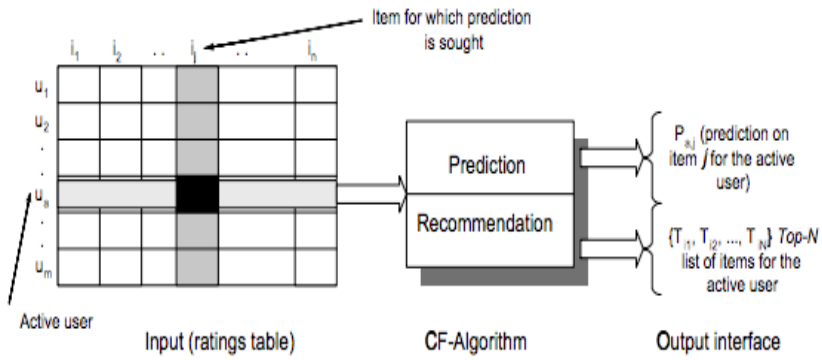


Figure 1: The Collaborative Filtering Process.

ITEM-BASED

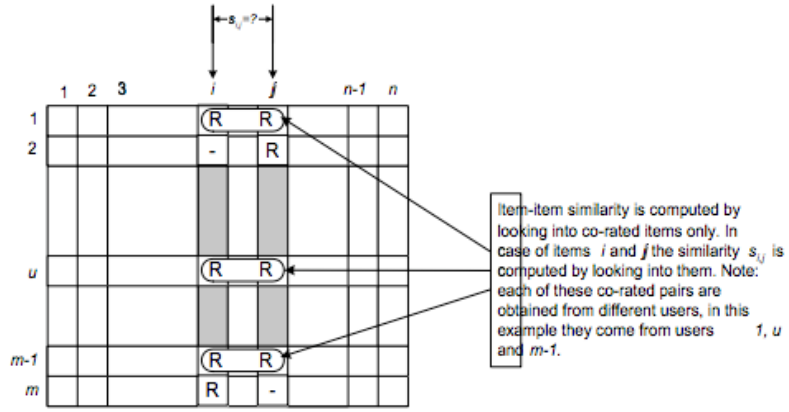


Figure 2: Isolation of the co-rated items and similarity computation

*Imágenes del articulo de Sarwar et al. 2001, **Item-Based Collaborative Filtering Recommendation Algorithms**

Sub-Tareas

1. Calcular Similaridad entre los Ítems
    - Cosine-Based Similarity
    - Correlation-Based Similarity
    - Adjusted Cosine Similarity
    - 1 - Jaccard distance (Das et al. 2007)
2. Cálculo de la Predicción
    - Suma ponderada (weighted Sum)
    - Regresión

Pseudocodigo cálculo Matriz de Similitudes

```
For each item in product catalog, I1
  For each customer C who purchased I1
    For each item I2 purchased by
       customer C
      Record that a customer purchased I1
       and I2
  For each item I2
    Compute the similarity between I1 and I2
```

- En el peor caso es $O(N^2M)$, en la práctica es cercano a $O(NM)$.

1. Calcular Similaridad de los items

- Cosine-Based Similarity

$$sim(i,j) = cos(\vec{i},\vec{j}) = \frac{\vec{i}\cdot\vec{j}}{\|\vec{i}\|_2 \times \|\vec{j}\|_2}$$

- Correlation-Based Similarity

$$sim(i,j) = \frac{\sum_{u\in U}(R_{u,i}-\bar{R}_i)(R_{u,j}-\bar{R}_j)}{\sqrt{\sum_{u\in U}(R_{u,i}-\bar{R}_i)^2}\sqrt{\sum_{u\in U}(R_{u,j}-\bar{R}_j)^2}}$$

- Adjusted Cosine Similarity

$$sim(i,j) = \frac{\sum_{u\in U}(R_{u,i}-\bar{R}_u)(R_{u,j}-\bar{R}_u)}{\sqrt{\sum_{u\in U}(R_{u,i}-\bar{R}_u)^2}\sqrt{\sum_{u\in U}(R_{u,j}-\bar{R}_u)^2}}$$

- 1 - Jaccard distance (Das et al. 2007)

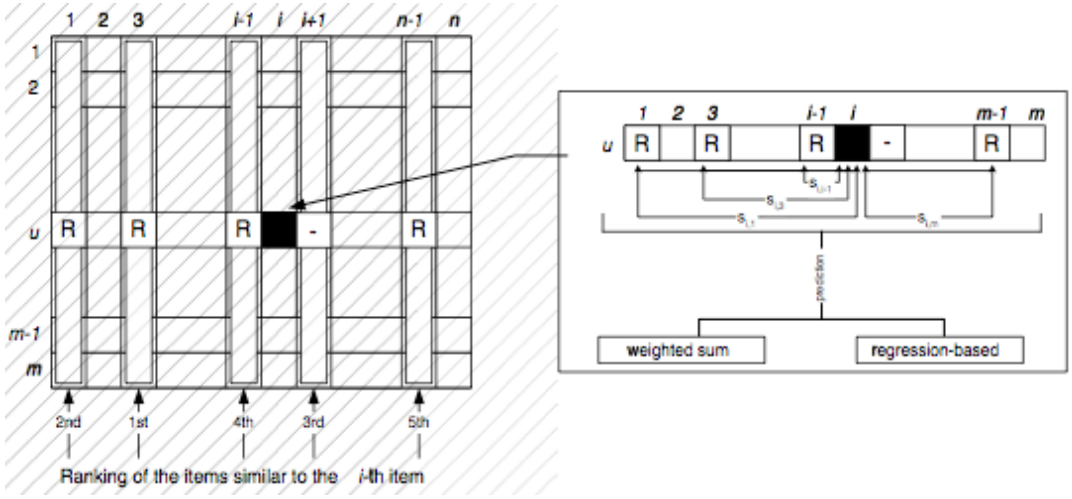
2. Cáculo de la Predicción

- Suma Ponderada (Weighted Sum)

$$\hat{P}_{u,i} = \frac{\sum_{all\ similar\ items,\ N} (sim(i,N) \cdot R_{u,N})}{\sum_{all\ similar\ items,\ N} sim(i,N)}$$

- Regresión

$$\vec{R'}_N = \alpha \cdot \vec{R'}_i + \beta + \epsilon$$



12/19

Análisis

- Si bien este método podría considerarse memory-based, los autores de Sarwar et al. lo consideran model-based, donde el parámetro principal del modelo es K (número de ítems similares a considerar)
- Los autores usan MAE (Mean Absolute Error) para evaluar métodos.
- Resultados importantes para considerar en el análisis:
    - Efecto de la métrica de Similaridad
    - Sensitividad de la proporción Training/Test
    - Tamaño del vecindario (K)
    - Comparación con otros métodos

Resultados I : Similaridad y proporción de training/testing

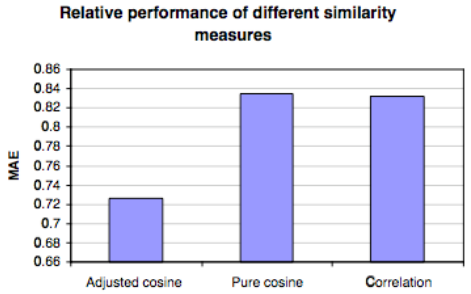


Figure 4: Impact of the similarity computation measure on item-based collaborative filtering algorithm.

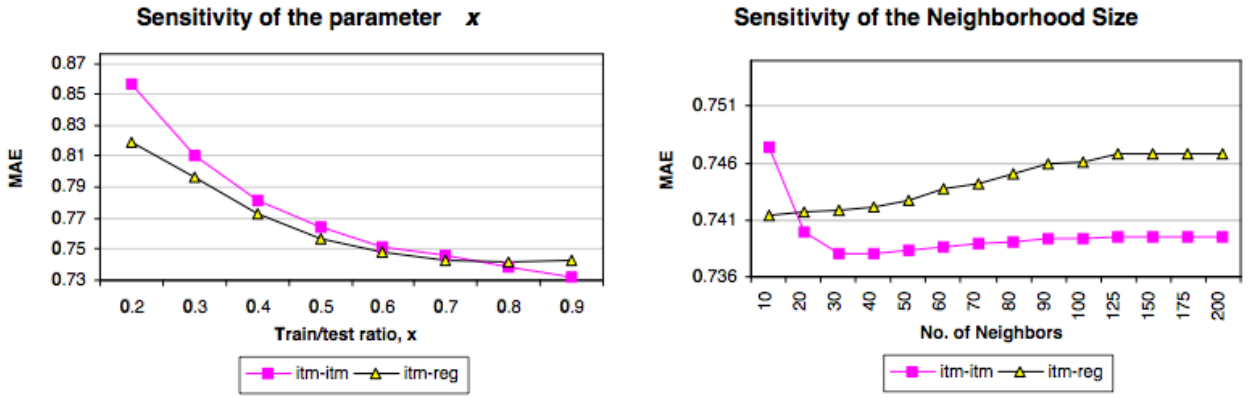


Figure 5: Sensitivity of the parameter x on the neighborhood size

Resultados II : Comparación con Otros métodos

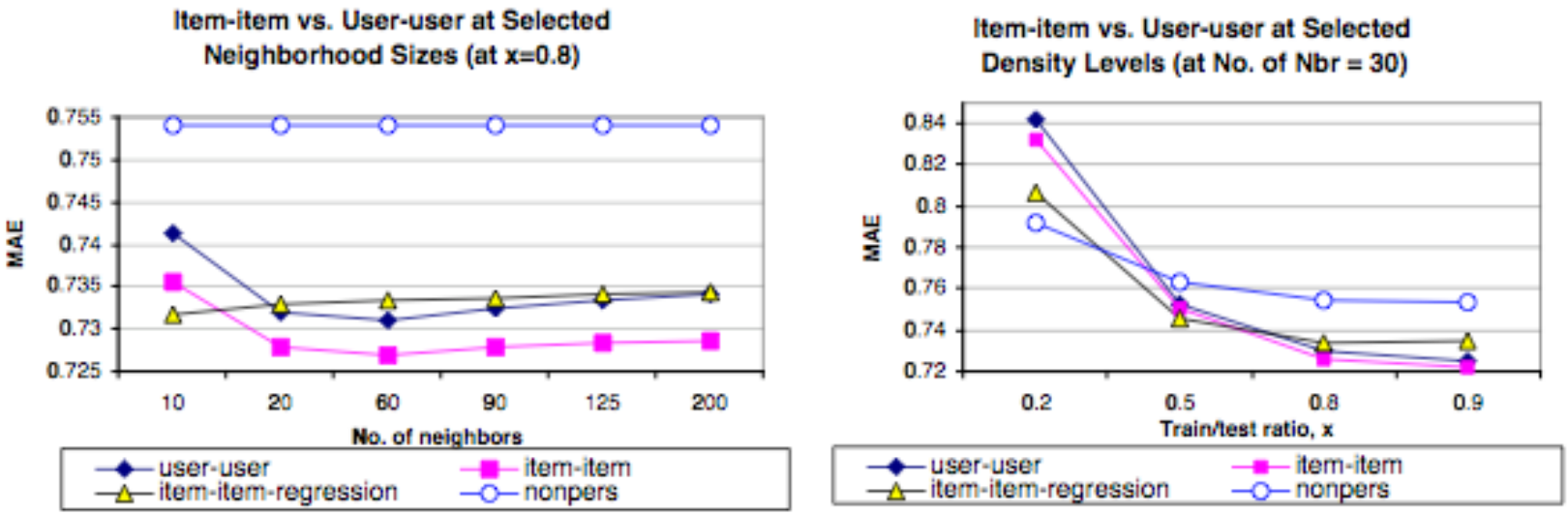


Figure 6: Comparison of prediction quality of *item-item* and *user-user* collaborative filtering algorithms. We compare prediction qualities at $x = 0.2, 0.5, 0.8$ and 0.9.

Resultados IV : Performance

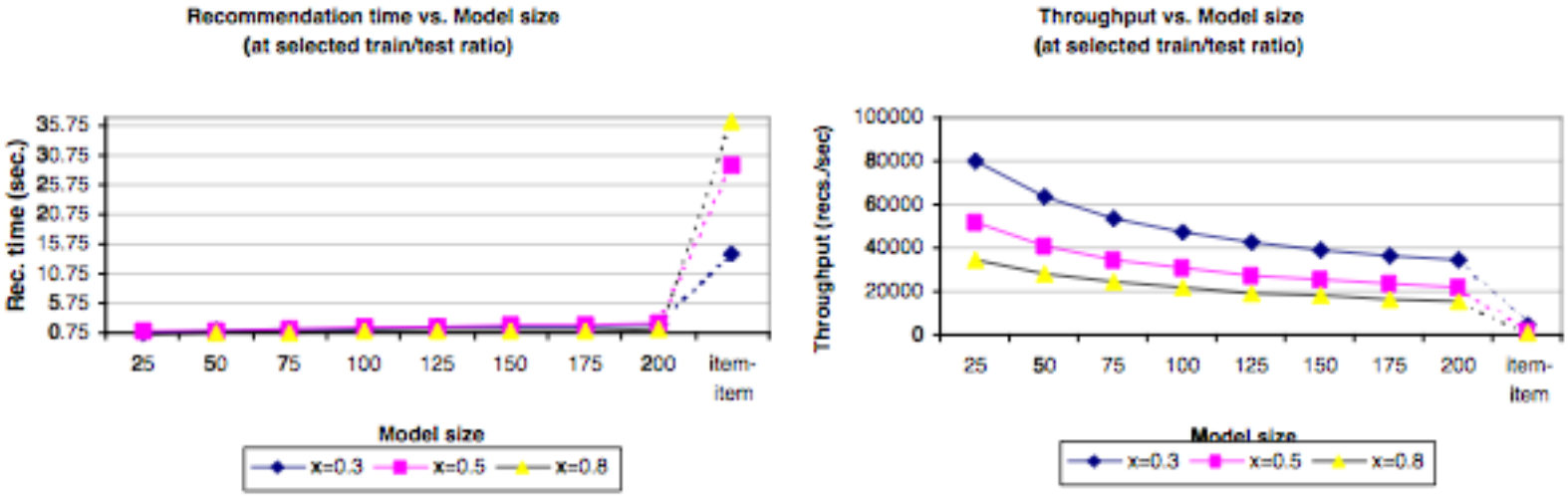


Figure 8: Recommendation time and throughput comparison between model-based scheme and full item-item scheme. The comparisons are shown at three different x values.

Sugerencia

Leer Das et. al "Google News Personalization: Scalable Online Collaborative Filtering", que, para usando patrones de co-visita, incluye:

- Minhash: Probabilistic Clustering Method
- LSH (Locality Sensitive Hashing)
- MapReduce (para implementar MinHash)
- PLSI (Probabilistic Latent Semantic Indexing)

Referencias

- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web (pp. 285-295). ACM.
- Lathia, N., Hailes, S., & Capra, L. (2008, March). The effect of correlation coefficients on communities of recommenders. In Proceedings of the 2008 ACM symposium on Applied computing (pp. 2000-2005). ACM.
- Breese, J. S., Heckerman, D., & Kadie, C. (1998, July). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence (pp. 43-52). Morgan Kaufmann Publishers Inc.
- Das, A. S., Datar, M., Garg, A., & Rajaram, S. (2007, May). Google news personalization: scalable online collaborative filtering. In Proceedings of the 16th international conference on World Wide Web (pp. 271-280). ACM.
- James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, and Dasarathi Sampath. 2010. The YouTube video recommendation system. In Proceedings of the fourth ACM conference on Recommender systems (RecSys '10).

Referencias Adicionales

- Neal Lathia, Stephen Hailes, and Licia Capra. 2008. The effect of correlation coefficients on communities of recommenders. In Proceedings of the 2008 ACM symposium on Applied computing (SAC '08). ACM, New York, NY, USA, 2000-2005
- Gui-Rong Xue, Chenxi Lin, Qiang Yang, WenSi Xi, Hua-Jun Zeng, Yong Yu, and Zheng Chen. 2005. Scalable collaborative filtering using cluster-based smoothing. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '05).
- O'Connor, M., & Herlocker, J. (1999, August). Clustering items for collaborative filtering. In Proceedings of the ACM SIGIR workshop on recommender systems (Vol. 128). UC Berkeley.
- Ungar, L. H., & Foster, D. P. (1998, July). Clustering methods for collaborative filtering. In AAAI workshop on recommendation systems (Vol. 1, pp. 114-129).
- Xavier Amatriain, Josep M. Pujol, and Nuria Oliver . 2009. I Like It... I Like It Not: Evaluating User Ratings Noise in Recommender Systems. In Proceedings of the 17th International Conference on User Modeling, Adaptation, and Personalization: formerly UM and AH (UMAP '09),
- Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS), 22(1), 5-53.