Taller 2: Modelos de recomendación híbridos y evaluación

Mateo Zapata López, Iván Arturo Salazar Ortiz, Nicole Bahamon Martínez
MINE-4201: Sistemas de Recomendación
Universidad de los Andes, Bogotá, Colombia
{ia.salazar, m.zapatal2, n.bahamon}@uniandes.edu.co
Fecha de presentación: mayo 7 de 2023

Último Commit del Repositorio: https://github.com/iaSalazar/SDR-MINE/commit/4087c22161260bafa5fa3bc3ec66b4ca009b7f38

Tabla de contenido

1 Introducción

Los modelos hibrido de sistemas de recomendación son modelos los cuales se basan en el uso de modelos basados en conocimiento y modelos basados en contenido, la idea con los modelos ensamble o híbridos es poder solucionar los problemas que cada modelo tiene por separado y tener mejores recomendaciones para los usuarios. Hay tres métodos principales para crear un sistema de recomendación hibrido:

- Diseño ensamble: es basado en ensamblar un modelo basado en contenido y un modelo colaborativo en una sola salida robusta.
- Diseño monolítico: en este método se usan los modelos basados en contenido y colaborativo para realizar una aproximación de los ratings obtenidos por cada modelo que compone la hibridación, el problema actual de este es que integra varias fuentes de datos y no es fácil visualizar los componentes individuales de cada modelo.

- Sistema mixto: este método usa diferentes algoritmos de sistemas de recomendación como una caja negra, es decir no se logra identificar los que realiza el modelo.

Los modelos híbridos de sistemas de recomendación pueden clasificarse en 7 categorías: Ponderado, Conmutación, Cascada, Aumentación de características, Combinación de características, Meta nivel y Mixto.

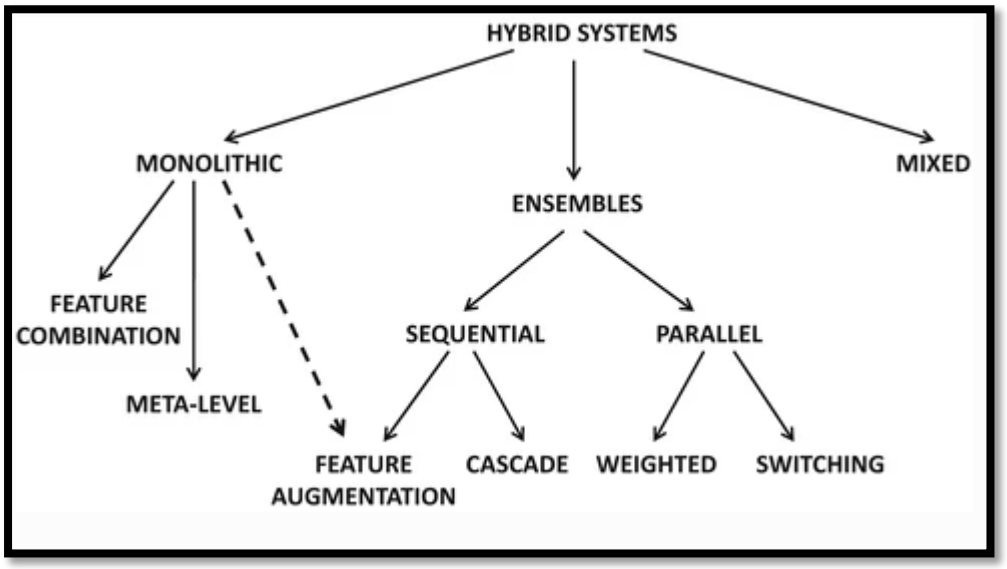


Imagen 1. Taxonomía de los sistemas híbridos de recomendación.

En este documento, se planea el desarrollo de un sistema de recomendación hibrido planteando objetivos organizacionales y de las partes interesadas del negocio haciendo uso del modelo conceptual de Jannach. Para este proyecto se hace uso del conjunto de datos de Yelp, este conjunto almacena información respecto a distintos negocios, sus características y la interacción que estos han tenido con distintos usuarios.

2 Conocimiento del dataset de trabajo

El conjunto de datos es de Yelp la cual es una plataforma que se creó en octubre de 2004 en EEUU por colaboradores de PayPal, la finalidad es conectar a las personas con negocios locales cerca de la ubicación del usuario, la plataforma clasifica las reseñas en dos categorías (recomendadas y no recomendadas).

El conjunto de datos este compuesto por 6 archivos tipo JSON que almacenan información de 6’990.280 reseñas, 150.346 negocios con 200.100 fotos que comprenden 11 áreas metropolitanas. Para los negocios se almacenan distintos atributos como horas de servicio, disponibilidad de parqueo, ambientes, entre otros. Cada uno de los conjuntos de datos se relacionan según el siguiente diagrama:

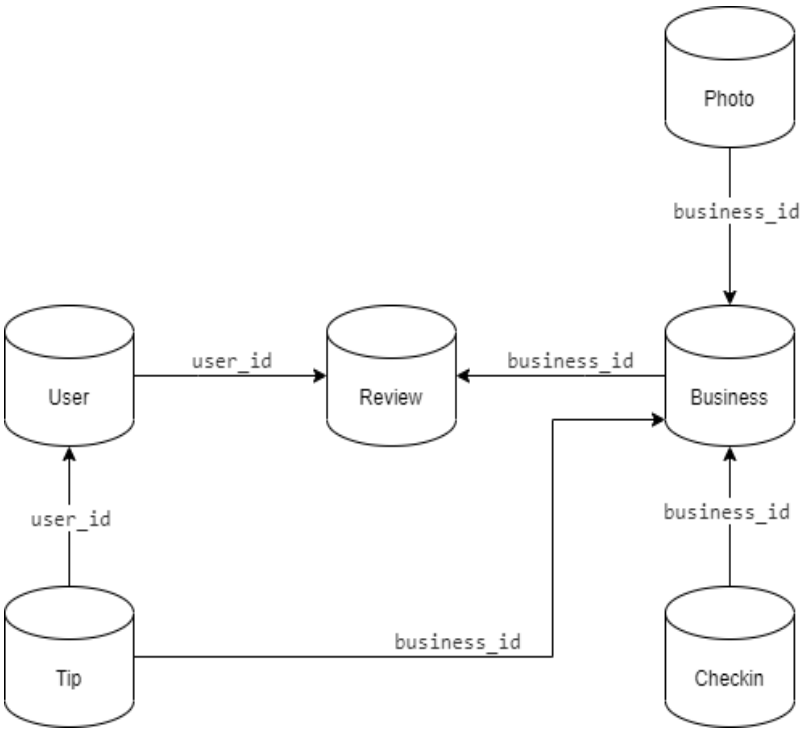


Figura 1. Esquema del dataset de Yelp

- **business.json:** Contiene información del negocio.
    - business_id: identificador único y llave local del archivo, campo tipo string de 22 caracteres.
    - name: nombre del negocio, campo tipo string.
    - address: dirección del negocio, campo tipo string.
    - city: ciudad del negocio, campo tipo string.
    - state: estado donde se encuentra el negocio, campo tipo string.
    - postal code: código postal de la ubicación del negocio, campo tipo string.
    - latitude: latitud de la ubicación del negocio, campo tipo float.
    - longitude: longitud de la ubicación del negocio, campo tipo float.
    - stars: rating ponderado del negocio, campo tipo float.
    - review_count: cantidad de reviews del negocio, campo tipo integer.
    - is_open: campo que indica si el negocio está abierto o cerrado, campo tipo integer [0-1].
    - attributes: características que posee el negocio como garaje, servicio de valet, servicio de domicilio propio, entre otros, campo tipo objeto.
    - categories: categorías en las cuales se encuentra el negocio, campo tipo objeto.
    - hours: horario del negocio, campo tipo objeto. Con valores para cada día de la semana
- **review.json:** contiene información de las reviews de los usuarios hacia un negocio en particular.
    - review_id: llave única del archivo con el cual se identifica el registro, campo tipo string.
    - user_id: llave foránea del archivo user.json, campo tipo string.
    - business_id: Llave foránea del Archivo business, Json.
    - stars: número de estrellas de la review, campo tipo integer.
    - date: fecha de la review, campo tipo string.

    - text: texto de la review, campo tipo string.
    - useful: número de votos útiles recibidos, campo tipo integer.
    - funny: número de votos chistosos recibidos, campo tipo integer,
    - cool: número de votos geniales recibidos, campo tipo integer.
- **user.json:** información de los usuarios.
    - user_id: llave primaria del archivo de usuarios, identificador único de cada usuario, campo tipo string.
    - name: primer nombre del usuario, campo tipo string.
    - review_count: cantidad de reviews que el usuario a escrito, campo tipo integer.
    - yelping_since: fecha exacta de cuando el usuario se unió a Yelp, campo tipo string.
    - friends: amigos vinculados a la cuenta del usuario, campo tipo objeto.
    - useful: número de votos útiles enviados por el usuario, campo tipo integer.
    - funny: número de votos chistosos enviados por el usuario, campo tipo integer.
    - cool: número de votos geniales enviados por el usuario, campo tipo integer.
    - fans: número de fans que el usuario tiene, campo tipo integer.
    - Elite: cantidad de años que el usuario ha sido elite, campo tipo objeto.
    - average_starts: número promedio de estrellas de las reviews dadas, campo tipo float.
    - Compliments: Distintos elogios dados por un usuario a otro sobre algunas de sus reseñas. Estos pueden ser hot, more, profile, cute, entre otros.
- **checkin.json:** checkins realizados en un negocio.
    - business_id: Llave foránea del Archivo business, Json.
    - date: lista de cada uno de los checkins realizados por un usuario, campo tipo string.
- **tip.json:** Tips que el usuario brinda sobre un negocio, posee comentarios cortos.
    - text: texto del tip, campo tipo string.
    - date: fecha de publicación del tip, campo tipo string.
    - compliment_count: cantidad de cumplidos que el tip posee, campo tipo integer.
    - business_id: Llave foránea del Archivo business, Json.
    - user_id: llave foránea del archivo user.json, campo tipo string.
- **Photo.json:** contiene fotos del negocio.
    - Photo_id: llave primaria del archivo, identificador único de cada foto, campo tipo string.
    - business_id: Llave foránea del Archivo business, Json.
    - caption: rotulo de la foto, campo tipo string.
    - label: categoría de la foto, campo tipo string.

La lectura de los datos en formato JSON tomaba una gran cantidad de tiempo inclusive haciendo uso de herramientas de Big Data como Spark. Esto se debe principalmente a que el JSON está arreglado para mostrar un elemento por línea (Figura 2). Por tal razón los datos debieron ser convertidos al formato Parquet para obtener un mejor manejo de los tiempos por medio del uso de Spark.

Figura 2. Previsualización de conjunto de datos de JSON

3 Definición y construcción de un modelo híbrido de recomendación

3.1 Objetivo de recomendación

El objetivo de recomendación es definido según el modelo de Jannach con un objetivo general por parte del usuario y del cliente de la siguiente manera:

Objetivo General:

- Usuario: El usuario busca una interacción con el sistema. De esta manera, busca interactuar con listas de ítems interesantes y encontrar todos los ítems que sean relevantes a sus gustos. De esta manera, el propósito de la recomendación para el usuario será que estos encuentren opciones según sus gustos y localización y los atributos del negocio
- Proveedor: El sistema de recomendación de Yelp busca conectar negocios locales con sus clientes, para de esta manera incrementar la exposición del negocio y retener como clientes a los negocios locales. El propósito de recomendación para los proveedores o el negocio es visibilizar más su negocio. Para Yelp el propósito es visibilizar la plataforma.

Para cumplir este objetivo se tiene que la tarea del sistema es crear una lista de recomendación variada de negocio interesantes para el usuario según las características y gustos que este haya demostrado anteriormente. Se buscará encontrar opciones con una predicción de rating elevada.

La métrica de evaluación que se usará es RMSE para encontrar los mejores modelos y así mismo los mejores ítems para recomendarle al usuario según sus gustos.

3.2 Arquitectura del sistema de recomendación

La arquitectura definida para el modelo de recomendación se puede encontrar en la siguiente imagen:

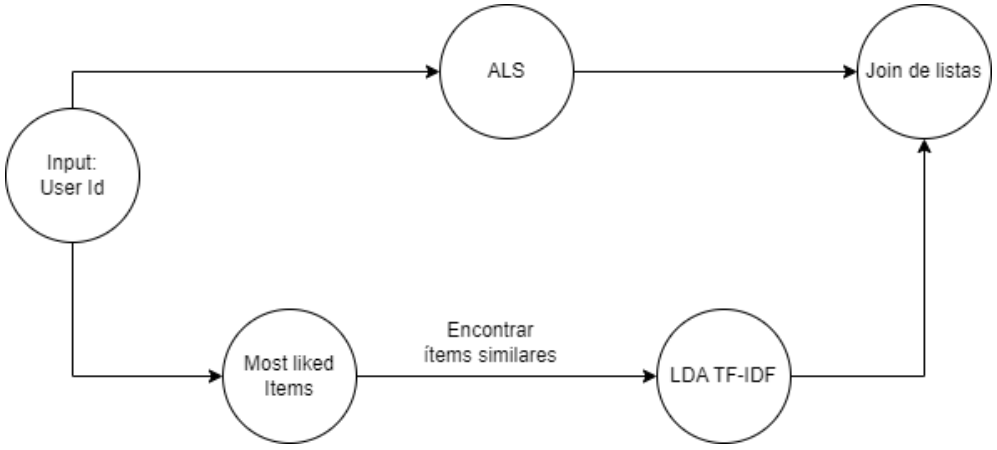


Figura 3. Arquitectura del Sistema de Recomendación Híbrido

La entrada se tiene como el id del usuario una vez se obtienen los negocios con los cuales el ha tenido las mejores interacciones.

- **Modelo basado en contenido:** Se obtienen las características de los ítems con los cuales el usuario ha tenido mejor interacción haciendo uso de LDA con TF-IDF. De esta manera se tienen las categorías con las que el usuario tiene mayor afinidad. LDA que es una forma de modelar temas estadísticos, de esta manera con esta metodología cada documento es representado por una mezcla de temas y cada tema es un conjunto de palabras.
- **Modelo ALS:** El modelo ALS que hace uso de la matriz de factorización para reducir la dimensionalidad de los datos. Al reducir la dimensionalidad se pueden obtener predicciones

de manera. Este modelo retorna una lista de negocios que el usuario no ha visitado con una calificación predicha. Este algoritmo funciona haciendo una reducción de 2 vectores con características latentes de usuario e ítem.

- **Unión de recomendaciones:** Para la unión de listas se toman todos los ítems del modelo ALS que tienen las categorías encontradas por el modelo basado en contenido. Si no se obtienen suficientes resultados de la unión de listas para él top N de recomendación se tomarán las recomendaciones restantes del modelo colaborativo.

4 Sintonización y evaluación del modelo

4.1 Modelo Filtrado Colaborativo por Factorización

4.1.1 Construcción

Este modelo se creó mediante ALS (Alternating Least Square) el cual es un filtro colaborativo de matriz de factorización, la cual descompone la matriz de usuario-ítem en matrices rectangulares de menor dimensión, en este modelo se puede representar los usuarios como filas y los factores latentes como columnas y la otra matriz descompuesta será los ítems como filas y las columnas como los factores latentes de los ítems. Este modelo permite predecir las recomendaciones de una manera más personalizada gracias a la descomposición de los factores latentes, así los ítems poco conocidos tendrán una buena representación por sus características, así como los ítems más conocidos permitiendo que se cree serendipia al recomendar.

El modelo ALS aplicado al dataset de YELP se configuro para correr con los siguientes parámetros en una primera iteración del modelo:

als = ALS(maxIter=5, implicitPrefs=False,nonnegative=True,userCol="user_id_numeric", itemCol="business_id_numeric", ratingCol="stars",coldStartStrategy="drop")

4.1.2 Evaluación

Los resultados de la primera evaluación fueron:

business_id_numeric	stars	user_id_numeric	prediction
134989	5.0	53	4.784931
146619	5.0	53	5.129341
45969	4.0	78	0.6580459
59279	4.0	103	4.3934727
147082	5.0	108	1.1513397
140514	5.0	137	3.8620214
136568	4.0	192	2.956036
44930	4.0	253	3.5839498
46363	5.0	253	3.3202004
106598	4.0	253	4.7191906
63989	5.0	271	3.7924955
60514	5.0	296	2.9828622
60999	5.0	296	2.4339318
23992	5.0	321	4.164449
9907	3.0	385	2.1610692
22265	5.0	412	3.5298643
50655	4.0	412	3.7664394
63411	5.0	412	4.4630632
74743	5.0	412	4.058633
116241	3.0	412	3.3426132

Imagen 2. Predicción primera iteración.

Como se puede observar en la imagen 2 la columna “prediction” muestra el número de estrellas de cada negocio. En estas predicciones es posible notar que algunas salen del rango esperado de 1-5. Por tal

razón, fue necesario hacer uso de un MinMax Scaler con el rango anterior. Con los ratings escalados se obtiene el siguiente resultado:

business_id_numeric	stars	user_id_numeric	prediction	prediction_Scaled
99905	5.0	27	2.1880224	1.749
134989	5.0	53	4.6566916	2.594
56535	1.0	78	2.6789527	1.917
99374	1.0	126	0.94163007	1.322
36105	5.0	159	1.9328552	1.662
53722	5.0	159	1.394835	1.478
39921	2.0	236	1.5677186	1.537
73020	3.0	296	3.1335728	2.073
706	5.0	321	4.0231266	2.377
1329	1.0	385	2.8808742	1.986
81619	5.0	406	1.9107007	1.654
11195	5.0	412	3.7310612	2.277
13874	5.0	412	3.2482476	2.112
40361	5.0	412	4.5415535	2.555
121654	2.0	412	3.742609	2.281
135477	5.0	412	3.40708	2.166
85462	5.0	436	4.566654	2.563
143427	2.0	460	2.0873175	1.715
81152	4.0	471	3.5137677	2.203
114767	4.0	471	4.6968937	2.608

Figura 4. Predicciones escaladas para ajustarse al rango de 1 a 5

A continuación, se muestran las métricas de evaluación del modelo.

```
[ ]  from pyspark.ml.evaluation import RegressionEvaluator

     evaluator = RegressionEvaluator(metricName="rmse", labelCol="stars",
                                     predictionCol="prediction")
     rmse = evaluator.evaluate(predictions)

     evaluator = RegressionEvaluator(metricName="r2", labelCol="stars",
                                     predictionCol="prediction")
     r2 = evaluator.evaluate(predictions)

     evaluator = RegressionEvaluator(metricName="mse", labelCol="stars",
                                     predictionCol="prediction")
     mse = evaluator.evaluate(predictions)


     print(rmse)
     print(mse)
     print(r2)
```

```
1.5432466395747944
2.381610190558895
-0.19882590791378085
```

Imagen 3. métricas evaluación primera iteración modelo asl.

De los valores mostrados en la imagen 3 se aprecia que los valores de RMSE, MSE y R2 indican que tan precisas son las predicciones y que tan certeras son con los valores originales, menores valores de RMSE indican un mejor ajuste y esta medida se utiliza para cuantificar la precisión, el MSE es la métrica más usada en los modelos supervisados y en regresiones e indica el resultado correcto para cada dato histórico y finalmente el R2 explica cuanta variabilidad de un factor puede ser causado por su relación con otro factor relacionado.

Realizando una búsqueda de hiper parámetros, variando la cantidad de iteraciones se encontró que el mejor resultado es dejando los parámetros de default nonegatives= True e Implicit False con 15 factores latentes y una regularización de 0.3

```
# hyper-param config
num_iterations = 15
ranks = [10,12,15]
reg_params = [0.1,0.2,0.01,0.15,0.3]

# grid search and select best model
start_time = time.time()
final_model = train_ALS(train_df, test_df, num_iterations, reg_params, ranks)

print ('Total Runtime: {:.2f} seconds'.format(time.time() - start_time))
```

```
10 latent factors and regularization = 0.1: validation RMSE is 2.205096938445347
10 latent factors and regularization = 0.2: validation RMSE is 1.9917038394270494
10 latent factors and regularization = 0.01: validation RMSE is 3.518367515215485
10 latent factors and regularization = 0.15: validation RMSE is 2.0680185091615213
10 latent factors and regularization = 0.3: validation RMSE is 1.9294583203459708
12 latent factors and regularization = 0.1: validation RMSE is 2.1913540852647464
12 latent factors and regularization = 0.2: validation RMSE is 1.9888842839640448
12 latent factors and regularization = 0.01: validation RMSE is 3.4082580408187284
12 latent factors and regularization = 0.15: validation RMSE is 2.0625524506687833
12 latent factors and regularization = 0.3: validation RMSE is 1.9278572059611998
15 latent factors and regularization = 0.1: validation RMSE is 2.1753425679812226
15 latent factors and regularization = 0.2: validation RMSE is 1.9855017146940772
15 latent factors and regularization = 0.01: validation RMSE is 3.242918668927559
15 latent factors and regularization = 0.15: validation RMSE is 2.0556866296285543
15 latent factors and regularization = 0.3: validation RMSE is 1.9276490066134366

The best model has 15 latent factors and regularization = 0.3
Total Runtime: 2402.24 seconds
```

```
# hyper-param config
num_iterations = 20
ranks = [10,12,15]
reg_params = [0.1,0.2,0.01,0.15,0.3]

# grid search and select best model
start_time = time.time()
final_model = train_ALS(train_df, test_df, num_iterations, reg_params, ranks)

print ('Total Runtime: {:.2f} seconds'.format(time.time() - start_time))
```

```
10 latent factors and regularization = 0.1: validation RMSE is 2.1817911638725316
10 latent factors and regularization = 0.2: validation RMSE is 1.9870114293571355
10 latent factors and regularization = 0.01: validation RMSE is 3.271182742360547
10 latent factors and regularization = 0.15: validation RMSE is 2.056243778203823
10 latent factors and regularization = 0.3: validation RMSE is 1.9307809997008545
12 latent factors and regularization = 0.1: validation RMSE is 2.1688354940595027
12 latent factors and regularization = 0.2: validation RMSE is 1.9832647467286304
12 latent factors and regularization = 0.01: validation RMSE is 3.1723252907184367
12 latent factors and regularization = 0.15: validation RMSE is 2.0498283610823465
12 latent factors and regularization = 0.3: validation RMSE is 1.9286660606362702
15 latent factors and regularization = 0.1: validation RMSE is 2.1543379501098836
15 latent factors and regularization = 0.2: validation RMSE is 1.9794591795082017
15 latent factors and regularization = 0.01: validation RMSE is 3.0261530410630115
15 latent factors and regularization = 0.15: validation RMSE is 2.0432338813640882
15 latent factors and regularization = 0.3: validation RMSE is 1.9281492202095938

The best model has 15 latent factors and regularization = 0.3
Total Runtime: 3121.50 seconds
```

Figura 5. Búsqueda de Hiper-paramétros

4.2 Modelo Filtrado Contenido

4.2.1 Construcción

Este modelo se realiza por medio de la matriz tf-idf el cual es un modelo basado en contenido donde analiza la cantidad de veces que aparecen los ítems en el corpus establecido. Adicionalmente, se hace uso de LDA con PySpark.

Varios algoritmos fueron probados antes de seleccionar el mencionado anteriormente. Algunos de los algoritmos que se evaluaron fueron:

- TF-IDF/KMeans: Métricas muy bajas a pesar de probar con distintos parámetros de K
- TF-IDF/KBISECKMeans: Comportamiento similar a KMeans.
- TF-IDF/Similitud Coseno: Poco eficiente para obtener similitud

```
2 clusters and Silhouette with squared euclidean distance = 0.1804815344086881
3 clusters and Silhouette with squared euclidean distance = 0.15823452932701282
4 clusters and Silhouette with squared euclidean distance = 0.1928495493519425
5 clusters and Silhouette with squared euclidean distance = 0.11152592026791004
6 clusters and Silhouette with squared euclidean distance = 0.10425933109976084
7 clusters and Silhouette with squared euclidean distance = 0.10648503046037486
8 clusters and Silhouette with squared euclidean distance = 0.11080322750831334
9 clusters and Silhouette with squared euclidean distance = 0.11148030868898186
10 clusters and Silhouette with squared euclidean distance = 0.11047653253396048
11 clusters and Silhouette with squared euclidean distance = 0.11345062726434875
12 clusters and Silhouette with squared euclidean distance = 0.09563779755629293
13 clusters and Silhouette with squared euclidean distance = 0.09890642688657138
14 clusters and Silhouette with squared euclidean distance = 0.08113685741416299
15 clusters and Silhouette with squared euclidean distance = 0.08406711452746392
16 clusters and Silhouette with squared euclidean distance = 0.0925298540197655
17 clusters and Silhouette with squared euclidean distance = 0.09355905846734784
18 clusters and Silhouette with squared euclidean distance = 0.07422800901367006
19 clusters and Silhouette with squared euclidean distance = 0.0714419735096137
20 clusters and Silhouette with squared euclidean distance = 0.07062714206174131
21 clusters and Silhouette with squared euclidean distance = 0.07538988956194426
22 clusters and Silhouette with squared euclidean distance = 0.07715598323274232
23 clusters and Silhouette with squared euclidean distance = 0.07708420727394781
24 clusters and Silhouette with squared euclidean distance = 0.08796306595058598
25 clusters and Silhouette with squared euclidean distance = 0.07956790023616918
26 clusters and Silhouette with squared euclidean distance = 0.07063842445912646
27 clusters and Silhouette with squared euclidean distance = 0.0700177938832984
28 clusters and Silhouette with squared euclidean distance = 0.06503006891377443
29 clusters and Silhouette with squared euclidean distance = 0.06655715623036719
30 clusters and Silhouette with squared euclidean distance = 0.06893248678697995
31 clusters and Silhouette with squared euclidean distance = 0.06964476281578053
32 clusters and Silhouette with squared euclidean distance = 0.07065628866473754
33 clusters and Silhouette with squared euclidean distance = 0.05944845815658811
```

Figura 6. Métricas de KMeans con TF-IDF. Métrica varia de 1 a -1

Finalmente, se decidió trabajar con ALD debido a que, a pesar de su demora para sacar los resultados, las métricas obtenidas son muy útiles debido a que permiten hacer un análisis visual por grupo.

Se utilizo un procesamiento de lenguaje natural (NLP), con el cual se quitaron las stopwords del idioma inglés y se quitaron caracteres especiales. Adicionalmente, se realizaron técnicas de trimming y normalización. Posteriormente se tokenizo la columna objetivo la cual es la unión de las columnas “state”, “city”, “name” y “categories”.

El modelo busca categorizar los negocios por palabras claves identificadas en la columna objetivo, básicamente es realizar un proceso de NLP y luego aplicar un modelo de clúster para identificar grupos.

4.2.2 Evaluación

La evaluación y los resultados del modelo se encuentran a continuación, donde para cada business id se logra asociar una categoría a la que pertenece gracias al contexto que le da la columna objetivo.

```
predictions_kmeans = model.transform(rescaledData)
predictions_kmeans.show()
```

business_id_numeric	business_id	city	state	name	stars	categories	concat_col	words	rawFeatures	features	prediction
1	xKoz9eM8hUELf5qix...	Plainfield	IN	African Plum Home...	4.5	Home & Garden, Ho...	IN Plainfield Afr...	[in, plainfield, ...]	(20,[3,5,6,9,12,1...]	(20,[3,5,6,9,12,1...]	1
3	7d9X9wnm_35UcdJRJ...	Indianapolis	IN	China King	4.0	Restaurants, Chinese	IN Indianapolis C...	[in, indianapolis...]	(20,[0,3,14,19],[...	(20,[0,3,14,19],[...	1
4	CFPWVAFSP3Ktfe-cE...	Philadelphia	PA	Liberty Real Esta...	1.5	Home Services, Re...	PA Philadelphia L...	[pa, philadelphia...	(20,[5,11,12,16,1...	(20,[5,11,12,16,1...	0
5	EgZH_qXEQH_WJyNFg...	Nashville	TN	Jamie	3.0	Shopping, Accesso...	TN Nashville Jami...	[tn, nashville, j...	(20,[2,6,8,13,17,...	(20,[2,6,8,13,17,...	1
6	b3KxHgc7le1Mc2Q3O...	Edmonton	AB	Flowers By Merle	3.5	Flowers & Gifts, ...	AB Edmonton Flowe...	[ab, edmonton, fl...	(20,[1,2,3,6,7,12...	(20,[1,2,3,6,7,12...	1
7	2tAi411srrpmMofeA...	Tucson	AZ	Just Breakfast on...	3.5	Coffee & Tea, Bre...	AZ Tucson Just Br...	[az, tucson, just...	(20,[0,4,7,9,12,1...	(20,[0,4,7,9,12,1...	1
8	dN2whsRNU90vPfocj...	Philadelphia	PA	Baltimore Pet Shoppe	4.5	Pet Stores, Pets	PA Philadelphia B...	[pa, philadelphia...	(20,[3,4,8,17,19]...	(20,[3,4,8,17,19]...	1
9	exWa73g8bG0p3pI7N...	Edmonton	AB	Careit Urban Deli	4.0	Restaurants, Deli...	AB Edmonton Carei...	[ab, edmonton, ca...	(20,[0,7,13,15,18...	(20,[0,7,13,15,18...	1
10	kdZr-IdMNOUNZxbZD...	Tucson	AZ	US Post Office	1.5	Public Services &...	AZ Tucson US Post...	[az, tucson, us, ...	(20,[0,4,8,10,12,...	(20,[0,4,8,10,12,...	1
12	vhe1SzsLE2rG6a9_O...	Nashville	TN	Dashwood Vintage ...	5.0	Home Decor, Home ...	TN Nashville Dash...	[tn, nashville, d...	(20,[2,3,5,6,11,1...	(20,[2,3,5,6,11,1...	0
13	92qmBM9q0Tozdou9Y...	Dunedin	FL	Baskin-Robbins	1.5	Ice Cream & Froze...	FL Dunedin Baskin...	[fl, dunedin, bas...	(20,[0,2,8,12,13,...	(20,[0,2,8,12,13,...	1
14	DfW0q_DJeBj5h4DM5...	Bensalem	PA	Everlasting Nails...	3.5	Beauty & Spas, Na...	PA Bensalem Everl...	[pa, bensalem, ev...	(20,[0,3,5,7,8,12...	(20,[0,3,5,7,8,12...	1
16	qE6PBhgB7AFtUHTp0...	Tucson	AZ	Brichta Infant an...	4.5	Local Services, P...	AZ Tucson Brichta...	[az, tucson, bric...	(20,[0,3,4,8,10,1...	(20,[0,3,4,8,10,1...	0
17	BnffoBFNuGmNAksel...	Philadelphia	PA	Cavanaugh's Resta...	3.0	Burgers, Caterers...	PA Philadelphia C...	[pa, philadelphia...	(20,[0,5,8,9,10,1...	(20,[0,5,8,9,10,1...	1
18	nKcxdo0ELwBj99vPt...	Metairie	LA	Pure Fitness	5.0	Boot Camps, Nutri...	LA Metairie Pure ...	[la, metairie, pu...	(20,[4,5,8,11,12,...	(20,[4,5,8,11,12,...	0
19	bbWRELNpKXXnFgQMk...	Philadelphia	PA	Trolley Car Station	3.0	Restaurants, Amer...	PA Philadelphia T...	[pa, philadelphia...	(20,[0,1,12,14,15...	(20,[0,1,12,14,15...	1
20	JkQz12i_Eubd_xQdL...	Tucson	AZ	HUB Ice Cream Fac...	4.0	Restaurants, Food...	AZ Tucson HUB Ice...	[az, tucson, hub,...	(20,[0,2,4,5,12,1...	(20,[0,2,4,5,12,1...	1
23	f8TyUEoSWQEgH257G...	Boise	ID	Bob's Sunrise Cafe	3.0	Restaurants, Amer...	ID Boise Bob's Su...	[id, boise, bob's...	(20,[0,12,13,14,1...	(20,[0,12,13,14,1...	1
25	ay5arm6V0XwWkwjoN...	Boise	ID	Alteration Excell...	4.0	Local Services, S...	ID Boise Alterati...	[id, boise, alter...	(20,[7,8,12,14,16...	(20,[7,8,12,14,16...	1
26	U3s0z5VAYvOlr4fCS...	Philadelphia	PA	Cycle Brewerytown	5.0	Gyms, Yoga, Activ...	PA Philadelphia C...	[pa, philadelphia...	(20,[7,10,11,12,1...	(20,[7,10,11,12,1...	0

only showing top 20 rows

Imagen 4. resultado predicción modelo 2.

Por medio de este modelo se identifican los grupos de ítems que estén en la misma categoría para el usuario que se está evaluando, por medio del contexto del ítem se relacionan otros ítems que posean similaridades.

topic	termIndices	termWeights	terms
0	[21, 25, 18]	[0.03922667448845111, 0.031328102335000836, 0.02530107309867177]	[auto, automotive, repair]
1	[34, 92, 1]	[0.050719903452965544, 0.020672365476454256, 0.011140880216406888]	[pet, pets, services]
2	[6, 7, 24]	[0.022393045729122197, 0.020049442803876043, 0.019298476318872472]	[home, shopping, stores]
3	[8, 19, 3]	[0.028920977886311816, 0.015494291992452327, 0.015159215781762518]	[bars, nightlife, food]
4	[74, 13, 15]	[0.034904789892728175, 0.027483961773703488, 0.024568820833126018]	[dentists, medical, health]
5	[33, 14, 23]	[0.029718151391274934, 0.029321064156444695, 0.023320544514171287]	[hotels, event, planning]
6	[17, 9, 10]	[0.03924273013603368, 0.03369185570176637, 0.0323644605179386]	[hair, spas, beauty]
7	[6, 1, 13]	[0.021357749715284115, 0.019275374546913134, 0.00993900040820934]	[home, services, medical]
8	[3, 2, 26]	[0.025703199001044306, 0.021063100489433068, 0.017711662940884002]	[food, restaurants, pizza]
9	[144, 18, 1]	[0.024372107056404024, 0.021000094202662294, 0.017116740183217804]	[mobile, repair, services]

Imagen 5. Categorías identificadas modelo 2.

Como se puede apreciar en la imagen 6 el modelo identifica palabras claves del texto para realizar una recomendación a los ítems, así se logra vincular los ítems para poder realizar una recomendación basada en tópicos.

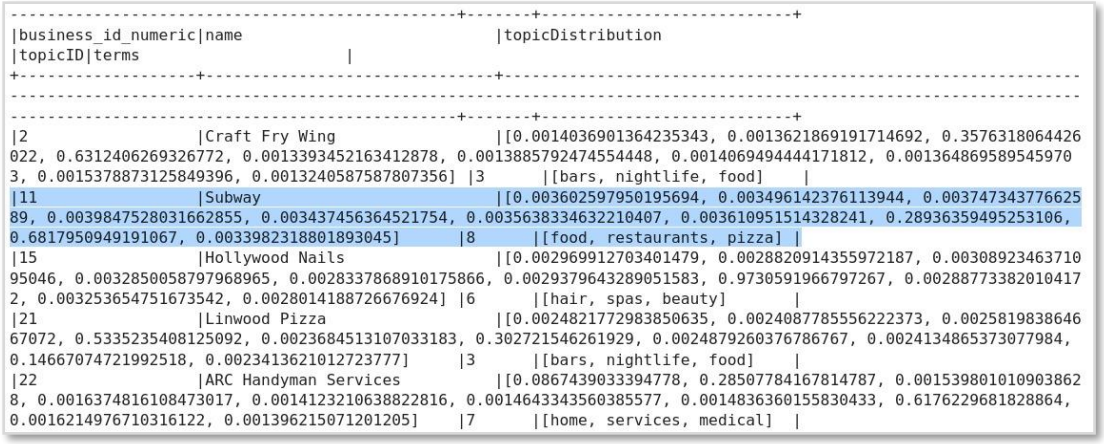
Un ejemplo de lo anteriormente mencionado se puede apreciar a continuación donde se vincula un negocio con su categoría a recomendar.

```
The topics described by their top-weighted terms:
+------------------+------------------------------+-------+--------------------------+
|business_id_numeric|name                          |topicID|terms                     |
+------------------+------------------------------+-------+--------------------------+
|2                 |Craft Fry Wing                |6      |[laundry, hair, dry]      |
|11                |Subway                        |1      |[pet, pets, services]     |
|15                |Hollywood Nails               |3      |[bars, spas, hair]        |
|21                |Linwood Pizza                 |2      |[shopping, home, stores]  |
|22                |ARC Handyman Services         |7      |[home, services, medical] |
|24                |Autozone                      |0      |[auto, automotive, repair]|
|29                |Noble Roman's Craft Pizza & Pub|0      |[auto, automotive, repair]|
|32                |Sacred Art Tattoo Studio      |9      |[tattoo, barbeque, arts]  |
|34                |Jose's Mexican Restaurant     |9      |[tattoo, barbeque, arts]  |
|40                |Johnny Carino's               |3      |[bars, spas, hair]        |
|49                |Mandarin Inn                  |5      |[event, hotels, planning] |
|52                |PetSmart                      |1      |[pet, pets, services]     |
|54                |ServiceMaster of Bux Mont     |7      |[home, services, medical] |
|55                |Lemus Construction            |7      |[home, services, medical] |
|66                |Spring Garden Restaurant      |8      |[food, restaurants, coffee]|
|71                |Paint Nail Bar -  St Pete     |3      |[bars, spas, hair]        |
|78                |Brownsburg Bowl               |5      |[event, hotels, planning] |
|81                |Bluegrass OB/GYN              |7      |[home, services, medical] |
|86                |Sandra's German Restaurant    |9      |[tattoo, barbeque, arts]  |
|94                |Woody's Detail                |4      |[mobile, repair, insurance]|
+------------------+------------------------------+-------+--------------------------+
only showing top 20 rows
```

Imagen 6. Resultados modelo 2, categorías identificadas.

En la imagen anterior se puede apreciar cómo se identifica que la categoría del negocio Hollywood Nails es [hair, spas, beauty] por ende el modelo de recomendación buscara elementos que perteneces a esas categorías.

El modelo 2 calcula los pesos de cada categoría para y realiza un ponderado para identificar a que categoría pertenece.



```
-----------------------------------------------+-------+--------------------------+
|business_id_numeric|name                        |topicDistribution
|topicID|terms                    |
+------------------+----------------------------+----------------------------------------------------------------
----------------------------------------------------------------------------------------------------------------
-----------------------------------------------+-------+--------------------------+
|2                 |Craft Fry Wing              |[0.0014036901364235343, 0.0013621869191714692, 0.3576318064426
022, 0.6312406269326772, 0.0013393452163412878, 0.0013885792474554448, 0.0014069494444171812, 0.0013648695895459970
3, 0.0015378873125849396, 0.0013240587587807356] |3      |[bars, nightlife, food]    |
|11                |Subway                      |[0.003602597950195694, 0.003496142376113944, 0.003747343776625
89, 0.0039847528031662855, 0.003437456364521754, 0.0035638334632210407, 0.003610951514328241, 0.28936359495253106,
0.6817950949191067, 0.003398231880189304 5]      |8      |[food, restaurants, pizza] |
|15                |Hollywood Nails             |[0.0029699127034014 79, 0.0028820914355972187, 0.00308923463710
95046, 0.0032850058797968965, 0.0028337868910175866, 0.0029379643289051583, 0.9730591966797267, 0.00288773382010417
2, 0.0032536547516735 42, 0.0028014188726676924] |6      |[hair, spas, beauty]       |
|21                |Linwood Pizza               |[0.0024821772983850635, 0.0024087785556222373, 0.0025819838646
67072, 0.5335235408125092, 0.0023684513107033183, 0.302721546261929, 0.0024879260376786767, 0.0024134865373077984,
0.14667074721992518, 0.0023413621012723777]     |3      |[bars, nightlife, food]    |
|22                |ARC Handyman Services       |[0.0867439033394778, 0.28507784167814787, 0.001539801010903862
8, 0.0016374816108473017, 0.0014123210638822816, 0.0014643343560385577, 0.0014836360155830433, 0.6176229681828864,
0.0016214976710316122, 0.001396215071201205]    |7      |[home, services, medical]  |
```

Imagen 7. Distribución de los tópicos, pesos de las categorías.

La búsqueda del mejor k para el modelo 2 muestra a continuación, anteriormente se describe que el proceso para la categorización de ítems es: Trimming (separar palabras o frases), Normalización (limpieza de puntos, comas, entre otros), Tokenizacion (identificación de palabras claves), Conteo de palabras TF-IDF.

```
2 clusters and Silhouette with squared euclidean distance = 0.1804815344086881
3 clusters and Silhouette with squared euclidean distance = 0.15823452932701282
4 clusters and Silhouette with squared euclidean distance = 0.1928495493519425
5 clusters and Silhouette with squared euclidean distance = 0.11152592026791004
6 clusters and Silhouette with squared euclidean distance = 0.10425933109976084
7 clusters and Silhouette with squared euclidean distance = 0.10648503046037486
8 clusters and Silhouette with squared euclidean distance = 0.11080322750831334
9 clusters and Silhouette with squared euclidean distance = 0.11148030868898186
10 clusters and Silhouette with squared euclidean distance = 0.11047653253396048
11 clusters and Silhouette with squared euclidean distance = 0.11345062726434875
12 clusters and Silhouette with squared euclidean distance = 0.09563779755629293
13 clusters and Silhouette with squared euclidean distance = 0.09890642688657138
14 clusters and Silhouette with squared euclidean distance = 0.08113685741416299
15 clusters and Silhouette with squared euclidean distance = 0.08406711452746392
16 clusters and Silhouette with squared euclidean distance = 0.0925298540197655
17 clusters and Silhouette with squared euclidean distance = 0.09355905846734784
18 clusters and Silhouette with squared euclidean distance = 0.07422800901367006
19 clusters and Silhouette with squared euclidean distance = 0.0714419735096137
20 clusters and Silhouette with squared euclidean distance = 0.07062714206174131
21 clusters and Silhouette with squared euclidean distance = 0.07538988956194426
```

Imagen 8. Búsqueda de hiper-paramétro, cantidad de clústeres.

Se elige el k=2 ya que es el mayor y según la métrica del codo es el que tiene menor varianza con respecto al siguiente k.

4.3 Modelo hibrido

4.3.1 Construcción

El modelo hibrido se construye mediante los dos modelos anteriores utilizando un diseño secuencial y de ensamble para obtener una recomendación más robusta teniendo en cuenta las características de cada modelo involucrado en modelo hibrido.

La construcción se realiza a partir de la salida que se obtiene del modelo ALS que a partir de los factores latentes y la factorización de la matriz la salida son ratings de cada ítem y/o usuario, después de ahí pasa al modelo 2 donde a partir de los ítems que le gustan al usuario se clasifica en temáticas o clústeres de categorías, allí se elige una temática de manera aleatoria y se busca recomendar un nuevo negocio que no conozca el usuario y que pertenezca a la misma categoría.

Ejemplificando la explicación anterior seria evaluar los negocios que el usuario le gustan obtener un ranking a partir de todos los ítems y validar las categorías del top de negocios, allí se buscaría un nuevo negocio que el usuario no haya visitado y se valida que pertenezca a la categoría elegida por el modelo, de esta manera estamos recomendando un nuevo ítem al usuario a partir de los negocios que más le gustan y pertenecientes a la misma categoría.

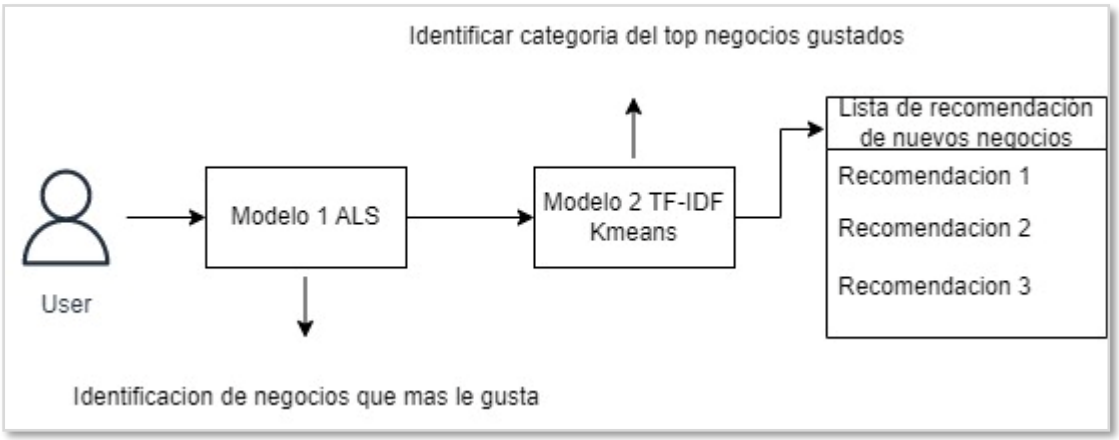


Imagen 9. Modelo hibrido arquitectura.

4.3.2 Evaluación

5 Conclusiones

- Se puede concluir que construir modelos híbridos de recomendación es bastante útil cuando se quiere combinar y ensamblar modelos de diferente naturaleza para lograr una recomendación más robusta en nuestro caso se realizó un ensamble en serie de un modelo colaborativo de factorización y un modelo basado en contenido.
- Poder realizar una descomposición de factores latentes es pertinente a la hora de no omitir ítems que no tienen mucha popularidad, ya que en otros modelos son omitidos ya que no poseen casi valoraciones o ratings.
- Poder identificar la categoría de un ítem permite filtrar más la búsqueda de negocios para una futura recomendación, plantear una estrategia de NLP es adecuado para tener en cuenta el contexto de cada ítem.
- Utilizar tecnologías para el procesamiento de grandes cantidades de datos es esencial para abarcar más cantidad de usuarios e ítems, y poder realizar un procesamiento rápido y robusto con altas cantidades de datos, Spark es una tecnología de big data que trabaja en memoria y permite procesar, transformar y manejar archivos de gran tamaño.

6 Bibliografía

[1] Prototyping a Recommender System Step by Step Part 2: Alternating Least Square (ALS) Matrix Factorization in Collaborative Filtering. Kevin Liao, Nov 17 de 2018

[2] Evaluando el error en los modelos de regresión. Ligdi Gonzalez, Nov 23 del 2018.

[3] Recommender Systems, Springer Link, Charu C. Aggarwal, March 29 de 2016. (Capitulo Ensemble-Based and Hybrid Recommender System).