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

IIC3633 Sistemas Recomendadores (2023-2)

Tarea 1

Indicaciones

- Fecha de entrega: **Viernes 29 de septiembre de 2023, 20:00 horas.**
- La tarea debe realizarse **en grupos de máximo tres personas**. La copia será sancionada con una nota 1.1 en la tarea, además de las sanciones disciplinarias correspondientes.
- Entrega a través de CANVAS.
- Se debe hacer la tarea en Google colab o en jupyter notebooks para facilitar la revisión. Deberán entregar estos notebooks ejecutados como parte de su código junto con el informe en pdf, un README del código y el submission de la actividad 7.

Objetivo

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

Dataset

En esta tarea utilizarán un subset del dataset de Yelp que contiene información de las interacciones de usuarios con negocios de comida.

El dataset con el que trabajarán consiste en:

- Dataset de train **yelp_train.csv**: 69,674 registros que contienen id del usuario, id del negocio con el que interactuó, rating, id del review, texto del review, etiquetas adicionales (useful, funny, cool) y fecha. Descargar aquí.
- Dataset de validación **yelp_val.csv**: 9,028 registros que contienen id del usuario, id del negocio con el que interactuó, rating, id del review, texto del review, etiquetas adicionales (useful, funny, cool) y fecha. Este dataset se utilizará para reportar sus resultados en el informe. Descargar aquí.

- Usuarios de test **yelp_test_user_ids.csv**: 9,028 registros sólo con id de usuario. Uno de los entregables de esta tarea debe ser un archivo *recomendaciones.json* con una lista de recomendaciones por usuario. Estas recomendaciones serán evaluadas con interacciones ocultas, lo que nos permitirá rankear las tareas (similar a las competencias de Kaggle). Más detalles sobre esto en el apartado de la Actividad 7. Descargar aquí.
- Metadata de los negocios **businesses_metadata.csv**: contiene id del negocio, nombre, dirección, ciudad, estado, código postal, latitud, longitud, está abierto, atributos, categoria y horario de atención. Descargar aquí.
- Fotos de los negocios **businesses_photos.zip**: carpeta que contiene todas las fotos de los negocios. El nombre del archivo contiene el id de la foto y la extensión *.jpg* de la imagen. Descargar aquí.
- Metadata de las fotos de los negocios **photos_metadata.csv**: contiene el id del ítem, id la foto del ítem, caption y label. Notar que un ítem puede tener más de una foto. Descargar aquí.
- Embeddings de las fotos **embeddings_photos.pkl**: representaciones vectoriales obtenidas con ResNet50 de las fotos de los negocios. La información viene como un diccionario donde la llave es el id de la foto y el valor es el embedding de la foto. Un negocio puede tener muchas fotos. Descargar aquí.
- Vectores de las caracteristicas de los negocios **business_tfidf_dict.pkl**: obtenidos utilizando TF-IDF recibiendo la concatenacion del nombre del negocio, categoria y ciudad. La información viene como un diccionario donde la llave es el id del negocio y el valor es el vector obtenido con TF-IDF. Descargar aquí.

Librerías

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

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

También se permite el uso de librerías como PyTorch o Tensorflow, para aquellos grupos que deseen explorar algoritmos basados en redes neuronales y aprendizaje profundo (ver actividades de bonus), aunque esto es opcional y queda a criterio de cada grupo.

Métricas de evaluación de 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 items a los que el usuario le dio un rating mayor o igual a 3 y no relevantes en caso contrario para recomendacion basada en feedback implícito.

Actividad 1: Exploración de datos (13%)

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.

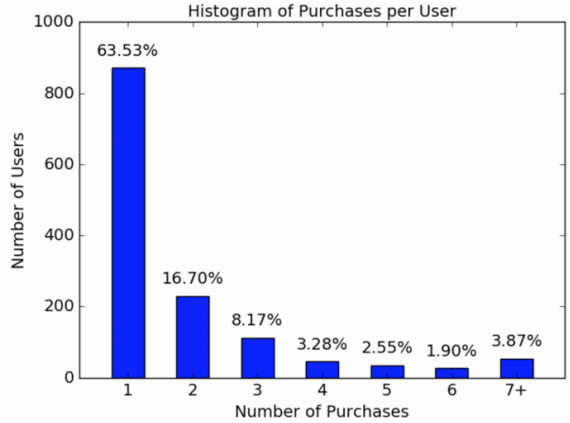

Figure 1: Ejemplo de gráfico de distribución, en este caso de compras por usuario. Haga algo similar para la cantidad de interacciones en el dataset de training de la tarea.

- Grafique la distribución de interacciones por negocio. Identifique los nombres e ids de los 5 negocios que han sido más evaluados. Comente la forma de la distribución y qué porcentaje de las interacciones han sido sobre estos 5 negocios.
- Genere una tabla con la cantidad de usuarios distintos, número de items distintos, promedio y desviación estándar de item por usuario, promedio y desviación estándar de usuarios por item y densidad del dataset (o *sparsity*) en cuanto a interacciones.

Actividad 2: Recomendación no personalizada (13%)

En esta actividad el objetivo es realizar dos recomendaciones. Primero se pide recomendar los 30 negocios más populares (*most popular*) y luego realizar una recomendación de 30 negocios escogidos de 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 (14%)

En esta actividad el objetivo es recomendar basándose en los negocios con los 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 (22%)

En esta actividad el objetivo es recomendar basándose en el contenido de los negocios con los que ha interactuado el usuario. Es decir si el usuario ha interactuado con restaurantes de hamburguesas en el dataset de entrenamiento, lo más probable es que en el futuro siga visitando negocios similares. Para ello les entregamos vectores TF-IDF de los nombres de los negocios, la categoria y la ciudad, y los embeddings de las imagenes de los negocios.

El proceso que se pide en esta actividad es:

1. Hacer la recomendación calculando alguna métrica de similaridad entre el vector del usuario y los vectores TF-IDF de los items.
2. Hacer la recomendación calculando alguna métrica de similaridad entre el vector del usuario y los embeddings de las fotos de los items.

La forma como representar el vector del usuario queda abierta a su criterio. Lo mismo para representar un embedding por item, dado que un negocio puede tener más de un embedding de una foto.

En esta actividad se les pide:

- Reportar los resultados de recomendación utilizando las metricas de evaluación solicitadas, utilizando las caracteristicas de los negocios de los vectores tf-idf.
- Reportar los resultados de recomendación utilizando las metricas de evaluación solicitadas, utilizando los embeddings de las imágenes de los negocios.

Actividad 5: Comparación de métodos (14%)

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), basada en contenido de texto y basada en contenido de imágenes. 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.

Actividad 6: Ejemplos de recomendación de negocios (14%)

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

En esta actividad se les pide seleccionar 3 usuarios que consideren representativos del dataset de validación y mostrar:

- Imágenes, categorías y nombres de negocios con los que estos usuarios interactuaron en el set de train.
- Imágenes, categorías y nombres de restaurantes que se les recomendaron a estos usuarios.
- Imágenes, categorías y nombres de negocios con los que estos usuarios deberían haber interactuado en el set de validación.

Comentar los resultados en términos cualitativos de si hacen sentido las recomendaciones y proponer como se podría mejorar.

Actividad 7: Recomendaciones a usuarios en el set de test (10%)

En esta sección se les pide entregar un archivo *recomendaciones.json* con el siguiente formato:

```
{
 "userId1": ["itemId1", "itemId2", ...],
 "userId2": ["itemId22", "itemId33", ...],
 ...
 "userIdN": ["itemId42", "itemId4", ...]
}
```

Las llaves del JSON deben ser los 9,028 user_id's del archivo **yelp_test_user_ids.csv**, y los valores deben ser arreglos con los ids de los negocios a recomendar. **Por cada usuario deben recomendar 30 negocios**. Estos pueden ser cualquiera de los negocios listados en el archivo **business_metadata.csv**. Esto nos permitirá calcular ndcg@10, ndcg@20, ndcg@30, MAP@10, MAP@20 y MAP@30 con interacciones ocultas, promediar estas 6 métricas y rankear las tareas.

Las tareas cuyas recomendaciones logren estar dentro del top 3 en el promedio de las 6 métricas al evaluarlas contra las interacciones ocultas tendrán un bonus de 3 décimas sobre la nota.

Importante: No se aceptarán archivos con formato equivocado porque en caso contrario no se podrán obtener las métricas para calcular su ranking.

En el informe deben explicar qué método usaron para generar las recomendaciones y justificar por qué escogieron dicho método. Además, el código utilizado para generar las recomendaciones debe ser incluído.

Actividad Bonus 1: Exploración de otros algoritmos de recomendación

El objetivo de este bonus es premiar con 2 décimas aquellas tareas que decidan ir más allá de lo estrictamente solicitado, al explorar otros algoritmos o métodos de recomendación.

Ejemplos de posibles experimentos válidos para este bonus:

- Reemplazar TF-IDF por una mejor técnica más acorde al estado del arte.
- Reemplazar los embeddings ResNet50 de las fotos por otros embeddings obtenidos con una mejor técnica más acorde al estado del arte.
- Utilizar un algoritmo de feedback implícito diferente o más sofisticado que los solicitados en la Actividad 3 (por ej. basado en redes neuronales profundas).
- Etcétera.

En el informe y en el código deben explicar el o los métodos explorados, justificar su elección, y reportar las métricas obtenidas en el set de validación.

Nota 1: Pueden utilizar modelos de esta actividad para generar las recomendaciones de la Actividad 7.
Nota 2: Implementar un método de ensamblaje no cuenta como parte de este bonus (ver siguiente bonus).

Actividad Bonus 2: Métodos de Ensamblaje

Como actividad opcional con un premio de 2 décimas, deben generar un modelo de ensamblaje (ensamble learning) que combine al menos uno de los métodos de recomendación basada en feedback implícito (ALS, BPR u otra variante si hacen el bonus 1) con al menos uno de los modelos de recomendación basada en contenido (vectores de texto y/o imagen u otra variante si hacen el bonus 1).

- Un método de ensamblaje es un algoritmo que combina dos o más algoritmos de aprendizaje para mejorar su estabilidad o predictibilidad. Existen distintas formas de realizar un modelo de ensamblaje (bagging, random forests, AdaBoost, etc.). Ustedes pueden elegir la forma que les parezca más conveniente para la realización de este ejercicio. En el siguiente link pueden encontrar un tutorial de Simplilearn que les puede servir para entender el concepto y las distintas formas de realizar ensemble learning.
- Pueden utilizar el modelo que obtengan en esta actividad para generar las recomendaciones de la Actividad 7.

En el informe y en el código deben explicar el o los métodos de ensamblaje utilizados, justificar su elección, y reportar las métricas obtenidas en el set de validación.

Entregables

La tarea deberá ser entregada a través de CANVAS por alguno de los integrantes y se debe subir un archivo zip que contenga el informe, el código en jupyter notebooks, un README que contenga una explicación de su código y el submission de test en un archivo llamado *recomendaciones.json*. El código debe tener 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, análisis y resultados que ustedes reporten en su informe esté debidamente respaldada por su código. Es decir, el código y el informe deben ser consistentes, de manera que el código muestre claramente cómo se calculó cada resultado y cómo se generó cada gráfico reportado en el informe ya sea para las secciones de bonus como las evaluaciones obligatorias. Por ende, es muy importante que el código esté ordenado y debidamente comentado.