Elección de Sistema de Recomendación para GuideMe

Andrés Felipe Medina Tascón, Oscar Alexander Ruiz Palacio
201667602, 201667600
Escuela de Ingeniería de Sistemas y Computación
Universidad del Valle
Tuluá, Colombia
{andres.medina, ruiz.oscar}@correounivalle.edu.co

***Resumen*— GuideMe es una aplicación móvil que tiene como objetivo principal caracterizar los lugares de interés del municipio de Ginebra y que todos estos lugares sean visibles para locales y turistas, para cumplir con este fin se implementará un sistema de recomendación. En este documento se examinarán diferentes técnicas de recomendación que recomienda la literatura para sistemas de recomendación en aplicaciones nuevas, tomando en cuenta técnicas basadas en contenido y de filtro colaborativo, y se elegirá una de estas técnicas para la aplicación teniendo en cuenta cobertura y complejidad computacional como criterios para elegir la técnica que se desarrollará para la aplicación**

***Palabras Clave*—GuideMe, Sistema de Recomendación, Filtro Colaborativo, Basado en Contenido, Complejidad.**

I. INTRODUCCION.

Se puede definir los sistemas de recomendación como "el conjunto de herramientas de software y técnicas que ofrecen sugerencias útiles al usuario."

Los sistemas de recomendación se crean para que los usuarios tengan sugerencias personalizadas de acuerdo a sus gustos y preferencias, esto se realiza para que el usuario tenga una visión amplia de que puede explorar alternativas que le puedan gustar, los sitios webs para compras como Amazon utilizan estas técnicas para que todos los productos de calidad que ofrecen puedan ser visualizados por los usuarios de acuerdo a sus gustos y le pueda ser de utilidad a los usuarios. "Los sistemas de recomendación intentan predecir cuales son los productos o servicios más adecuados para el usuario de acuerdo a sus preferencias y restricciones".

Existen una gran variedad de técnicas a la hora de realizar un sistema de recomendación el éxito o no de este dependen del diseño. Por lo cual es esencial tener en cuenta los usuarios, los ítems y las transacciones entre estos, por ejemplo, las calificaciones[1].

A. Filtro Colaborativo

Los sistemas de recomendación con filtro colaborativo es una de las técnicas más utilizadas a la hora de realizar sugerencias en aplicaciones con bastantes usuarios. Según Michael D. Ekstrand, John T. Riedl y Joseph A. Konstan "Es un algoritmo de recomendación popular que basa sus predicciones y recomendaciones en calificaciones o comportamientos de otros usuarios en el sistema"[2]. Para este proceso es necesario tener en cuenta la comunidad que forma parte del sistema tomando la retroalimentación que estos usuarios le dan con sus opiniones y calificaciones encontrando así características en común que puedan ser de utilidad para los usuarios[3].

Para el filtro colaborativo se tienen dos tipos esencialmente que son los que tienen información de los usuarios de forma explícita ya sea mediante las calificaciones que les dé a los artículos de su interés o ya que ingresa sus gustos directamente. Por otro lado están los usuarios que no le otorgan tanta información al sistema de manera tan directa, es decir, se obtiene de manera implícita, de estos usuarios la información se toma por número de clics, historial de navegación, historial de compras en caso de e-commerce, son algunas de las maneras que se extrae información de estos a la hora de realizar recomendaciones[1].

B. Filtro Basado en Contenido

El filtro basado en contenido toma los ítems con su descripción y según el perfil de los usuarios determina cuales pueden ser de su interés, este proceso se realiza sobre todo en las aplicaciones en las cuales se hacen bastantes compras. "Las recomendaciones de estos sistemas depende de los artículos con los que el usuario ha interactuado. En particular, varios artículos candidatos se comparan con los artículos previamente calificados por el usuario y se recomiendan los artículos que mejor combinen." [1], [4], [5]

C. Técnicas Híbridas

Estas técnicas de sistemas de recomendación se basan en la combinación de técnicas mencionadas anteriormente. Un sistema híbrido que combina las técnicas A y B, e intenta utilizar las ventajas de A para corregir las desventajas de B. Por ejemplo, los métodos de filtro colaborativo sufren

problemas con nuevos ítems, es decir, no pueden recomendar artículos que no tienen calificaciones. Esto no tiene límites para el filtro basado en el contenido ya que la predicción para nuevos los artículos se basan en su descripción (características) que normalmente están disponibles.[1]

II. METODOLOGÍA

Para elegir la técnica de recomendación para usar en la aplicación móvil se consulta la literatura para examinar cuál de las técnicas es apropiada para el sistema de recomendación que se plantea realizar, y se realiza una prueba de concepto de estás técnicas con datos generados aleatoriamente.

Se busca en las técnicas teniendo en cuenta que la aplicación en un inicio no tendrá usuarios, pero si tendrá ítems (lugares).

A. Filtro Colaborativo Usuario-Usuario[2]

También conocido como filtro colaborativo k-NN, es una interpretación algorítmica sencilla de la premisa del filtro colaborativo: encontrar otros usuarios cuyos gustos sean similar al del usuario actual usando las calificaciones de ellos en otros ítems para predecir que le gustaría al usuario actual.

Además de la matriz de calificaciones R, un sistema de filtro colaborativo usuario-usuario requiere una función de similitud $s: U \times U \rightarrow \mathbb{R}$ computando la similitud entre dos usuarios y un método para usar las similitudes y las calificaciones para generar predicciones.

Para generar predicciones o recomendaciones para un usuario u, esta técnica usa s para computar una vecindad $N \subseteq U$ vecinos de u. Una vez N ha sido computado, el sistema combina las calificaciones de los usuarios en N para generar la predicción de las preferencias de un usuario u por un ítem i. Esto se hace típicamente computando el peso promedio de las calificaciones que los vecinos le dan a i, usando la similitud como los pesos:

$$p_{u,i} = \bar{r}_u + \frac{\sum_{u' \in N} s(u,u')(r_{u',i} - \bar{r}_{u'})}{\sum_{u' \in N} |s(u,u')|}$$

Para calcular la similitud se puede utilizar la similitud de coseno el cual tiene un enfoque vectorial, en donde los usuarios son representados como vectores $|I|$-dimensionales y la similitud es medida por la distancia coseno entre los vectores de calificaciones. Este puede ser computado eficientemente tomando su producto punto dividiéndolo por de sus normas:

$$s(u,u') = \frac{r_u \cdot r_{u'}}{||r_u||_2 \, ||r_{u'}||_2}$$

$$s(u,u') = \frac{\sum_i r_{u,i} \, r_{u',i}}{\sqrt{\sum_i r^2{}_{u,i}} \, \sqrt{\sum_i r^2{}_{u',i}}}$$

B. Filtro Colaborativo Item-Item[2]

Esta técnica en vez de usar similitudes entre las calificaciones de los usuarios para predecir preferencias, ítem-ítem usa las similitudes entre los patrones de calificación de los ítems. Si dos ítems tienden a tener los mismos usuarios que les gusta y los que no les gusta, entonces son similares y los usuarios se espera tener preferencias similares para ítems similares.

El filtro colaborativo ítem–ítem genera predicciones utilizando las calificaciones del usuario para otros ítems combinados con las similitudes de esos ítems con el ítem objetivo, en lugar de las calificaciones y similitudes de otros usuarios como en usuario - usuario. Similar al usuario-usuario, el sistema de recomendación necesita una función de similitud, esta vez $s: I \times I \rightarrow \mathbb{R}$, y un método para generar predicciones a partir de valoraciones y similitudes.

En los dominios de calificación de valor real, las puntuaciones de similitud se pueden utilizar para generar predicciones utilizando un promedio ponderado, similar al procedimiento utilizado en usuario-usuario CF. Las recomendaciones son luego generadas tomando los ítems candidatos con las predicciones más altas.

Después de recoger un conjunto S de ítems similares a i, $p_{u,i}$ se puede predecir:

$$p_{u,i} = \frac{\sum_{j \in S} s(i,j) r_{u,j}}{\sum_{j \in S} |s(i,j)|}$$

La similitud de coseno entre los vectores de valoraciones de ítems es la métrica de similitud más popular.

$$s(i,j) = \frac{r_i \cdot r_j}{||r_i||_2 \, ||r_j||_2}$$

C. Modelo de Espacio Vectorial basado en palabras clave[1]

Vector Space Model (VSM) es una representación de documentos de texto. En ese modelo cada documento es representado por un vector en un espacio n-dimensional, donde cada dimensión corresponde a un término del vocabulario general de una colección dada de un documento.

La representación del documento en VSM plantea dos cuestiones: ponderar los términos y medir la similitud del vector. El esquema de ponderación de términos más utilizado, es la ponderación TF-IDF (Term Frequency-Inverse Document Frequency), está basada en las observaciones empíricas respecto a los textos:

- Los términos raros no son menos relevantes que los términos frecuentes (suposición de IDF)
- Las múltiples apariciones de un término en un documento no son menos relevantes que las ocurrencias únicas (suposición de TF)

- Los documentos largos no se prefieren a los documentos cortos (suposición de normalización).

$$\mathrm{TF-IDF}(t_k, d_j) = \mathrm{TF}(\mathrm{t}_k, \mathrm{d}_j) \cdot \log \frac{\mathrm{N}}{n_k}$$

Donde N describe el número de documentos en el corpus, y n_k describe el número de documentos en la colección en la cual el término t_k ocurre al menos una vez.

$$\mathrm{TF}(\mathrm{t}_k, \mathrm{d}_j) = \frac{f_{k,j}}{max_z f_{z,j}}$$

Donde el máximo es computado sobre las frecuencias $f_{z,j}$ de todos los términos t_z que ocurren en el documento d_j. Para que las ponderaciones estén en el intervalo $[0, 1]$ y para que los documentos sean representados por vectores de igual tamaño, las ponderaciones usualmente se normalizan por la normalización de coseno:

$$w_{k,j} = \frac{\mathrm{TF-IDF}(t_k, d_j)}{\sqrt{\sum_{s=1}^{|T|} \mathrm{TF-IDF}(t_s, d_s)^2}}$$

Una medida de similitud es requerida para determinar que tanto se parecen dos documentos. Muchas medidas de similitud han sido derivadas para describir la proximidad de dos vectores, entre esas medidas, la similitud de coseno es de las más usadas:

$$sim(d_i, d_j) = \frac{\sum_k w_{ki} \cdot w_{kj}}{\sqrt{\sum_k {w_{ki}}^2} \cdot \sqrt{\sum_k {w_{kj}}^2}}$$

En los sistemas de recomendación basados en contenido basado en VSM, el perfil del usuario y los ítems ambos son representados como vectores de términos ponderados. Las predicciones de un interés del usuario en un ítem particular pueden ser derivadas computando la similitud de coseno.

D. Slope One[6]

El algoritmo Slope One está basado en un modelo lineal unario $f(x) = x + b$ para la predicción. El principio de Slope One es calificar ítems sin calificar basado en la desviación promedio de las valoraciones de los ítems. Este consta de dos partes, el cálculo de la formula de desviación y la formula de predicción. Dados dos ítems i y j ($i \neq j$), $R_{u,i}$ es la calificación de un usuario u a un ítem i. $S_{u,i}$ es una colección de usuarios que calificaron ambos ítems i y j. $|S_{i,j}|$ es el número de usuarios en el conjunto $S_{i,j}$, y la desviación entre el ítem i y el ítem j se calcula así:

$$dev_{i,j} = \frac{\sum_{u \in S_{i,j}} (R_{u,i} - R_{u,j})}{|S_{i,j}|}$$

Obteniendo la desviación, se puede predecir la calificación de un usuario u para un ítem i:

$$pre_{u,i} = \frac{\sum_{j \in S(u)-\{i\}} (R_{u,j} - dev_{i,j})}{|S(u) - \{i\}|}$$

Donde, $S(u)$ representa el conjunto de ítems calificados por el usuario u, y $S(u) - \{i\}$ representa el conjunto de ítems en los cuales hay al menos uno calificado por el usuario

III. Resultados

Para elegir uno de los algoritmos que se encontraron en la literatura se toman en cuenta los siguientes aspectos:

- Escalabilidad
- Cobertura

Se toman en cuenta estos aspectos ya que son los que se pueden medir inmediatamente sin depender de los datos que se tengan para el algoritmo, ya que se mira solo algunos puntos de estos aspectos como lo es la complejidad computacional de cada algoritmo, y si estos presentan el problema del coldstart. Ya que es un prototipo para una aplicación nueva, por lo tanto no tendrá usuarios inicialmente, pero si contará con lugares, como máximo 40.

Algoritmo	Complejidad Computacional
Usuario-Usuario	$O(D * U)$
Item-Item	$O(D * I)$
VSM con TF-IDF	$O(T * D)$
Slope One	$O(U * I^2)$

Fuente: Elaboración Propia.

La complejidad computacional del algoritmo Usuario-Usuario con kNN es $O(D * U)$ donde D es la dimensión de la matriz y U es la cantidad de usuarios, para el algoritmo Item-Item la complejidad es la misma solo que en vez de ser la cantidad de usuarios, es la cantidad de ítems I.[2][7] VSM con TF-IDF tiene una complejidad $O(T * D)$ donde T es la cantidad de términos y D es la cantidad de documentos[8], finalmente la complejidad computacional de Slope One es $O(U * I^2)$ donde U es la cantidad de usuarios e I es la cantidad de ítems[6].

Los algoritmos Usuario-Usuario, Item-Item y Slope One todos dependen y predicen calificaciones de los usuarios por lo tanto estos presenta el problema del coldstart mientras que el algoritmo VSM con TF-IDF no.[9]

IV. Discusión de resultados

Comparando el algoritmo Usuario-Usuario con el Item-Item el primero es mejor cuando existen más usuarios que ítems, y el segundo es mejor en una situación contraria, por lo tanto el algoritmo Usuario-Usuario presenta problemas de escalabilidad respecto al crecimiento de usuarios, por lo que este algoritmo no es apropiado para la aplicación, por otro lado el Item-Item mientras sean pocos ítems tendrá un tiempo de ejecución independiente de la cantidad de

usuarios eso hace que sea apropiado para el sistema que se requiere desarrollar. El VSM para sistemas de recomendación al ser una técnica basada en contenido no depende de la cantidad de usuarios solo dependería de los términos, los cuales serían las preferencias del usuario y la cantidad de documentos, que vendrían a ser los lugares, el Slope One depende de los usuarios y de los ítems, y tiene peor costo computacional que el Item-Item sin embargo Hu y Zhou muestran que tiene buenos resultados[6]

V. Conclusiones y Trabajo Futuro

Se concluye que la mejor opción para el sistema de recomendación para la aplicación para cuando salga es VSM, ya que así se evitará el coldstart y podrá recomendar lugares cuando esté disponible para el público, además de que tiene un costo computacional razonable para el proyecto que se está realizando.

Como trabajos futuros cuando la aplicación posea más datos se puede realizar un estudio más a fondo y cambiar el enfoque de sistemas de recomendación basados en contenido a sistemas de filtro colaborativo ya qué estos presentan el problema de sobre especialización y no puede brindar diversidad en las recomendaciones, un aspecto que es importante en las recomendaciones en turísticas[4].

VI. Bibliografía

[1] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, *Recommender Systems Handbook*. 2011.

[2] M. D. Ekstrand, "Collaborative Filtering Recommender Systems," *Found. Trends® Human–Computer Interact.*, vol. 4, no. 2, pp. 81–173, 2011.

[3] F. M. Hsu, Y. T. Lin, and T. K. Ho, "Design and implementation of an intelligent recommendation system for tourist attractions: The integration of EBM model, Bayesian network and Google Maps," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3257–3264, 2012.

[4] D. Gavalas, C. Konstantopoulos, K. Mastakas, and G. Pantziou, "Mobile recommender systems in tourism," *J. Netw. Comput. Appl.*, vol. 39, no. 1, pp. 319–333, 2014.

[5] L. Sharma and A. Gera, "A Survey of Recommendation System : Research Challenges," *Int. J. Eng. Trends Technol.*, vol. 4, no. 5, pp. 1989–1992, 2013.

[6] H. Hu and X. Zhou, "Recommendation of Tourist Attractions Based on Slope One Algorithm," *2017 9th Int. Conf. Intell. Human-Machine Syst. Cybern.*, no. 3, pp. 418–421, 2017.

[7] "scikit-learn." [Online]. Available: https://scikit-learn.org. [Accessed: 20-Jul-2019].

[8] W. Zhang, T. Yoshida, and X. Tang, "TFIDF, LSI and multi-word in information retrieval and text categorization," *Conf. Proc. - IEEE Int. Conf. Syst. Man Cybern.*, pp. 108–113, 2008.