

Inducción de preferencias a partir del contexto de elección del usuario en filtrado colaborativo

Roberto Abalde¹ and Marcelo G. Armentano²

¹Universidad de Buenos Aires

rabalde@dc.uba.ar

²ISISTAN Research Institute (CONICET / UNICEN)

marcelo.armentano@isistan.unicen.edu.ar

Abstract. Los sistemas de recomendación que usan técnicas de filtrado colaborativo basado en memoria predicen la preferencia de un usuario por un ítem usando solamente las preferencias expresadas anteriormente e ignorando el contexto de elección del usuario, esto es, el conjunto de ítems que fueron recomendados al usuario pero que fueron ignorados. En este trabajo se presentan modificaciones a los algoritmos clásicos de filtrado colaborativo basado en memoria para que utilicen el contexto de elección del usuario para predecir las preferencias. Una serie de experimentos desarrollados verifican que nuestra propuesta presenta un mejor desempeño que los algoritmos de filtrado colaborativo clásicos.

Keywords: Sistemas de recomendación; Filtrado colaborativo basado en memoria; Información de contexto; Contexto de elección del usuario

1 Introducción

El crecimiento de internet ha permitido presentar a los usuarios una cantidad cada vez mayor de alternativas entre las que elegir y este problema ha impulsado la investigación y desarrollo de los sistemas de recomendación [8], un conjunto de técnicas y herramientas de software con el objetivo de asistir a un usuario durante la toma de decisiones, recomendándole ítems que le podrían ser interesantes, por ejemplo qué productos comprar, qué música escuchar o qué noticias leer.

Las interacciones entre los usuarios y los sistemas de recomendación se desarrollan en dos partes: el sistema le recomienda al usuario una lista de ítems y luego el usuario expresa sus preferencias sobre lo recomendado seleccionando un conjunto (posiblemente vacío) de ítems de la lista e ignorando el resto.

Para inducir modelos de preferencia, los algoritmos actuales de filtrado colaborativo utilizan sólo la información histórica sobre los ítems que fueron seleccionados, desaprovechando la información relativa a los ítems ignorados.

En el presente artículo, se presentan modificaciones a las técnicas clásicas de filtrado colaborativo basado en memoria para que utilicen, en el proceso de inducción de preferencias, el contexto de elección del usuario, es decir el conjunto de ítems recomendados al usuario pero no seleccionados. Además se

presentan resultados de una serie de experimentos que permitieron verificar que las modificaciones realizadas mejoran el desempeño de los algoritmos de filtrado colaborativo.

En la siguiente sección se define el problema de los sistemas de recomendación y se presentan las técnicas de filtrado colaborativo que serán modificadas. En la sección 3 se discuten las adaptaciones realizadas a conjuntos de datos existentes. En la sección 4 se presentan los experimentos realizados para comparar el desempeño entre las técnicas de recomendación que usan el contexto de elección del usuario y las técnicas clásicas y finalmente se muestran los resultados obtenidos en la sección 5.

2 Sistemas de recomendación

2.1 Definición del problema

Consideremos las interacciones usuario-sistema en un sistema de recomendación: tenemos un conjunto de usuarios $U = u_1, \dots, u_n$ y un conjunto de ítems $I = i_1, \dots, i_n$; cuando un usuario u ingresa al sistema, éste calcula un conjunto de ítems recomendados $O \subseteq I$ para dicho usuario y luego el usuario u selecciona un subconjunto (posiblemente vacío) de ítems $D \subseteq O$ y realiza alguna acción sobre ellos (por ejemplo compra alguno de los productos recomendados y seleccionados).

Para poder calcular las recomendaciones, el sistema de recomendación mantiene un registro histórico de las interacciones de la forma (u, O_t, D_t) donde t es el índice de una interacción particular. El objetivo de un sistema de recomendación es poder calcular conjuntos de recomendaciones O que maximicen la satisfacción del usuario con respecto a los ítems recomendados.

2.2 Filtrado Colaborativo

En el filtrado colaborativo se cuenta con una serie de observaciones de la forma (u, i, y_i) donde y_i es una respuesta observada, por ejemplo, una calificación dada por el usuario u al ítem i o la indicación de que el usuario u realizó una acción sobre el ítem i . Mientras que estas observaciones construyen una gran matriz que puede tener millones de ítems y usuarios, usualmente se tienen muy pocas observaciones (menos del 1% en los conjuntos de datos públicos).

El filtrado colaborativo utiliza la similitud entre usuarios o entre ítems para predecir la utilidad que le va a dar un usuario a un ítem no calificado anteriormente. Estas técnicas se basan en el supuesto de que dos usuarios con preferencias similares van a calificar el mismo ítem en forma similar o que dos ítems calificados por los usuarios en forma similar tendrán calificaciones similares en el futuro [8].

Cuando el filtrado colaborativo se basa en un modelo de vecinos, para poder realizar las predicciones de las utilidades se calcula un conjunto de vecinos usando una medida de similitud entre usuarios o ítems, seleccionando los k más cercanos

o con los que se encuentren por debajo de un umbral de similitud dado. Cuando sólo se toma en cuenta los conjuntos de ítems calificados y no sus calificaciones o cuando las calificaciones son unarias, como es el caso en los experimentos realizados, se suele usar como medida el coeficiente de similitud de Jaccard [2].

En el caso de calificaciones unarias, la predicción se calcula como la media de las similitudes de los vecinos que calificaron el ítem (ecuación 1)

$$p_{u,i} = \frac{\sum_{v \in K} S_{v,i} w_{u,v}}{\sum_{v \in K} w_{u,v}} \quad (1)$$

donde $p_{u,i}$ es la predicción de la utilidad del ítem i para el usuario u , $w_{u,v}$ es la similitud entre los usuarios u y v , K es el conjunto de usuarios similares a u (seleccionados por cantidad o por un umbral de similitud) y $S_{v,i}$ es un indicador cuyo valor es 1 cuando el usuario u calificó el ítem i , sino es 0.

Cuando la cantidad de usuarios es mucho mayor a la de ítems se presentan problemas de escalabilidad y para evitarlos se desarrolló el filtrado colaborativo basado en la similitud entre ítems [4] que busca ítems similares a los calificados por el usuario en vez de buscar usuarios similares. El supuesto detrás de este enfoque es que los usuarios van a preferir ítems similares a los que ya les gustaron.

En el filtrado colaborativo basado en ítems para el caso de calificaciones unarias, la predicción se calcula como indica la ecuación 2:

$$p_{u,i} = \frac{\sum_{j \in K} S_{u,j} w_{i,j}}{\sum_{j \in K} w_{i,j}} \quad (2)$$

donde $p_{u,i}$ es la predicción de la utilidad del ítem i para el usuario u , $w_{i,j}$ es la similitud entre los ítems i y j , K es el conjunto de ítems calificados por u que son similares a i (seleccionados por cantidad o por un umbral de similitud) y $S_{u,j}$ es un indicador cuyo valor es 1 cuando el usuario u calificó el ítem j , sino es 0.

3 Inducción de preferencias a partir del contexto de elección del usuario

La investigación sobre sistemas de recomendación contextuales [1, 6] usualmente consideran el contexto de los usuarios o de los ítems a recomendar, pero no el *contexto de elección del usuario*, esto es, la información del contexto en que el usuario realiza la elección efectiva de los ítems recomendados, puntualmente qué ítems fueron recomendados y cuáles fueron ignorados.

3.1 El contexto de elección del usuario

Cuando el usuario interactúa con un sistema de recomendación no expresa una preferencia absoluta sobre los ítems, sino que expresa una preferencia de los ítems seleccionados sobre todo el resto de los recomendados [7] ya que no los conoce

todos los ítems disponibles. En otras palabras, está expresando una preferencia relativa: prefiere los ítems seleccionados por sobre los que fueron ignorados.

El conjunto de ítems ignorados, que denominaremos *contexto de elección del usuario*, es un tipo de información contextual no utilizada hasta hace poco tiempo en los sistemas de recomendación. En el ámbito de recuperación de información se ha utilizado una idea similar [3]: Cuando un buscador presenta una lista de enlaces a un usuario luego de realizar una consulta y el usuario selecciona uno de ellos, se induce que el usuario está diciendo implícitamente que los enlaces anteriores al seleccionado no son importantes para él.

El primer uso conocido del contexto de elección del usuario en sistemas de recomendación [9] fue presentado como parte de un modelo de factores latentes (filtrado colaborativo basado en modelos) que deriva en un modelo *logit* multinomial.

El presente trabajo hace uso del contexto de elección del usuario para mejorar los sistemas de recomendación pero, a diferencia de los trabajos anteriores, se enfoca en adoptar las técnicas de filtrado colaborativo basado en memoria.

3.2 Adaptación de los métodos de filtrado colaborativo

En el filtrado colaborativo basado en usuarios clásico, la creación de la lista de ítems a recomendar comienza con la creación de un conjunto de ítems candidatos, formado por todos aquellos ítems existentes que el usuario no seleccionó en ninguna de sus interacciones con el sistema. Luego, para cada ítem del conjunto de candidatos, se calcula la predicción de su utilidad como la suma de las distancias de los usuarios similares que seleccionaron dicho ítem. Por último, se crea la lista de ítems a recomendar ordenando los ítems del conjunto de candidatos por la utilidad calculada.

Veamos este proceso en un ejemplo. Supongamos que todos los ítems en el sistema de recomendación son $I = \{i_1, i_2, i_3, i_4, i_5, i_6, i_7, i_8, i_9, i_{10}\}$ y que el usuario u seleccionó en interacciones anteriores los ítems $S = \{i_3, i_5, i_7\}$. Así, los ítems candidatos para u son $C = I - S = \{i_1, i_2, i_4, i_6, i_8, i_9, i_{10}\}$. Ahora, supongamos que la predicción de las utilidades los ítems candidatos es $P = (0.2, 0.03, 0.37, 0.16, 0.1, 0.14, 0.0)$ donde P_j es la utilidad del ítem i_j . La predicción de utilidad de un ítem puede ser 0.0, como el caso de i_{10} , porque los vecinos más cercanos del usuario actual no seleccionaron nunca ese ítem. Luego, en base a P ordenamos C y obtenemos la lista de ítems a recomendar $R = (i_4, i_1, i_6, i_9, i_8, i_2, i_{10})$.

Ahora, supongamos que los ítems ignorados por el usuario son $T = \{i_1, i_9\}$. Estos ítems deberían estar cerca de los últimos puestos de la lista de recomendaciones ya que si fueron ignorados por el usuario, podemos suponer que tienen menor o igual utilidad que los ítems nunca vistos y esto nos podría permitir hacer un mejor ordenamiento de C .

Entonces, para calcular la predicción de utilidad de un ítem i para un usuario u , tomando en cuenta los ítems ignorados, primero sumaremos las similitudes para los ítems seleccionados por los vecinos cercanos a u . A esta suma la ponderaremos con un parámetro β y, si i fue ignorado por u , le restaremos $(1 - \beta)$.

El parámetro β es un valor entre 0 y 1 que sirve para balancear el peso de los ítems seleccionados sobre los ignorados.

Volviendo al ejemplo anterior, si tomamos $\beta = 0.9$ tenemos que $P = \beta P = (0.18, 0.027, 0.333, 0.144, 0.09, 0.126, 0.0)$. Luego, si le restamos $(1 - \beta)$ a los ítems ignorados, tenemos $P = (0.08, 0.027, 0.333, 0.144, 0.09, 0.026, 0.0)$ con lo que el ordenamiento de C sería $R = (i_4, i_6, i_8, i_2, i_{10}, i_1, i_9)$ teniendo los ítems ignorados al final de la lista.

Esta modificación traducida al cálculo de la predicción de utilidad de un ítem para un usuario (ecuación 1) puede calcularse como indica la ecuación 3

$$p_{u,i} = \beta \frac{\sum_{v \in K} S_{v,j} w_{u,v}}{\sum_{v \in K} w_{u,v}} - (1 - \beta) T_{u,i} \quad (3)$$

donde $p_{u,i}$ es la predicción de la utilidad del ítem i para el usuario u , $w_{u,v}$ es la similitud entre los usuarios u y v , K es el conjunto de usuarios similares a u , (seleccionados por cantidad o por un umbral de similitud), $S_{v,i}$ es un indicador cuyo valor es 1 cuando el usuario v calificó el ítem i , sino es 0, $T_{u,i}$ es un indicador cuyo valor es 1 cuando el usuario u ignoró el ítem i , sino es 0 y β es un valor entre 0 y 1 que pondera los ítems seleccionados contra los ignorados.

De forma análoga, modificamos la forma en que el filtrado colaborativo basado en ítems calcula la predicción de la utilidad (ecuación 2) de la siguiente manera:

$$p_{u,i} = \beta \frac{\sum_{j \in K} S_{u,j} w_{i,j}}{\sum_{j \in K} w_{i,j}} - (1 - \beta) T_{u,j} \quad (4)$$

donde $p_{u,i}$ es la predicción de la utilidad del ítem i para el usuario u , $w_{i,j}$ es la similitud entre los usuarios i y j , $S_{u,j}$ es un indicador cuyo valor es 1 cuando el usuario v calificó el ítem i , sino es 0, $T_{u,j}$ es un indicador cuyo valor es 1 cuando el usuario u ignoró el ítem i , sino es 0, K es el conjunto de los k usuarios que calificaron i y que son similares a u (seleccionados por cantidad o por un umbral de similitud) y β es el peso que se le da a los ítems seleccionados y a los ignorados.

Cabe destacar que cuando $\beta = 1.0$ no se realiza la penalización de los intereses seleccionados porque se anula el término que está restando, por lo que las ecuaciones modificadas son iguales a las originales. A estos casos los llamaremos "base" mientras que al resto lo llamaremos "contexto".

4 Experimentos

Para evaluar las modificaciones propuestas en la sección anterior en un entorno real, se desarrolló y puso en línea un sistema de recomendación de películas¹ que genera recomendaciones a partir de 8048 películas argentinas, británicas, españolas y estadounidenses, estrenadas entre 1970 y 2012.

¹ Disponible en <http://pelisquemegustan.inductia.com>

El sistema de recomendación implementa dos algoritmos de filtrado colaborativo por similitud de usuarios: el modelo clásico y el que refleja las modificaciones propuestas en este trabajo. Cuando un usuario se registra en el sistema, se le asigna automáticamente y al azar uno de los dos algoritmos implementados. De esta forma se dispone de un grupo de prueba y un grupo de control.

La escala de las calificaciones utilizada por el sistema implementado es unaria debido al poco esfuerzo que dicha escala requiere del usuario para expresar sus preferencias y por ello la distancia Jaccard para calcular los vecinos más cercanos.

En cada interacción el sistema le recomienda 8 películas al usuario. Se eligió esta cantidad en base a la teoría de que la memoria de trabajo de una persona puede albergar 7 ± 2 objetos al mismo tiempo [5]. Entonces, podemos suponer que el usuario observa todos los ítems presentados, selecciona aquellos que le interesan e ignora el resto. Así, el contexto de elección del usuario va a tener entre 1 y 8 películas en cada interacción. Después de cada interacción con un usuario, el sistema almacena la identificación del usuario, la lista de ítems recomendados, la lista de los ítems seleccionados y la fecha y la hora en que se realizó la interacción.

Usando el conjunto de datos generado por el sistema de recomendación desarrollado² y un conjunto de datos público *movielens-100k*³, verificamos que mejora el desempeño del sistema de recomendación.

Debido a que el conjunto de datos *movielens-100k* no tiene el contexto de elección del usuario, el mismo fue simulado utilizando la política descrita en [9]: se toman las calificaciones con valor '5' como los ítems seleccionados y el resto de las calificaciones como ítems ignorados. Los ítems que no tienen una valoración en el conjunto de datos original tampoco la tienen a tener en el conjunto modificado. Con el contexto de elección simulado, se tienen 21.201 ítems seleccionados (1.34% del total) y 78.799 ítems ignorados (3.7 ítems ignorados por cada seleccionado).

El conjunto de datos recolectado por la aplicación implementada, consiste de 115 usuarios que utilizaron el algoritmo clásico, con 2.975 películas seleccionadas (0.32% del total disponible) en 9.107 interacciones con el sistema. Este conjunto de datos no tiene películas ignoradas ya que no era necesario para el algoritmo registrarlas. El conjunto del algoritmo modificado consiste de 114 usuarios que realizaron 7.057 interacciones con el sistema, seleccionando 5.339 películas (0.58 % del total) e ignorando 50.274. En total se tienen 8.314 películas seleccionadas y 120.941 ignoradas, esto es, 14.4 películas ignoradas por cada seleccionada.

5 Resultados

Las pruebas se realizaron simulando una situación "en línea", esto es, se reprodujeron cada una de las interacciones que realizó cada usuario con el sistema, siguiendo la línea de tiempo presente en el conjunto de datos.

Se comparó el algoritmo de filtrado colaborativo basado en usuarios clásico ($\beta = 1.0$) con el modificado con $\beta = 0.1$ y $K = 90$, siendo estos valores los que mostraron mejor desempeño en pruebas preliminares realizadas fuera de línea.

² Disponible en <http://pelisquemegustan.inductia.com/dataset/>

³ <http://www.grouplens.org/datasets/movielens/>

Los resultados usando el conjunto de datos *movielens-100k* se muestran en la Tabla 1. Las métricas utilizadas son las clásicas utilizadas en sistemas de recomendación: precisión y ganancia acumulada descontada normalizada (NDCG), ambas tomadas para listas de recomendaciones 1, 5 y 10 elementos.

Algoritmo	NDCG@1	NDCG@5	NDCG@10	P@5	P@10
Base	0,0218	0,0172	0,0149	0,0160	0,0132
Contexto	0,0338	0,0242	0,0202	0,0218	0,0172

Valores estadísticamente significativos ($p < 0.001$)

Table 1. Comparación de las medias de las métricas resultantes de aplicar el algoritmo base y el algoritmo propuesto para el conjunto de datos *movielens-100k*

Puede observarse que el algoritmo modificado supera los valores del algoritmo clásico para todas las métricas. Las diferencias en porcentaje resultantes del algoritmo propuesto con respecto al algoritmo base fueron de 55,34%, 40,19% y 35,61% para NDCG y 36,13% y 30,31% para AP.

Para la segunda prueba en línea, se utilizó el conjunto de datos obtenido del sistema de recomendación desarrollado. En este caso se calculan las métricas NDCG@8 y P@8 ya que el sistema recomienda al usuario 8 ítems en cada interacción. La Tabla 2 muestra los resultados obtenidos.

Algoritmo	NDCG@8	P@8
Base	0,0386	0,0411
Contexto	0,0976	0,0957

Valores estadísticamente significativos ($p < 0.001$)

Table 2. Comparación de las medias de las métricas resultantes de aplicar el algoritmo base y el algoritmo propuesto para el conjunto de datos creado por el sistema desarrollado

Podemos observar que el algoritmo modificado ($\beta = 0, 1$) domina al modelo clásico ($\beta = 1, 0$) notablemente para ambas métricas, un 152,8% para NDCG y un 132,8% para Precisión.

6 Conclusiones

En este trabajo presentamos modificaciones a los algoritmos clásicos de filtrado colaborativo basado en memoria para que utilicen el contexto de elección del usuario en sus predicciones de las preferencias de los usuarios. Las modificaciones propuestas son simples y de fácil implementación en los sistemas de recomendación en producción.

Además, hemos verificado que las modificaciones propuestas mejoran significativamente el desempeño de los algoritmos clásicos de filtrado colaborativo basados en memoria. Las pruebas realizadas mostraron que los algoritmos modificados tienen un desempeño mucho mejor que los algoritmos clásicos.

El código fuente de nuestra implementación de los algoritmos modificados se encuentra disponible online y puede ser utilizado como plataforma para nuevos trabajos de investigación.

7 Agradecimientos

Este artículo se deriva de un trabajo de tesis de la Maestría en la Explotación de Datos y Descubrimiento del Conocimiento de la Universidad de Buenos Aires.

References

1. G. Adomavicius, A. Tuzhilin, S. Berkovsky, E. W. De Luca, and A. Said. Context-aware recommender systems. In *Research Workshop and Movie Recommendation Challenge*, New York, NY, USA, 2010. ACM.
2. John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, UAI'98, pages 43–52, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.
3. Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 133–142, New York, NY, USA, 2002. ACM.
4. Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, January 2003.
5. G. A. Miller. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97, 1956.
6. Umberto Panniello, Alexander Tuzhilin, and Michele Gorgoglione. Comparing context-aware recommender systems in terms of accuracy and diversity. *User Modeling and User-Adapted Interaction*, 24(1-2):35–65, February 2014.
7. Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 452–461, Arlington, Virginia, United States, 2009. AUAI Press.
8. F. Ricci, L. Rokach, B. Shapira, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. *Recommender Systems Handbook*. Springer, 2011.
9. Shuang-Hong Yang, Bo Long, Alexander J. Smola, Hongyuan Zha, and Zhaohui Zheng. Collaborative competitive filtering: Learning recommender using context of user choice. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11, pages 295–304, New York, NY, USA, 2011. ACM.