# Recomendador Turistico M.L

Jhoan S. Moscoso, Juan M. Agudelo & Simon Garcia Lujan
Email: jmanuel.agudelo1@udea.edu.co & jhoan.moscoso@udead.edu.co
Instituto de Física, Universidad de Antioquia U de A, calle 70 No. 52-21, Medellín, Colombia

***—Abstract*** In this article, we present the development of a tourism recommendation system based on machine learning techniques, which integrates linear models and dense neural networks to predict the rating a user would assign to a tourist destination. Based on a rigorous analysis and cleansing of the Yelp dataset, personalized profiles were created for both users and establishments, combining metrics such as the adjusted rating, review trust, TF-IDF, and social influence. The results obtained demonstrate that despite the complexity and high dimensionality of the data, both approaches achieve comparable performance, with a mean absolute error (MAE) close to 1 star, underscoring the viability of these techniques in enhancing the quality and personalization of recommendations.

***—Resumen*** En este artículo presentamos el desarrollo de un sistema de recomendación turística basado en técnicas de machine learning, que integra modelos lineales y redes neuronales densas para predecir la puntuación que un usuario asignaría a un destino turístico. A partir de un riguroso análisis y depuración del conjunto de datos de Yelp, se construyeron perfiles personalizados tanto para usuarios como para establecimientos, combinando métricas como el rating ajustado, la confianza en las reseñas, el TF-IDF y la influencia social. Los resultados obtenidos demuestran que, pese a la complejidad y alta dimensionalidad de los datos, ambos enfoques alcanzan un desempeño comparable, con un error medio absoluto (MAE) cercano a 1 estrella, lo que subraya la viabilidad de estas técnicas para mejorar la calidad y personalización de las recomendaciones.

## I. Introduccion

El turismo es reconocido desde hace mucho tiempo como una de las industrias más dinámicas y económicamente significativas del mundo. A medida que el numero de viajeros aumenta, el reto de conciliar las preferencias individuales con la diversa gama de atracciones disponibles es cada vez más complejo. En este articulo, abordamos el problema de predecir la puntuación que un turista asignaría a un lugar determinado, aprovechando modelos predictivos entrenados en el conjunto de datos de turismo de Yelp [1]. Nuestro enfoque construye un perfil de usuario utilizando una formula personalizada que asigna pesos a diferentes propiedades de un lugar, capturando sutilezas en el historial de reseñas, lugares comunes entre amigos y preferencias del usuario. Paralelamente generamos un perfil para cada lugar turístico empleando una técnica de frecuencia inversa de documentos *(IDF)* [2] aplicada a sus propiedades, lo que destaca de manera efectiva las características distintivas de cada lugar. Al introducir estos perfiles en un modelo predictivo, nuestro sistema aprende las relaciones subyacentes entre las preferencias de los usuarios, lo que le permite prever la puntuación que un usuario podría asignar a un destino en particular.

## II. metodología

### *II-A. Análisis y depuración del conjunto de datos*

Al explorar el conjunto de datos de Yelp, identificamos la necesidad de reducir su alta dimensionalidad, ya que contiene aproximadamente 7 millones de reseñas y 150,400 establecimientos con alrededor de 1,300 características distintas. Tras evaluar diversas estrategias de segmentación, optamos por filtrar las reseñas de usuarios que habían realizado más de 7 aportes, lo que redujo el volumen de datos en un 71.6 %, abarcando así 228,195 usuarios únicos. De manera similar, analizamos la frecuencia de las categorías de los establecimientos y eliminamos aquellas con menos de 15 incidencias, disminuyendo el número de categorías de 1,311 a 946. A pesar de esta depuración, el volumen de datos seguía siendo considerable. Para gestionar eficientemente esta información, decidimos almacenarla en archivos Parquet. Este formato permite una compresión mas efectiva y una velocidad de lectura mayor

### *II-B. **Creacion del Perfil de los usuarios***

Para construir un perfil personalizado y robusto para cada usuario, empleamos la siguiente fórmula:

$$W_{c,u} = AdjustedRating_{user}(c) \times Confidence_{user}(c) \times TF-IDF_{user}(c) + SocialBoost_{user}(c) \quad (1)$$

Donde:

1. **Rating Ajustado**

$$AdjustedRating_u(c) = \frac{\sum_i (r_{u,i} - \mu_u)}{Count_u(c)} + \mu_{global}$$

- $r_{u,i}$ : El rating de un usuario sobre los lugares $i$
- $\mu_u$ : El rating promedio de un usuario sobre todas las categorias
- $\mu_{global}$ : El promedio global

Este ajuste considera que una calificación de 4 estrellas de un usuario con promedio de 3.5 es más significativa que la misma calificación de un usuario con promedio de 4.5.

2. **Puntaje de confidencialidad**

$$Confidence_u(c) = \frac{Count_u(c)}{Count_u(c) + C}$$

- $C$ : Parámetro de ajuste establecido en 12, indicando que las reseñas de usuarios con más de 12 evaluaciones son más confiables.

3. **TF-IDF (especificidad categoría)**

$$TF - IDF_u(c) = TF_u(c) \times IDF(c)$$

- $TF_u(c)$ : Frecuencia de la categoría $c$ en las reseñas del usuario $u$.
- $IDF(c) = \log(\frac{TotalUsers}{Users\ who\ reviewed\ c})$

Este enfoque penaliza categorías comunes, como Restaurantes", para resaltar preferencias más específicas.

4. **SocialBoost (Influencia social)**

$$SocialBoost_u(C) = \alpha \times \frac{\sum_f W_f(c) \times Similarity(u, f)}{\sum_f Similarity(u, f)}$$

- $\alpha$ : Factor de escala, establecido en 0.3
- $Similarity(u, f)$ : Similitud coseno entre el usuario $u$ y sus amigos $f$ en términos de vectores de categorías.

Esta fórmula integra las calificaciones ajustadas, la confianza en las reseñas, la especificidad de las categorías y la influencia social para construir un perfil de usuario detallado y preciso.

### *II-C. Creación del Perfil de los Lugares*

Construimos de manera análoga un perfil para cada lugar empleando la siguiente formula:

$$W_{c,b} = TF - IDF(c) \times \left( \frac{\sum(UserRating \times UserConfidence) + \alpha\mu}{\sum UserConfidence + \alpha} \right) \times \log(1 + ReviewCount) \quad (2)$$

1. **TF-IDF (especificidad categoría)**

$$TF - IDF_b(c) = TF_b(c) \times IDF(c)$$

A diferencia del perfil de los lugares, el IDF lo calculamos en esta ocasión como:

$$IDF(c) = \log(\frac{TotalBusinesses}{Businesses\ with\ category\ c})$$

2. **Bayesian-Adjusted Rating (Credibilidad del puntaje del peso)**

$$AdjustedRating = \frac{\sum(UserRating \times UserConfidence)\alpha\mu}{\sum UserConfidence + \alpha}$$

- $\mu$ : Promedio del rating global de todos los lugares
- $\alpha$ : Parámetro de ajuste, establecido como 5

Esto se tiene en cuenta, puesto que los usuarios con mayor indice de confidencialidad (reviewers frecuentes) contribuye mucho mas al rating. Por otro lado, los negocios con menos reviews son empujados con el promedio global $\mu$ evitando los valores extremos.

3. **Escala de popularidad**

$$Popularity = \log(1 + ReviewCount)$$

Para compensar moderadamente la popularidad y no dejar que lugares con una gran cantidad de reviews dominen.

## III. Resultados

1. **Aproximación lineal: Stochastic Gradient Descent Regressor (SGDRegressor).**

Una de las principales preocupaciones que se tenían al intentar ajustar un modelo lineal para este proyecto era el hecho de que probablemente cualquier ajuste lineal de una cantidad alta de parámetros (1966) no fuera muy precisa. Lo anterior sumado al hecho de que nuestro *target* era un rango fijo de enteros de uno a cinco, produjo un error absoluto considerable: Para todas las configuraciones de hiperparámetros que se probaron, el MAE (*Mean Absolute Error*) se mantuvo alrededor de 1. Esta preocupación la podemos ver ejemplificada en las siguientes dos gráficas, en las cuales podemos notar como la curva de validación disminuye y gradualmente se estabiliza.

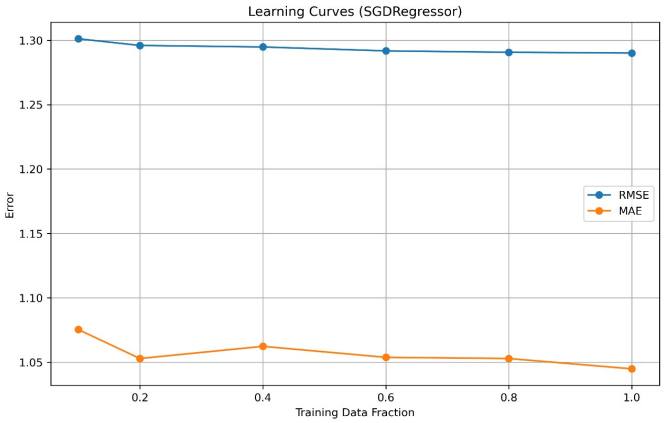


Figura 1. Curvas de aprendizaje para el SGD Regression con parámetros

- *Penalty: l2*
- *Alpha: 0.0001*
- *Max Iter: 1000*
- *Loss: squared error*
- *Learning rate: invscaling*

En esta configuración, la tasa de aprendizaje (Learning Rate) disminuye conforme avanza el entrenamiento, lo cual favorece la convergencia rápida y estable del modelo, sobre todo cuando se entrena con conjuntos de datos de menor tamaño. Las curvas de aprendizaje muestran una disminución constante en la función de pérdida, estabilizándose a medida que se incrementa el volumen de datos.

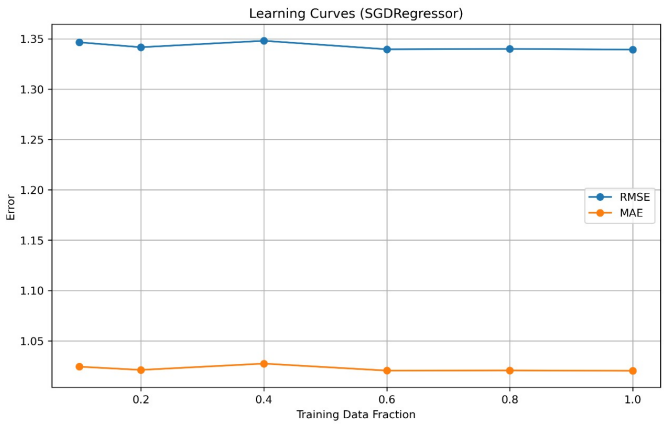


Figura 2. Curvas de aprendizaje para el SGD Regression con parámetros

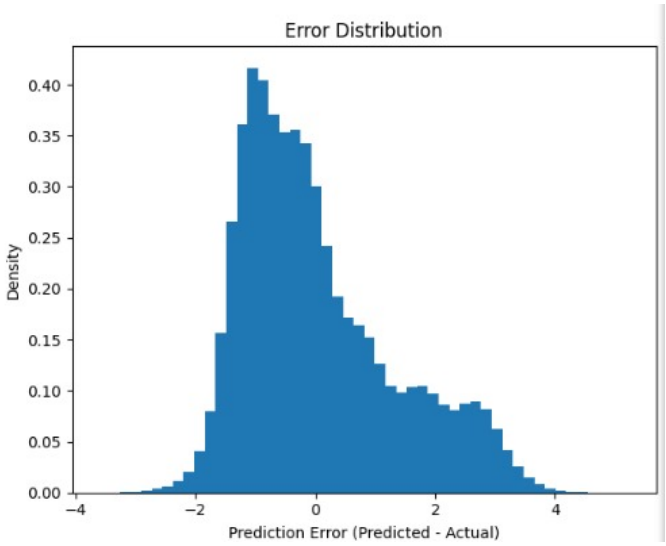


Figura 3. Distribucion del error de predicion (DNN)

- *Penalty: elasticnet*
- *Alpha: 0.001*
- *Max Iter: 1000*
- *Loss: Huber*
- *Learning rate: adaptative*
- *eta0: 0.01*

Al combinar penalizaciones L1 y L2, esta configuración logra eliminar características irrelevantes, reduciendo la varianza del modelo y mejorando su robustez frente a datos ruidosos y a la abundancia de ceros presentes en el conjunto. Ambas configuraciones arrojaron un error medio absoluto (MAE) menor a 1 estrella, lo cual es razonable dado que nuestro objetivo (target) corresponde a calificaciones enteras en el rango de 1 a 5.

2. **Aproximación mediante DNN: Dense Neuronal Network**
   Se desarrolló un modelo de red neuronal densa configurado de la siguiente manera:

- *Número de capas:* 5
- *Distribución de neuronas:* 512, 256, 128, 1
- *Funciones de activación:* [ReLU], lineal
- *Optimizador:* Adam
- *Épocas:* 5

Tras el entrenamiento, el modelo DNN alcanzó un MAE de aproximadamente 1.03 estrellas, similar en magnitud al obtenido con el modelo lineal. La Figura 3. muestra la distribución de los errores de predicción, definidos como la diferencia entre la calificación predicha y la calificación real.

En la gráfica se observa que la distribución de los errores presenta un ligero desplazamiento hacia la izquierda, dicho desplazamiento implica que, en promedio, el modelo DNN tiende a subestimar la calificación real; es decir, las predicciones son ligeramente inferiores a las calificaciones otorgadas por los usuarios.
Este sesgo leve puede deberse a diversos factores, entre los cuales se encuentran:

- La configuración del modelo o la función de pérdida utilizada, que podría favorecer una aproximación que minimice ciertos errores en detrimento de otros.
- La distribucion de las reviews positivas con respecto a las negativas en la fase de entrenamiento y limpieza de datos

Ambos enfoques – el modelo lineal (SGDRegressor) y la red neuronal densa – presentan desempeños comparables, con un MAE en torno a 1 estrella, lo cual conlleva a que la elección entre un lugar u otro en un conjunto de datos dependa de otros factores como la cercanía del usuario al lugar, el clima, la intención del usuario.

## IV. Conclusiones

La integración de técnicas de machine learning en sistemas de recomendación turística ha demostrado ser una herramienta poderosa para personalizar la experiencia del viajero y optimizar la planificación de sus desplazamientos. El enfoque propuesto, que combina la construcción de perfiles detallados de usuarios y establecimientos permite capturar las sutilezas de las preferencias individuales y las características distintivas de cada destino. Cabe destacar que, aunque el sistema presenta un margen de error que podría considerarse elevado en ciertos contextos, su fortaleza reside en la capacidad de sugerir un conjunto de destinos de interés que facilita la toma de decisiones por parte del usuario. El presente trabajo sienta las bases para el desarrollo de sistemas de recomendación turísticas

adaptativas, ademas de abrir la puerta a una implementación futura.

## V. Referencias

1. Yelp. *Yelp Open Dataset* Retrieved 7 de Febrero de 2025 https://www.yelp.com/dataset
2. Speech and Language Processing (3rd ed. draft), Dan Jurafsky and James H. Martin, chapter 14. https://web.stanford.edu/~jurafsky/slp3/14.pdf
3. Speech and Language Processing (3rd ed. draft), Dan Jurafsky and James H. Martin, chapter 14. https://web.stanford.edu/~jurafsky/slp3/14.pdf