PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC3633 Sistemas Recomendadores (2021-2)

Tarea 1

Indicaciones

- Fecha de entrega: **Miércoles 15 de septiembre de 2021, 20:00 horas.**
- La tarea debe realizarse **individualmente o en grupos de máximo dos personas**. La copia será sancionada con una nota 1.1 en la tarea, además de las sanciones disciplinarias correspondientes.
- Entrega a través del repositorio personal en GitHub asociado a esta clase y tarea, por anunciar. Basta que lo entregue uno de los miembros del grupo, pero el README.md debe identificar claramente a lo(a)s otros miembros del grupo.
- Cada hora o fracción de atraso descuenta 0.5 puntos de la nota obtenida, llegando a 1.0 en 6 horas. Se considera como entrega el último *commit* presente en el repositorio, es decir, la hora en la que este hace presente en GitHub (no su hora de creación). No se revisarán *commits* anteriores.
- Se sugiere hacer la tarea en Google colab o en jupyter notebooks para facilitar la revisión. Deberán entregar estos notebooks ejecutados como parte de su código.

Objetivo

En esta tarea tendrán la oportunidad de poner en práctica sus conocimientos sobre Sistemas Recomendadores. En particular, exprimentarán con recomendación no personalizada, basada en feedback implícito y basada en contenido.

Dataset

En esta tarea utilizarán un subset del dataset Anime Recommendation Database de Kaggle que contiene información de las interacciones de usuarios con series, películas y comics de animé.

El dataset con el que trabajarán consiste en:

- Dataset de train (*train.csv*): **1,475,055** registros que contienen id del usuario y id del anime con el que interactuó. Descargar aquí.
- Dataset de validación (*val.csv*): **30,102** registros que contienen id del usuario y id del anime con el que interactuó. Descargar aquí.

- Dataset de test (*test_users.csv*): una lista de **14,893** IDs de usuarios. Para cada uno de estos usuarios debes generar una lista de 20 recomendaciones. Recuerda que con tus predicciones sobre este dataset compites por un bono de 5 décimas (nota máxima 7.0) si quedas entre los 5 mejores grupos del curso en métricas MAP@20 y nDCG@20. Descargar aquí.
- Información adicional de animes (*anime_info.pkl*): información adicional de **2,326** animes como título, sinopsis, géneros y vector de características o embeddings que serán utilizados en la actividad de recomendación basada en contenido. Viene en formato *pkl* por lo que para abrirlo necesitan utilizar pickle, pandas también tiene la funcionalidad para abrir archivos en formato pickle. Descargar aquí.

Librerías

Pueden utilizar cualquier librería en python implementadas para recomendación. Las más utilizadas son pyreclab, surprise e implicit, pero esto queda a su criterio.

Para recomendación basada en contenido pueden utilizar funciones de similaridad y de reducción de dimensionalidad ya implementadas en librerías como scikit-learn.

Métricas de evaluación modelos

En esta tarea las métricas que se les pide para evaluar el desempeño de todos los modelos de recomendación son **ndcg@10, ndcg@20, ndcg@30, MAP@10, MAP@20 y MAP@30**.

Importante para la evaluación de todos los modelos de recomendación
Considere como relevantes aquellos animes a los que el usuario le dió un rating mayor o igual a 7 y no relevantes en caso contrario.

Actividad 1: Exploración de datos (15%)

En esta actividad se le pide hacer el siguiente análisis exploratorio sobre los datos de training:

- Grafique la distribución del número de interacciones por usuario, identifique los ids de los 5 usuarios más activos en el dataset. Comente la forma de la distribución obtenida y qué porcentaje de las interacciones han sido hechas por estos 5 usuarios.
- Grafique la distribución de interacciones por animé. Identifique los nombres y ids de los 5 animes que han sido más vistos. Comente la forma de la distribución y qué porcentaje de las interacciones han sido sobre estos 5 animes.
- Genere una tabla con la cantidad de usuarios distintos, número de items distintos, promedio y desviación estándar de animes por usuario, promedio y desviación estándar de usuarios por animes y densidad del dataset (o *sparsity*) en cuanto a interacciones.

Actividad 2: Recomendación no personalizada (15%)

En esta actividad el objetivo es realizar dos recomendaciones. Primero se pide recomendar los 30 animes más populares (*most popular*) y luego realizar una recomendación de 30 animés escogidos de

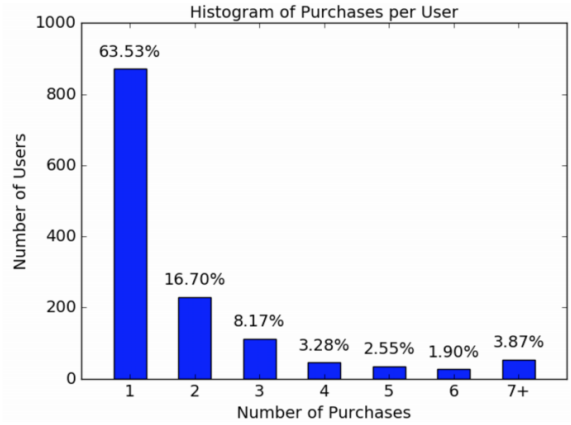

Figure 1: Ejemplo de gráfico de distribución, en este caso de compras por usuario. Haga algo similar para la cantidad de interacciones en el dataset de training de la tarea.

manera aleatoria (*random*). Por lo general estos métodos no personalizados se utilizan como baseline para comprobar que los métodos utilizados funcionan bien y tienen un buen rendimiento.

Se les pide calcular métricas de evaluación en el dataset de validación para ambos métodos no personalizados: *random* y *most popular*.

Actividad 3: Recomendación basada en feedback implícito (20%)

En esta actividad el objetivo es recomendar basándose en las animes con las que ha interactuado el usuario y su grado de relevancia, dado por el rating que ha recibido.

En este caso utilizarán dos modelos: Factorización Matricial optimizada con Alternate Least Squares (ALS) y Factorización Matricial optimizada con Bayesian Personalized Ranking (BPR).

Se le pide:

- Mostrar un análisis de sensibilidad de resultados en el dataset de validación para métricas MAP@10 y nDCG@10 modificando dimensión de factores latentes (50,100,200,500,1000) y el algoritmo de optimización utilizado (ALS o BPR). Grafique cada uno y comente los resultados.
- Reportar tiempos de entrenamiento en cada uno de los casos.

Actividad 4: Recomendación basada en contenido (20%)

En esta actividad el objetivo es recomendar basándose en el contenido de los elementos con que ha interactuado el usuario. Es decir si el usuario ha visto todas las series de *Dragon Ball* en el dataset de entrenamiento, lo más probable es que en el futuro siga viendo series de contenido similar.

Para ello les entregamos el título, synopsis y géneros del anime y un vector que representa su contenido, más bien conocidos como embeddings calculados utilizando *universal sentence encoding* disponibilizado abiertamente por Google, para más detalles sobre este método ir al siguiente link.

El proceso que se pide en esta actividad desde procesar los embeddings hasta realizar la recomendación es el siguiente:

1. Reducir dimensionalidad de los vectores o embeddings de animes utilizando PCA, el objetivo de este paso es para que el cálculo de la similaridad sea lo más eficiente posible al momento de hacer la recomendación.
2. Calcular una representación vectorial de cada usuario como el promedio de vectores de animes con los que ha interactuado el usuario en el dataset de entrenamiento.
3. Hacer la recomendación calculando alguna métrica de similaridad entre el vector del usuario y los vectores de animes.

Se les pide:

- Hacer un análisis de sensibilidad de resultados en el dataset de validación para métricas MAP@10 y nDCG@10 modificando dimensión de los vectores luego de reducir dimensionalidad (10,50,100) y métricas de similaridad (coseno, euclideana y manhattan). Grafique cada uno y comente los resultados.

Actividad 5: Comparación de métodos (15%)

En esta actividad se le pide:

- Hacer una tabla comparativa de los resultados de las métricas solicitadas en el dataset de validación para el mejor modelo de recomendación (con mejor combinación de hiperparámetros) de cada uno de los métodos vistos, es decir recomendación no personalizada (Random y Most Popular), basada en interacciones (Matrix Factorization ALS y Matrix Factorization BPR) y basada en contenido (solo el que obtuvo los mejores resultados). Recuerde mostrar en la tabla la mejor combinación de hiperparámetros de cada modelo que los llevaron a obtener dichos resultados.
- Hacer un análisis y discusión de los resultados que expliquen posibles razones que puedan estar incidiendo en los resultados obtenidos.
- Seleccione e indique el mejor de todos los métodos (a su parecer) y con este, genere 10 recomendaciones por cada usuario del dataset de testing. La lista de recomendaciones **debe** ser entregada en este formato:

```
{
"user_id1": [anime_id1, anime_id2192, ...],
"user_id2": [anime_id121, anime_id234, ...],
"user_id3": [anime_id191, anime_id223, ...],
...
}
```

Actividad 6: Ejemplos de recomendación de Anime (15%)

En esta sección se les pide mostrar ejemplos de las recomendaciones generadas en la actividad anterior.
Se les pide

- Seleccionar 3 usuarios de ejemplo del dataset de testing y desplegar información de los 10 elementos que le fueron recomendados en la étapa anterior. En particular, se pide que indiquen, para cada una de estas recomendaciones, el título del anime, sinopsis y géneros.

- Seleccionar e indicar animes previamente vistos por cada usuario escogido. (La cantidad de ejemplos elegidos queda a criterio de cada estudiante)
- Comentar brevemente los resultados.

Entregables

La tarea deberá ser entregada a través del repositorio de su grupo y tarea en **GitHub classroom**. Siga el siguiente link y las instrucciones para crear el repositorio de su grupo e inscribirse: link a tarea 1.

Se deberá ENTREGAR un informe en formato PDF, así como código en uno o varios **Jupyter Notebook con todas las celdas ejecutadas**, es decir, no se debe borrar el resultado de las celdas antes de entregar. Si las celdas se encuentran vacías, se asumirá que la celda no fue ejecutada. Es importante que toda la información solicitada de parámetros y análisis tengan una explicación, es decir, no basta con el *output* de una celda para responder una pregunta, se debe explicar qué se está respondiendo.

Informe. En el repositorio de su grupo debe estar un informe en formato PDF que contenga el desarrollo de cada una de las actividades solicitadas.

Código. Por cada uno de los métodos solicitados debe entregar en su repositorio el código que permita replicar los resultados obtenidos. Se solicita entregar uno o varios jupyter notebooks que permitan replicar experimentos.

Es obligatorio agregar un archivo README.md al repositorio de su tarea que permita entender la estructura de archivos y detalles necesarios para replicar los experimentos realizados.