

Proyecto Fase #1

Carlos Calderón (15219)

Jennifer Barillas(15307)

Jorge Azmitia(15202)

Algoritmos y estructuras de datos



Universidad del Valle de Guatemala

Facultad de Ingeniería

Guatemala

2016

Design Thinking

Empatía y definición

Tema de interés

Recomendaciones de catedráticos: Un sistema de recomendaciones basado en grafos. Que ayude a alumnos con sus asignaciones de catedráticos que se dan semestre a semestre.

Perfiles

Para el sistema de recomendaciones , se especificaron 2 perfiles en el ámbito académico. Los cuales son las personas con mayor repercusión y número en la Universidad del Valle de Guatemala. En este caso, en primer lugar se escogió a los alumnos, pues son ellos quienes pueden presentar alguna necesidad o inquietud respecto a la manera en que se asignan catedráticos de manera semestral. Asimismo, se escogió a catedráticos pues son ellos a quienes los alumnos quieren llegar en este proyecto.

Alumnos: Un grupo de personas que estudian una carrera en específico dentro de la universidad. A ellos se les preguntó para los posibles usos que se le podrían dar en su contexto actual y en qué época de su vida estudiantil les hubiera sido más útil esta herramienta.

Catedráticos: Un grupo de personas selecto de personas especializadas en algún ámbito científico o humanístico. Se encargan de impartir cursos que para estudiantes de licenciatura tienen una duración de 5 meses.

Entrevistas

(Adjuntas en otro documento por fines estéticos).

Mapa de empatía

(Adjunto por fines estéticos)

Insights

En base a las entrevistas con los distintos perfiles de personas con las que logramos interactuar en la universidad se plantean los siguientes insights:

- Los estudiantes confían en el criterio de sus compañeros respecto a escoger catedrático.
- En la mayoría de las veces escogen a la suerte los catedráticos, siendo esto más común en los alumnos de primer año.

Definición del Problema y Proyecto

Muchos estudiantes en la Universidad del Valle de Guatemala especialmente en primer año presentan dificultades al momento de escoger catedráticos en sus asignaciones semestrales. Este problema se presenta debido a que como es de esperar, los alumnos nuevos no tienen muchas referencias de catedráticos. De igual manera, esta situación tiene una significativa frecuencia en segundo año. Por otra parte, en los años posteriores ya la situación es un tanto distinta. Pues ya a ese punto de la carrera usualmente las opciones de asignación son más escasas, pues los cursos más específicos de la carrera son impartidos por un grupo más reducido de gente.

En base a los insights y a las entrevistas, que manifestaron los potenciales usuarios, se llevará a cabo un sistema de recomendaciones. Pues muchos usuarios basan su asignación en amistades cercanas, usualmente las amistades tienen rasgos similares con el usuario. Esto implica que las aptitudes que tienen son similares. Sin embargo, hay algunos casos en los que hay amistades con criterios distintos para asignar un profesor. Lo cual puede causar incomodidad para ciertas personas, pues no todas aprenden o bien no a todas les gusta recibir una clase de la misma forma. Por otra parte, muchos otros alumnos no saben mucha información respecto a ciertos catedráticos que imparten ciertos cursos, lo que igual en ocasiones los lleva a asignarse de manera trivial. Llevándolos en ocasiones a no estar totalmente conformes con su asignación. Así pues, el sistema de recomendaciones tomará en cuenta una diversidad de rasgos del usuario para que la recomendación sea significativa.

Ideación

Reflexión Lluvia de Ideas

- Cada usuario tiene parámetros diferentes para escoger a un buen catedrático, por lo tanto se planea tomar en cuenta las características del usuario y compararlas con las características de usuarios anteriores y así dar la mejor referencia posible. Esto hace menos subjetiva la selección.
- Como bien sabemos cada persona tiene diferentes métodos para seleccionar un catedrático acorde a sus necesidades, y como vimos en las entrevistas, la mayoría se

deja llevar por la experiencia de sus compañeros. Por lo tanto, se planea tener diferentes parámetros de los cuales el usuario podrá elegir cual es el que a él le interesa, ya sea comentarios, porcentaje de aprendizaje, dificultad del curso con dicho profesor, etc.

- Poniendo de base la segunda idea, uno de los parámetros será “dificultad del curso con dicho profesor”, pero se tomará en cuenta el rendimiento del estudiante a lo largo de su carrera. Esto se hará para que el usuario vea la veracidad de su argumento.
- También se tomará en cuenta la carrera de los estudiantes, ya que los estudiantes de la misma carrera en muchas ocasiones su forma de pensar es similar, o bien sus gustos.

Conclusión de Lluvia de Ideas

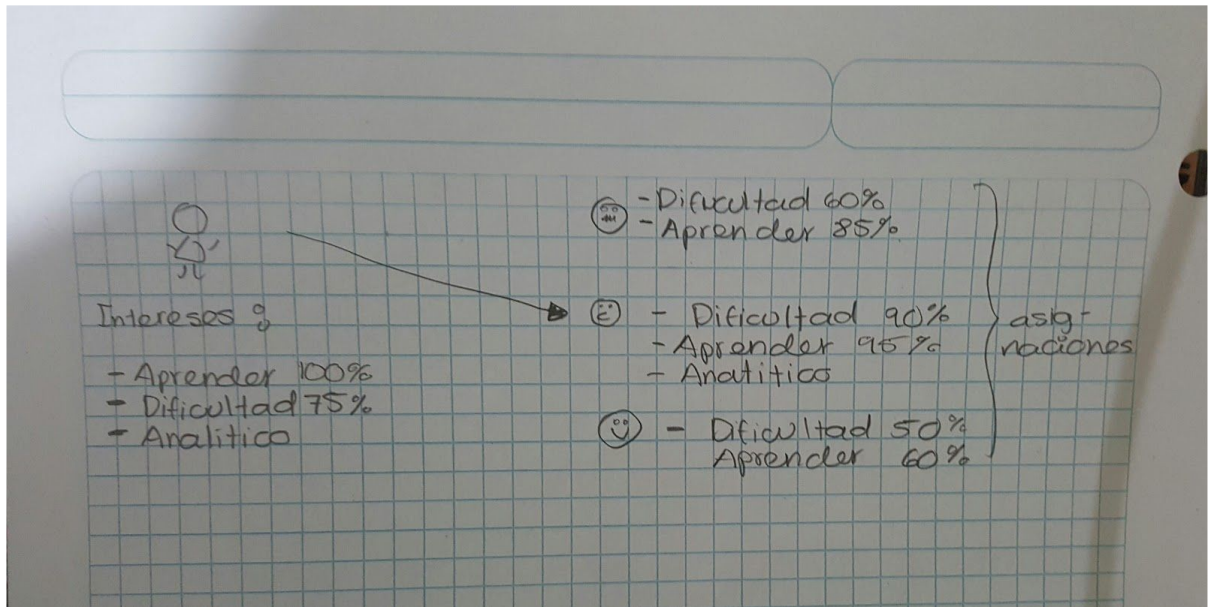
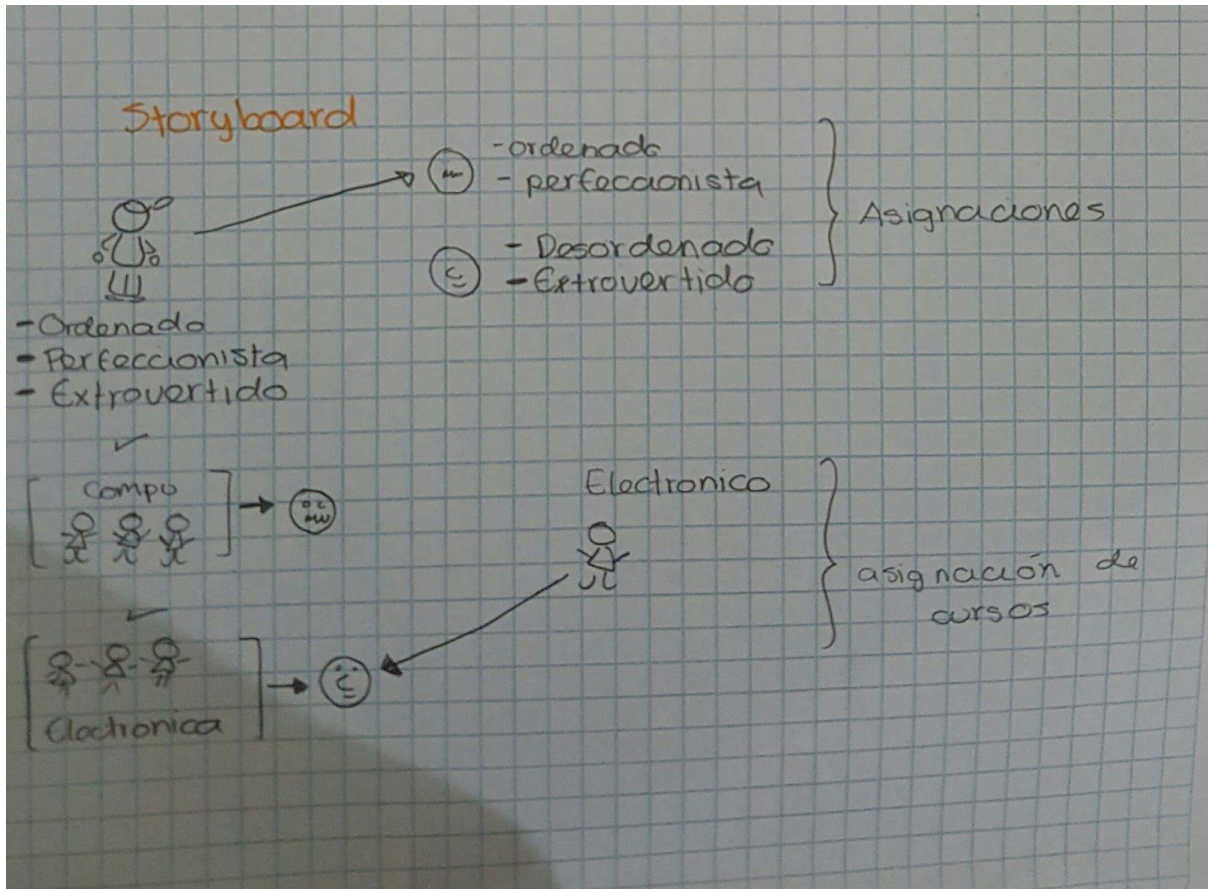
- Parámetros a utilizar:
 - Perfil del estudiante, aquí se mencionan las características que tiene, así como cualidades y defectos que tiene como persona/estudiante. Podemos decir como ejemplo calificativos como disciplinado, perfeccionista, etc.
 - Historial académico, en este aspecto se mencionan datos importantes del estudiante como lo son:
 - Carrera del estudiante.
 - Colegio del que se graduó.
 - Rendimiento académico.
- Los usuarios escogen catedráticos en base a recomendaciones de amigos. Con los cuales comparten gustos.

Evidencias de Lluvia de Ideas

lluvia de ideas

1. Conexión entre personas (Alumno \leftrightarrow Catedrático)
2. Mayoría de estudiantes de la misma carrera ganan con profesor & Usuario también.
3. Parametros a usar :
 - Comentarios
 - % Aprendizaje
 - % dificultad
4. Enlazar argumentos de los estudiantes y su rendimiento académico.
5. Darle importancia a los comentarios de los otros estudiantes.

Storyboard



Solución propuesta

Se propone llevar a cabo un programa con bases de datos basado en grafos. En el cual el usuario pueda tener una recomendación óptima de catedrático que se ajuste a sus necesidades. El programa va dirigido principalmente a gente de primer y segundo año, pues es donde principalmente hay una amplia gamma de catedráticos en cursos de distinta índole. Asimismo, es una etapa (principalmente en primer año) donde recién se comienza a tomar conocimiento sobre los catedráticos en la universidad.

Cabe destacar que en primera instancia, se planea basar las recomendaciones que el programa desplegará en base a opiniones pasadas. Para esto, se planea un Sistema basado en Filtrado Colaborativo. Ya que este consiste en ver qué usuarios son similares al usuario activo y a continuación, recomendar aquellos catedráticos que no han sido votados por el usuario activo y que han resultado bien valorados por los usuarios similares. Estas similitudes estarían basadas en ciertos rasgos característicos del perfil estudiantil del usuario.

Prototipar y testear

Prototipo

DATOS :

Catedrático :

Curso :

Intereses :

- Organización
- Analítica
- Despreocupado
- Claridad en la explicación
- Explicación gráfica
- trabajar en grupo
- trabajo individual

Dificultad

Aprendizaje

LISTO

Sugerencias

Nombre: Nombre 1

Características de interés:

Años de experiencia:

-

-

-

-

-

Investigación de algoritmos existentes

Dentro de los sistemas de recomendaciones, existen diferentes técnicas de cómo filtrar la información almacenada para obtener las recomendaciones deseadas. Una de ellas es la técnica del filtrado colaborativo, la cual se basa en las similitudes entre los usuarios o los objetos. Más concretamente, se basa en la idea de que si a dos personas les gusta el mismo objeto, es muy probable que tengan la misma opinión en el resto de objetos que les gustan a cada uno. Por ello los algoritmos investigados se basan en esta técnica.

Nearest Neighborhood

Uno de los algoritmos más usado comúnmente en sistemas recomendadores es el Nearest Neighborhood. En una red social, un usuario particular usa un vecino con gustos e intereses similares que pueden ser encontrados calculándolos con una correlación de Pearson, colectando la información de preferencia de los primeros N-vecinos para un usuario en particular, (pesado por su similitud), la preferencia del usuario se puede predecir usando ciertas técnicas.

Heat Spreading

El algoritmo Heat Spreading es un algoritmo que destaca, principalmente, por ofrecer recomendaciones diversas a los usuarios. Para ello, el algoritmo distribuye los pesos utilizando la matriz que se presenta seguidamente, de la cual podemos destacar su normalización por filas:

$$W_{\alpha\beta}^H = \frac{1}{k_\alpha} \sum_{j=1}^u \frac{a_{\alpha j} a_{\beta j}}{k_j}$$

Esto es equivalente a distribuir, en dos pasos, los pesos de los objetos sobre el grafo bipartito. En un primer paso, cada usuario recibe un peso igual a la media de los pesos que poseen sus vecinos (dividiendo la suma de los pesos de todos los vecinos por el grado del usuario, k_i). En el segundo paso, se realiza el mismo procedimiento, siendo los objetos los que reciben un peso igual a la media de los pesos que poseen sus vecinos (dividiendo la suma de los pesos de todos los vecinos por el grado del objeto, k_α).

Pseudocódigo

Entrada: Un grafo y un usuario.

Salida: conjunto de objetos

- 1: A cada vecino del usuario objetivo, asignar un peso inicial de 1.
- 2: para cada objeto o que tenga un peso diferente de 0 hacer
- 3: para cada usuario u vecino de objeto hacer
- 4: Asignar a u el peso de o dividido entre el grado de u .
- 5: fin para cada
- 6: fin para cada
- 7: para cada usuario u que tenga un peso diferente de 0 hacer
- 8: para cada objeto o vecino de usuario hacer
- 9: Asignar a o el peso de u dividido entre el grado de o .
- 10: fin para cada
- 11: fin para cada
- 12: Ordenar los objetos decrecientemente según su peso final.

Probability Spreading

El algoritmo Probability Spreading es un algoritmo que destaca, principalmente, por ofrecer recomendaciones precisas a los usuarios. Para ello, el algoritmo distribuye los pesos utilizando la matriz que se presenta seguidamente, de la cual podemos destacar su normalización por columnas:

$$W_{\alpha\beta}^P = \frac{1}{k_\beta} \sum_{j=1}^u \frac{a_{\alpha j} a_{\beta j}}{k_j}$$

Esto es equivalente a distribuir, en dos pasos, los pesos de los objetos sobre el grafo bipartito. En un primer paso, para cada objeto se distribuye su peso repartido equitativamente entre todos sus usuarios vecinos (dividiendo el peso por el grado del objeto, k_β). En el segundo paso, los usuarios que han recibido los pesos los redistribuyen a sus objetos vecinos, repartiéndolos equitativamente también (dividiendo el peso por el grado del usuario, k_j).

Pseudocódigo

Entrada: Un grafo y un usuario.

Salida: Conjunto de objetos recomendados, en orden descendiente.

- 1: A cada vecino del usuario objetivo, asignar un peso inicial de 1.
- 2: para cada objeto o que tenga un peso diferente de 0 hacer
- 3: para cada usuario u vecino de objeto hacer
- 4: Asignar a u el peso de o dividido entre el número de vecinos de o.
- 5: fin para cada
- 6: fin para cada
- 7: para cada usuario u que tenga un peso diferente de 0 hacer
- 8: para cada objeto o vecino de usuario hacer
- 9: Asignar a o el peso de u dividido entre el número de vecinos de u.
- 10: fin para cada
- 11: fin para cada
- 12: Ordenar los objetos decrecientemente según su peso final.

Algoritmo híbrido

El algoritmo Probability Spreading ofrece un gran precisión a la hora de realizar las recomendaciones en detrimento de una baja diversidad; mientras que, el algoritmo Heat Spreading ofrece unas recomendaciones con un grado de diversidad mucho mayor pero penalizando la precisión. Por lo tanto, con el objetivo de lograr un algoritmo que obtenga tanto recomendaciones precisas como diversas, se ha construido un algoritmo híbrido a partir de los dos anteriores para intentar beneficiarse de las características de ambos. Dicho algoritmo distribuye los pesos utilizando la matriz que se presenta seguidamente:

$$W_{\alpha\beta}^{H+P} = \frac{1}{k_{\alpha}^{1-\lambda} k_{\beta}^{\lambda}} \sum_{j=1}^u \frac{a_{\alpha j} a_{\beta j}}{k_j}$$

Esto es equivalente a distribuir, en dos pasos, los pesos de los objetos sobre el grafo bipartito. Más concretamente, se trata de realizar los mismos pasos que en los anteriores algoritmos pero simultáneamente, otorgando más poder a un algoritmo o otro según el factor de lambda. De esta manera, se puede afinar λ para aumentar las características de un algoritmo o otro, según convenga o nos interese.

Pseudocódigo

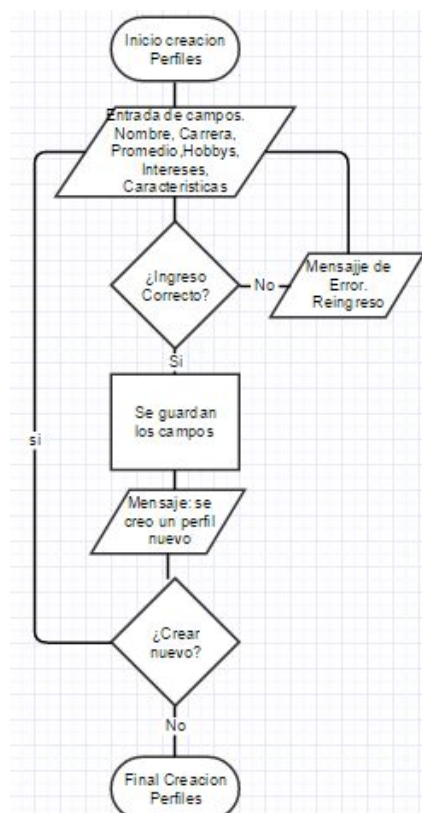
Entrada: Un grafo, un usuario y un número decimal entre 0 y 1 (λ).

Salida: conjunto de objetos

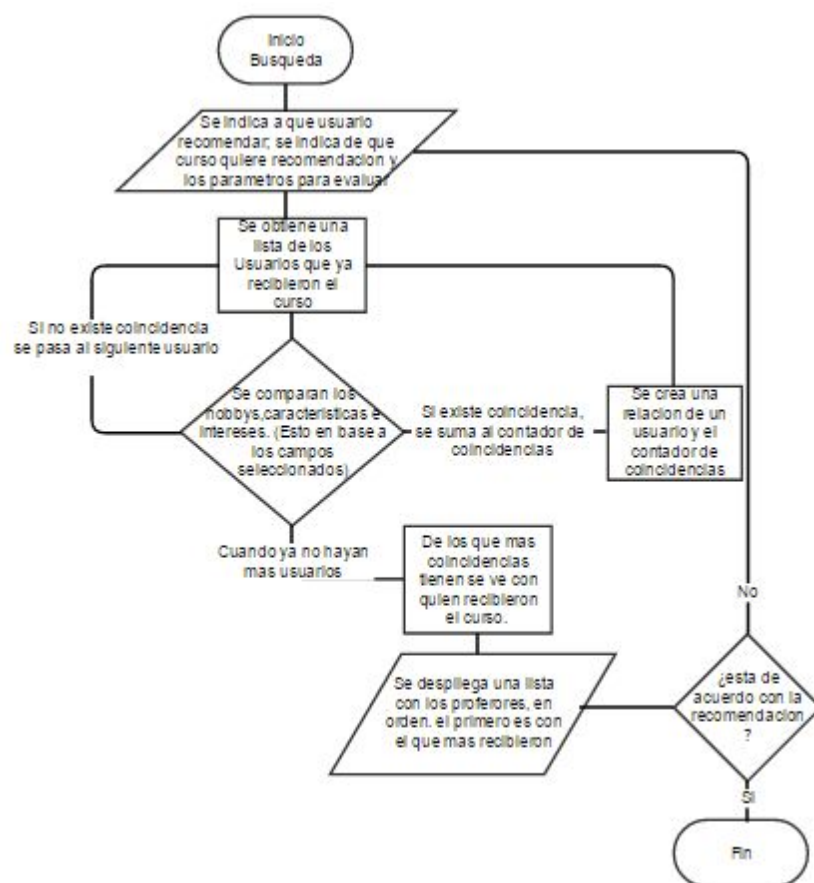
- 1: A cada vecino del usuario objetivo, asignar un peso inicial de 1.
- 2: para cada objeto o que tenga un peso diferente de 0 hacer
- 3: para cada usuario u vecino de objeto hacer
- 4: grado := grado del usuario u
- 5: vecinos := número de vecinos del objeto o
- 6: tmp := $\text{grado}^{(1-\lambda)} * \text{vecinos}^{\lambda}$
- 7: Asignar a u el peso de o dividido entre tmp.
- 8: fin para cada
- 9: fin para cada
- 10: para cada usuario u que tenga un peso diferente de 0 hacer
- 11: para cada objeto o vecino de usuario hacer
- 12: grado := grado del objeto o
- 13: vecinos := número de vecinos del usuario u
- 14: tmp := $\text{grado}^{(1-\lambda)} * \text{vecinos}^{\lambda}$
- 15: Asignar a o el peso de u dividido entre tmp.
- 16: fin para cada
- 17: fin para cada
- 18: Ordenar los objetos decrecientemente según su peso final.

Pseudocódigo/diagrama inicial a utilizar

1. Creación perfiles :



2. Búsqueda:



Base de datos

En nuestro caso, inicialmente nuestra base de datos cuenta con 7 personas. Para tener un panorama general de cómo funcionarán las recomendaciones. El primer paso es insertar personas con el comando “Create” de Neo4J, con varios campos que representan los intereses y perfiles del usuario. Específicamente son 4 usuarios que simulan a usuarios pasados. Luego 2 catedráticos, y un alumno nuevo. Así pues los nodos son de tipo “Alumno” y tipo “Maestro”, más un par de características propias de usuarios. En esta base de datos al ser un modelo a escala, solo se incluyeron algunos datos de usuarios. Estos no son todos los que se van a usar, pero como se mencionó antes, es para dar un panorama general.

Posteriormente se hacen las respectivas relaciones. Y por último, se procede a recomendar en base a algún parámetro, un catedrático al usuario nuevo.

En el ejemplo del Script (adjunto) se Andrea es un usuario nuevo, que puede buscar recomendación en base a cómo se asignaron alumnos de su misma carrera o mismo perfil.

Referencias

- [1] Kantor, P. B., Rokach, L., Ricci, F., & Shapira, B. (2011). Recommender systems handbook. Springer.
- [2] Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2010). Recommender systems: an introduction. Cambridge University Press
- [3] Martinez,S. (2014). Diseño e implementación de un sistema de recomendaciones. Universitat Politècnica de Catalunya (UPC) - BarcelonaTech. 107 pp.