

Universidad Internacional de La Rioja (UNIR)

Escuela Superior de Ingeniería y Tecnología

Máster en Análisis y Visualización de Datos Masivos

**Sistema de recomendaciones web pa-
ra selección de sitios de interés y lu-
gares turísticos de Ecuador basado
en las preferencias del usuario**

Trabajo Fin de Máster

presentado por: Sánchez Cevallos, María Gabriela

Director: Martí, Julio Marcelo

Ciudad: Quito

Fecha: 19 de septiembre de 2019

Gracias a mi familia, especialmente a mi madre que ha sido mi apoyo y ejemplo durante todo mi camino hasta llegar a este punto y quien es mi inspiración para ser mejor cada día.

Tambien quiero agradecer a mis profesores, compañeros de trabajo y amigos que separaron un tiempo para darme apoyo y colaboración.

Abstract

The main objective of this report is to show the knowledge acquired through the development of a recommendation system using a web page that allows us to see the recommendations of tourist places and places of interest in Ecuador based on user preferences.

By means of user-based collaborative filtering algorithms, it is proposed to show the main recommendations for a user, based on similarity between information of similar users, the best choice for their next trip is attempted.

Its development is based on the CRISP-DM data mining methodology based on which a predictive model of recommendation will be implemented and the proposed results will be evaluated.

Keywords: recommendation, tourism, filtering, algorithm, Ecuador, Collaborative Filtering, User, Mining, model, Recommendations

Resumen

La presente memoria tiene como principal objetivo mostrar los conocimientos adquiridos por medio del desarrollo de un sistema de recomendación por medio de una página web que nos permita ver las recomendaciones de lugares turísticos y sitios de interés de Ecuador basado en las preferencias del usuario.

Por medio de algoritmos de filtrado colaborativo basado en el usuario se plantea mostrar las principales recomendaciones para un usuario, es decir, basado en similitud entre información de usuarios similares se intenta mostrar la mejor elección para su próximo viaje.

El desarrollo de la misma se basa en la metodología de minería de datos CRISP-DM en base a la cual se implementará un modelo predictivo de recomendación y se evaluará los resultados propuestos.

Palabras Clave: Recomendación, Turismo, Filtrado, Algoritmo, Ecuador, Filtrado Colaborativo, Usuario, Minería, Modelo, Recomendaciones

Índice de ilustraciones

1	Indicadores turísticos - Boletín Mayo 2019 - Ministerio de Turismo	3
2	Oferta Turística destacada - Ecuador-Travel	4
3	Ciclo vida y fases de la metodología CRISP DM (Chapman et al., 2000)	7
4	Catastro Turístico (Ministerios de Turismo)	15
5	Entradas y Salidas Internacionales 2018 (INEC 2018)	16
6	Lugares turísticos (Ministerio de Turismo)	17
7	Entradas y Salidas Internacionales - Serie histórica (Ministerio de Turismo) . . .	18
8	Script para recopilar datos desde Python	19
9	Configuración para descarga de datos desde Zapier	20
10	Datos recopilados de Twitter - Modelo 1	21
11	Datos recopilados de Twitter - Modelo 2	21
12	Limpieza y validación de datos	22
13	Ciclo de Integración de la solución planteada	25
14	Pantalla Principal del Sitio Web	26
15	Formulario de perfilamiento	27
16	Esquema de funcionamiento de un FC	28
17	Descomposición de Valores Singulares	31
18	Entrenamiento: Ingresar datos	33
19	Entrenamiento: Muestra de Datos	33
20	Entrenamiento:: Información de los campos de ingresados	34
21	Items: Ingresar datos de lugar y tipo	34
22	Items: Muestra de Datos de lugar	34
23	Items: Información de los datos cargados	35
24	Evaluación: Ingresar datos de validación	35
25	Evaluación: Muestra de datos para validación	35
26	Scrip: Ingresar datos de entrenamiento	35
27	Matriz: Generar matriz de Usuario y Lugares Turísticos	36
28	Matriz: Muestra de datos desplegados	36
29	Métricas: Definición de las cálculos	37
30	Datos: Definición matriz completa usuarios y Lugares de Interés	38
31	Datos: Muestra matriz completa usuarios y Lugares de Interés	38
32	Evaluación: Definición y proceso de evaluación	39
33	Recomendación: Definición del proceso de recomendación	39

34	Recomendación: Ejecución y entrenamiento con ALS	39
35	Recomendación: Sugerencias en base a ALS	40
36	Recomendación: Ejecución y entrenamiento con BPR	40
37	Recomendación: Sugerencias en base a BPR	40
38	Recomendación: Ejecución de la Sugerencia al usuario	40
39	Recomendación: Sugerencias para el usuario	41
40	Evaluación: Resulpados con el algoritmo ALS	42
41	Evaluación: Resulpados con el algoritmo BPR	42
42	Evaluación: Ejecución de los algoritmos ALS y BPR	42

Índice de tablas

1	Catastro Turístico	16
2	Pasajeros por Mes	16
3	Serie histórica de Entradas y Salidas Internacionales	18
4	Matriz de lugares turísticos por tipo de lugares	23
5	Datos de Entrenamiento	23
6	Resultados de ejecución de la evaluación	43

Índice de contenido

Índice de ilustraciones	III
Índice de tablas	IV
1 Estado del Arte	1
1.1 Introducción	1
1.2 Contexto del Problema	1
1.3 Situación Turística actual de Ecuador	2
1.4 Objetivos	4
1.5 Estructura del trabajo	5
2 Análisis de Requerimientos y herramientas	6
2.1 Metodología de CRISP-DM	6
2.2 Sistemas de Recomendación	8
2.2.1 Descripción de componentes de un sistema de recomendación	9
3 Diseño de la Propuesta	15
3.1 Conocimiento de los datos	15
3.1.1 Integración con la red social Twitter	19
3.2 Preparación de los datos	21
3.3 Definición y Preparación de datos para el modelamiento	22
4 Desarrollo del Sistema Web y algoritmo de recomendación	24
4.1 Desarrollo del SW	24
4.2 Algoritmo de recomendación.	27
4.2.1 Filtrado Colaborativos con retroalimentación Implícita	29
4.2.2 Análisis Semántico Latente LSA	31
4.2.3 Descomposición de Valores Singulares SVD	31
4.2.4 Redes Bayesianas	32
4.3 Tecnología utilizada en el desarrollo	32
5 Análisis y Evaluación de resultados	33
5.1 Algoritmo de recomendación	33
5.1.1 Ingreso de Conjuntos de datos	33
5.1.2 Métricas	36
5.1.3 Procesamiento de datos	37
5.1.4 Evaluación del Modelo	38
5.1.5 Modelo de Recomendación	39
5.2 Evaluación de Resultados	41
5.2.1 Método de evaluación	41
5.2.2 Resultados de la evaluación	42
5.2.3 Conclusiones de la evaluación	43
6 Conclusiones	44
7 Trabajo futuro	45
Referencias	46
8 Anexos	49

1. Estado del Arte

1.1. Introducción

El presente documento representa el trabajo final del master universitarios en Análisis y Visualización de datos masivos correspondiente a la línea de desarrollo de software de un sistema de recomendaciones web para selección de sitios de interés y lugares turísticos de Ecuador basado en las preferencias del usuario, la memoria cuenta con una presentación del contexto del problema que se pretende atender, un detalle de la información turística, objetivos del sistema de recomendación, metodologías y algoritmos a aplicar.

1.2. Contexto del Problema

En la actualidad, existe grandes cantidades de información turística, demanda de paquetes, vuelos, reservas y sitios de interés, aplicaciones realizadas con el objetivo de cumplir la demanda turística, información de la biodiversidad de cada destino turístico y consejos para viajeros pero mucha de esta información no está personalizada para el usuario y en referente a Ecuador los lugares que se muestran son pocos con relación a la potencialidad turística del país.

Ecuador se encuentra entre los países con mayor biodiversidad en el mundo y cuenta con varios reconocimientos internacionales como ser Patrimonio cultural de la humanidad, existe mucha información del país disponible en internet gracias a iniciativas que se han creado para potencializar el turismo y dar a conocer la oferta turística que posee, se han creado portales que muestran destinos, atractivos, diversidad, condiciones climáticas sociales y culturales de cada región

Pese a existir iniciativas para fomentar el turismos e información abundante del país, no se ha puesto a disposición una herramienta que permita al usuario conocer atractivos que puedan ser de su interés, no existen ofertas turísticas que permitan personalizar en base a las preferencias del usuario que sitios son los más opcionados para cada usuario y que cuente con el catastro completo de reservas, playas, y sitios de interés que Ecuador puede ofrecer. (Ramírez y col., 2016, Monge y Perales, 2016)

En base a esto se ha pensado realizar la presente memoria que permita ser el inicio de una nueva forma de presentar la oferta turística en el Ecuador, el sistema de recomendación

tomará el catastro completo de sitios de interés y las preferencias de usuarios para otorgar un top de recomendaciones basados en los datos que el usuario proporcione.

1.3. Situación Turística actual de Ecuador

Ecuador es uno de los países sudamericanos que ha decidido aumentar su oferta turística en base a toda la potencialidad que su diversidad, ubicación y cultura puede proporcionar, se encuentra potencializando y creando iniciativas que permitan llegar a este objetivo. (Prieto, 2011)

El turismo en Ecuador se ha catalogado como uno de los principales pilares de la economía, las cifras que aporta el sector turístico para el país indica que la contribución directa del turismo como muestra el Banco Central del Ecuador en su Boletín (www.bce.fin.ec) que indica el porcentaje del PIB es correspondiente al 2 por ciento, se han registrado 2427,600 llegadas de extranjeros al país en el año 2018, se ha receptorado aproximadamente 1,043.4 millones de dólares por ingresos de divisas por turismo en el mismo año(Ordóñez, 2005).

Estos datos se han recompilado de la información pública que entrega el Ministerio de Turismo en su sitio web y permite ver la importancia de este sector para el país Ballesteros y Carrión, 2007 (servicios.turismo.gob.ec).

En la figura 1 se puede ver de las cifras anteriormente descritas, igualmente muestra el origen de turistas que llegan a Ecuador.

Indicadores Turísticos

Información relevante del Turismo en el Ecuador
MAYO 2019

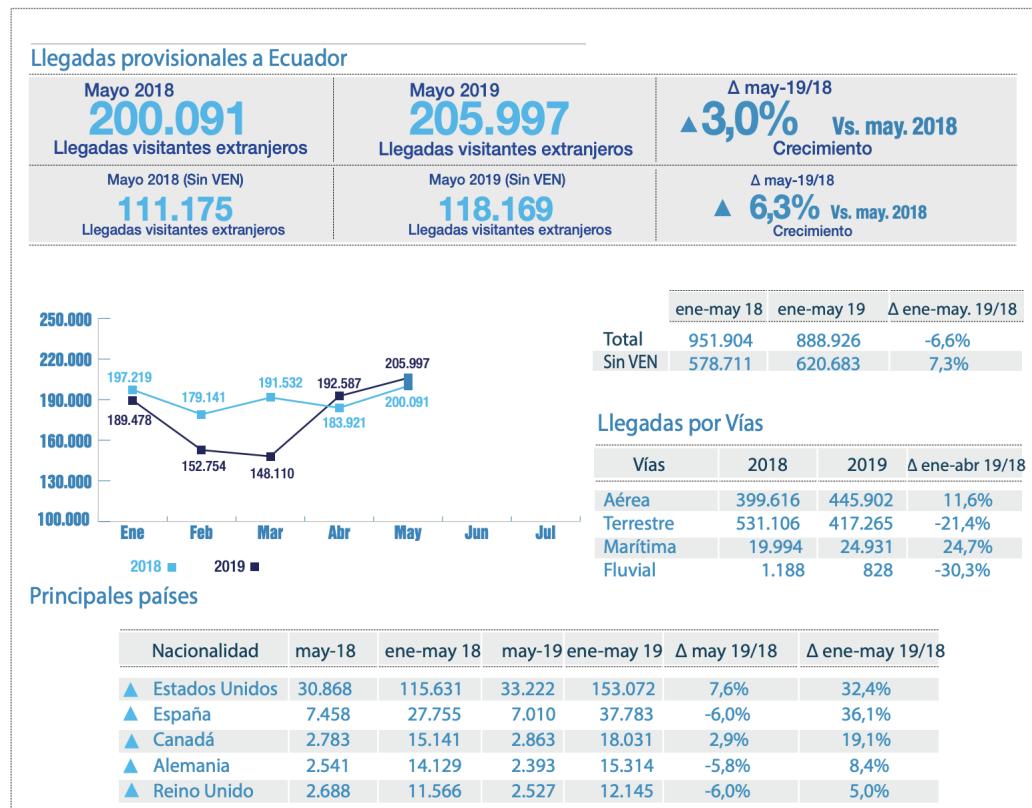


Figura 1: Indicadores turísticos - Boletín Mayo 2019 - Ministerio de Turismo

En la actualidad Ecuador cuenta con 7 reconocimientos como Patrimonios de la humanidad y con 25 galardones de los World Travel Awards edición Sudamérica que respalda la labor que se ha realizado para el sector turístico y que avalan el las maravillas turísticas y espacios naturales del país.

Para potencializar el turismo en Ecuador y dar a conocer el país como objetivo turístico se han creado iniciativas tanto de parte del gobierno como del sector privado, sitios web como www.ECUADOR.travel, GoRaymi, Ecuador Explorer, Clúster Turismo de Romance, entre otras.

Entregan información completa del país, guías para viajar, actividades recomendadas y mucha información de interés para el turista y el objetivo principal de la misma es enamorar al usuario de sus atractivos y paisajes que el país proporciona.



Figura 2: Oferta Turística destacada - Ecuador-Travel

A pesar de las iniciativas creadas que se ha explicado en muchos sitios y aplicaciones solo se presentan destinos puntuales para Ecuador, como Quito, Galápagos, Guayaquil y Cuenca, se muestran algunas reservas como Yasuní pero no se presentan ofertas turísticas para deportes de aventura o paseos que muestren la biodiversidad de flora y fauna que el país posee. Para cubrir este tipo de demanda se está desarrollando este sitio de recomendación, con la información existente y la recopilación de datos por redes sociales se ha creado categorías para diferentes formas de turismo que permita presentar destinos que se acoplen de mejor manera a los gustos y deseos de los usuarios.

1.4. Objetivos

El propósito principal que propone este trabajo de fin de master es elaborar un sistema de recomendación por medio de un modelo de filtrado colaborativo con ponderaciones que realice un análisis híbrido entre la información de contenido almacenada y las preferencias del usuario, entregando, de esta forma un listado de recomendaciones turísticas en un formato Web. Para cumplir con este objetivo general se ha establecido los siguientes puntos:

- Realizar un análisis de perfiles turísticos en base a datos como edad, preferencia climática, preferencias: depuración de la información existente actual con relación a preferencias de turistas al momento de seleccionar su destino turístico.

- Integrar el desarrollo web con la funcionalidad del algoritmo de recomendación: Estudiar las formas de integración entre los diferentes lenguajes de programación para la convivencia de las plataformas a ser implementadas.
- Evaluar los resultados y casos de estudio que presente el modelo: realizar análisis de muestras y datos particulares que entregue el modelo en la fase de evaluación para probar la eficacia del modelo.

1.5. Estructura del trabajo

Con el objetivo de llegar a las metas planteadas para esta memoria se presenta la estructura de la siguiente manera:

Capítulo I – En este capítulo se da a conocer la descripción de la problemática que se pretende soluciona, hacer un análisis de requerimientos y objetivos que se plantean y conocer la metodología que se emplea en el transcurso del desarrollo de esta memoria con el fin de llegar al objetivo planteado

Capítulo II – En este capítulo se estudiará a detalle las técnicas de filtrado colaborativo, sus aplicaciones y diferentes escenarios, también se revisarán las fuentes y depuración necesarias para el aprendizaje del modelo de recomendación que se pretende implementar, para ello se usarán los diagramas y casos de usos que sean necesarios para la descripción de los procesos antes mencionados.

Capítulo III–En este capítulo se describirá los modelos, herramientas y algoritmos que con los que se desarrolla el sistema web de recomendaciones, se detallará las configuraciones modelos y datos que se usan para la implementación

Capítulo IV– En este capítulo se mostrará los detalles de las evaluaciones que muestre el modelo, se realizará análisis y comparación de los algoritmos utilizados y los resultados que entrega la herramienta de acuerdo a las parametrizaciones y los datos usados. Como resultado de este capítulo tendremos el modelo depurado y la integración con la plataforma web que mostrará los resultados.

Capítulo V– En este capítulo podremos revisar los resultados de la evaluación del modelo implementado, se revisarán casos particulares de estudio y se detallará los resultados a desplegar.

2. Análisis de Requerimientos y herramientas

2.1. Metodología de CRISP-DM

Para el desarrollo de ese trabajo de fin de master se ha tomado como referencia la metodología CRISP DM en forma de un laboratorio continuo para ir fortaleciendo el modelo y las fuentes que se van a utilizar para el desarrollo del algoritmo. Pollo Cattaneo, Britos, Pesado y García Martínez, 2009

Es una guía utilizada en el desarrollo de proyectos de minería de datos, viene de las siglas Cross - Industry Standard Process for data mining, y está orientada a facilitar, documentar y mostrar resultados en cada una de sus tapas.

CRISP DM Cobos, Zuñiga, Guarín, León y Mendoza, 2010 comenzó como un proyecto dirigido por grandes empresas como SPSS, Teradata entre otras, quienes aportaron con sus experiencias en manejo de datos y fusionaron sus conocimientos impulsando de esta manera la primera versión de la metodología, empresas como SPSS hasta ahora potencializan la utilización de la metodología en sus productos de análisis y minería de datos

Esta metodología es considerada como una de las más completas dentro de su entorno al tener en cuenta la aplicación de los resultados de negocio es de esta manera que se convirtió en una de las guías más opcionada para implementar en proyectos de minería de datos

CRISP DM divide el proceso de minería de datos en fases principales, tareas, y sus respectivas salidas, cada fase indica la dependencia más importantes y frecuencia y constituyen un ciclo vital, cada uno de estos elementos cuentan con consejos detallados.

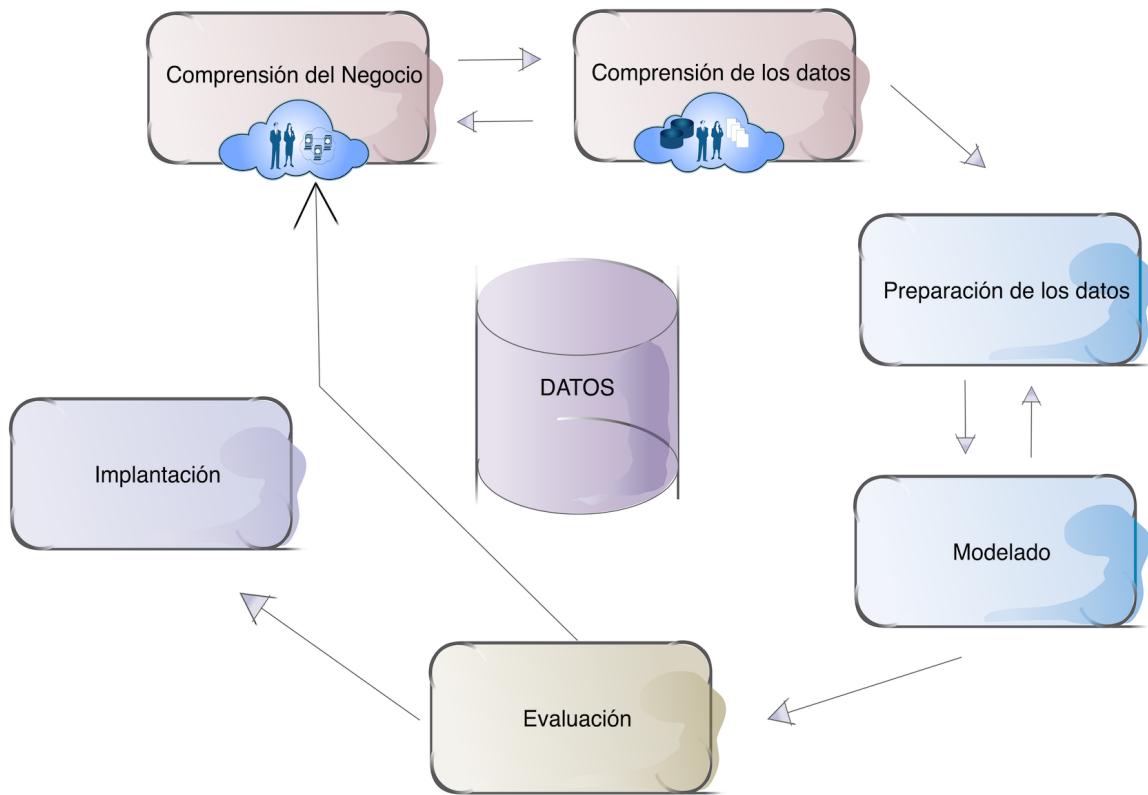


Figura 3: Ciclo vida y fases de la metodología CRISP DM (Chapman et al., 2000)

Las fases de la metodología CRISP DM se componen de:

- Comprensión de negocio: conocimiento funcional del negocio, objetivos y expectativas, dentro de esta fase se establecerán los objetivos principales del negocio, se hará una evaluación inicial y se resaltarán las metas y objetivos de la minería de datos
- Comprensión de los datos: conocimiento de los datos con los que cuenta el negocio, fuentes, estructuras, objetivos de las fuentes a utilizar. En esta fase se realizará la recopilación inicial de datos, exploración, descripción y verificación de la calidad de los mismos.
- Preparación de los datos: depuración de los datos para obtener los conjuntos de datos necesarios para el modelamiento y evaluación, en esta fase se concentran los procesos de selección, limpieza y depuración de datos, se crearán las estructuras, formatos necesarios para el almacenamiento y manejo de la información.
- Modelado: técnicas de minería de datos, en base a los objetivos de negocio en esta fase se selecciona las técnicas y modelos con los que se efectuará la minería de datos, las

tareas principales en esta fase son la selección, diseño y construcción del modelo con las técnicas seleccionadas.

- Evaluación: determinar el grado de eficacia de cada una de las fases anteriores, en esta fase las tareas más importantes son la evaluación de resultados del modelado, verificar que las fuentes y datos permiten llegar a los objetivos planteados anteriormente, verificar que el proceso y resultados del modelo cumplen con los requerimientos del negocio. En esta fase se determinará si se requiere volver a ejecutar las fases anteriores, de esta manera se crea un ciclo que nos permita ir perfeccionando el proceso hasta llegar al objetivo del negocio.
- Implantación: integrar las utilidades del modelo en las tareas de toma de decisión de la organización, en esta fase se realizarán todos los ajustes necesarios para el despliegue de los resultados, las tareas principales en este proceso será la planificación y puesta en producción de los resultados, el monitoreo y mantenimiento del modelo y la revisión del cumplimiento de los objetivos planteados al inicio del proyecto.

Para el desarrollo de la presente memoria se toma en cuenta cada una de las fases que se han descrito, se ha personalizado cada fase a fin de cumplir con el objetivo principal de este trabajo que es crear una aplicación web que permita recomendar lugares turísticos y sitios de interés ecuatorianos basados en las preferencias del usuario.

2.2. Sistemas de Recomendación

En la actualidad, con el uso internet y el crecimiento del comercio electrónico se ha ido creando una sobrecarga de datos, que bajo el correcto manejo y limpieza se constituye en una valiosa fuente de información que proporciona conocimiento tanto para el negocio como para saber el comportamiento de nuestros usuarios. Por medio de este conocimiento se genera los sistemas de recomendación o SR como los llamaremos en adelante. **Yager**

Un SR es un procesamiento que utiliza técnicas de inteligencia artificial supervisada o no supervisada dependiendo el tipo de algoritmos que se generen y permita proporcionar al usuario un listado sugerencias de productos o servicios, es decir predice las mejores opciones y muestra una o varias recomendaciones dentro de una gran cantidad de datos. Nieto, 2007

Con el desarrollo de las técnicas de recomendación, grandes empresas como Amazon, Netflix o Youtube han desarrollado estrategias y algoritmos en sus plataformas para poten-

cializar su negocio basados en filtrado colaborativo FC por medio de las cuales de manera dinámica siempre están presentando recomendaciones con sugerencias interesantes a sus usuarios. En definitiva, un SR está basado en una o varias técnicas que entregan una sugerencia en base a ponderaciones o puntuaciones anteriormente almacenadas y procesadas y muestran un resultado basado en sugerencias, entre las técnicas más utilizadas se encuentra el FC como lo hemos mencionado anteriormente. **Ricci**

2.2.1. Descripción de componentes de un sistema de recomendación

A continuación, se enumerarán los principales componentes de un SR y la importancia de cada uno de los mismos. Este pequeño preámbulo nos mostrará un panorama más amplio de que consiste y que es necesario para los mismos.

Perfil:

Cuando trabajamos con un SR es necesario que tengamos en cuenta cómo y de donde vamos a empezar con la exploración de los datos, para generar un primer punto de partida se crea un perfil de usuario que nos permita crear un primer panorama de las preferencias de los clientes.

Para crear el perfil podemos escoger entre los métodos de recolección implícitas o explícitas, para comprender a qué se refiere cada uno de ellos es necesario que verificar en qué consiste cada uno de los mismos. Perfilamiento Implícitos, se refiere a recolección de datos que se genera en la navegación, transacción o búsquedas que genere el cliente, este proceso se puede generar guardando los temas que visitó el usuario, obteniendo listados de artículos seleccionados o vistos, analizando frecuentemente el número de visitas o la puntuación que ha creado el mismo o por medio de un análisis de sentimiento a redes sociales. Perfilamiento Explícito, se concentra en recolección de datos otorgados por el usuario en forma de solicitud, para este proceso se pueden utilizar varias herramientas que permitan almacenar las preferencias del usuario tales como ponderaciones, listas de deseos, permitir crear preferencias, encuestas, cuestionarios, o solicitudes al inicio de la navegación. **Herlocker**

Con los datos que ha proporcionado el usuario ya sea de manera implícita o explícita se valorará sus preferencias y se podrá generar un perfil que determine el comportamiento y preferencias del usuario y se podrá empezar a trabajar en el SR con esta información.

Entre las técnicas con las que se procesa esta información se encuentran los árboles de decisión, redes bayesianas, reglas de asociación entre otras que se explicará más ampliamente en el transcurso de este documento.

Técnicas de Recomendación:

Otro de los componentes de un SR y que será el centro de esta investigación son las técnicas o algoritmos usados para las sugerencias al usuario, se basan en filtrar la información es decir generar procesos de selección de datos en base a un o varios argumentos que nos proporcionen un grupo de datos más específico, entre las formas de filtrado que se han desarrollado se encuentran:

■ Filtrado demográfico:

Consiste en trabajar con los datos demográficos de la persona, tales como edad, sexo, ocupación, entre otros, para relacionar grupos de usuarios. Este tipo de filtrado requiere que el perfil del usuario esté con datos personales sobre los que se realizará la clasificación demográfica (Burke, 2000)

Clasifica en grupos demográficos a todos los usuarios según los datos previamente recopilados y se enriquece el modelo alimentando con varios atributos como profesión, nivel de estudios, edad, género y en base a los cuales se centrará las recomendaciones. (Pazzani, 1999)

■ Filtrado basado en contenidos:

Los SR basados en contenidos también se los llama no Colaborativos y se centran en realizar recomendaciones utilizando las preferencias que ha sido ingresadas por el usuario, para esto el usuario debe ser un cliente activo y en base a todas las interacciones que haya realizado el mismo el sistema le mostrará una recomendación.

La recolección de datos ya sea explícita o implícita en este tipo de SR es esencial para poder entregar una recomendación acertada, mientras más datos se tenga del usuario más cercano a la realidad estará la predicción. Este tipo de filtrado usa algoritmos que generen similitudes en función a contenidos que poseen características dentro de las preferencias del consumidor.

■ Filtrado Colaborativo:

El FC es un sistema de recomendación que se basa en el conocimiento del usuario,

consiste en analizar las preferencias entre los ítems y la información del cliente. La recopilación de la información en este tipo de filtrado se basa en los datos ingresados de puntuaciones, retroalimentación a productos, visitas, análisis de sentimiento en redes sociales. Se basa principalmente en las preferencias del usuario y necesita de grandes volúmenes de información para entregar un resultado confiable. (Burke, 2000)

Describiendo esta técnica de una manera sencilla se podría decir que el filtrado colaborativo consiste en tener dos usuarios, ya sean estos A y B, en una misma plataforma, y tienen gustos similares, las preferencias de B pueden ser recomendadas al usuario A. (Herlocker, Konstan, Terveen y Riedl, 2004)

El filtrado colaborativo se divide en dos clasificaciones de filtrado, las cuales son:

- **Métodos basados en modelos:** Se basan principalmente en algoritmos de aprendizaje automático y análisis de datos utilizando técnicas de minería de datos y consiste en generar entrenamiento de grandes volúmenes de información que generen patrones que permitan determinar predicciones, este tipo de técnicas tiene un mejor rendimiento, pero su implementación puede requerir arquitecturas más robustas. El cimiento de las recomendaciones para este tipo de FC son los modelos de aprendizaje automático basados en algoritmos, entre ellos podemos hablar de:

Modelos de Clustering Permite identificar agrupaciones o segmentaciones de acuerdo a similitudes entre los objetos de un conjunto de datos. (Kaufman y Rousseeuw, 2009)

Redes Bayesianas: Permite identificar patrones por medio de un modelo probabilístico que representa un conjunto de variables aleatorias y dependencias de un nodo también llamado grafo, permite conectar la secuencia de variables las cuales generar aprendizaje de estructuras y parámetros (Pearl, 1994, Césari, 2006)

Semática latente: Permite identificar patrones de contenido por medio de indexación y recuperación de datos, se le llama descomposición de valores singulares (SVD), extrae información por medio de significados similares. (Baeza-Yates, Ribeiro-Neto y col., 1999)

- **Métodos basados en memoria:** Se basa principalmente en algoritmos de aprendizaje supervisado, es decir en base a los datos proporcionados se evalúa las similitudes entre los conjuntos de datos similares y se realiza una comparación por

medio de cálculos matemáticos en base a los cuales se realizará la recomendación al usuario. Es uno de los algoritmos de filtrado colaborativo más usado y fácil de implementar, un ejemplo podemos ver el Sistemas de recomendación para webs de información sobre la salud (Seguido Font, 2009)

Entre los principales modelos de aprendizaje supervisado utilizados para este modelo podemos encontrar Similitud de cosenos: Consiste en medir la similitud entre dos vectores, es una función que nos proporciona el coseno del ángulo y despliega un valor comprendido entre 1 y -1, si los valores entre los vectores forman ángulos similares o aproximadas quiere decir que son ángulos para el mismo sentido.

Algoritmo K-Veinos o K Nearest Neighbours K-NN: Es un algoritmo de aprendizaje supervisado del tipo de clasificación que trabaja con el nuevo conjunto de datos de un usuario y calcula las distancias entre los elementos almacenados y el vector actual para entregar la distancia entre los elementos y mostrar la distancia más corta entre ellos. (Keller, Gray y Givens, 1985, Cover y Hart, 1967)

Este modelo no necesita un grupo de datos previamente entrenados, cuando el usuario genera una muestra de datos el algoritmo se realiza el cálculo para determinar la cercanía entre los datos almacenados y el nuevo registro. Este algoritmo tiene dos componentes muy importantes y sensibles que son, la variable K , que se refiere al número de valores que se tomarán en cuenta y según este valor crezca o disminuya los resultados serán diferentes y la métrica de similitud, es el valor que nos permite contener los pesos que mostrarán la cercanía. (Denoeux, 1995)

Donde se puede ver claramente cómo cambia su clasificación según la variable K. Para K+1 el punto principal es categorizado como blanco, para k= 2 no existe un criterio de clasificación claro y para K =3 se clasificará como negro. (Dudani, 1976)

Para determinar la métrica de distancia se puede utilizar:

Distancia Euclides: se mide la longitud que une a dos segmentos, se dice que es una consecuencia del teorema de Pitágoras donde triángulo sea (c,a) y (d,b)

$$d_{AB} = \left[(X_A - X_B)^2 + (Y_A - Y_B)^2 \right]'$$
 (1)

Distancia Manhattan: indica que la distancia entre dos puntos es la suma absoluta

de sus coordenadas, un ejemplo es la longitud de una escalera que va desde (a,b) al punto (c,d), una ruta q une el segmento a-b y c-d

$$d_{AB} = |X_A - X_B| + |Y_A - Y_B| \quad (2)$$

MSD (diferencia cuadrática media): Esta medida es la exactitud de los valores en una serie de tiempo, se utiliza para ajustar los resultados de modelos en una serie de tiempo, mientras menor sea el espacio es menor el ajuste también

$$sim(x, y) = 1 - \frac{1}{\#B_{x,y}} \sum_{i \in I_u} \left(\frac{r_{x,i} - r_{y,i}}{\max - \min} \right)^2 \in [0, 1] \quad (3)$$

Correlación de Pearson: Es el valor utilizado para medir la relación entre dos variables.

$$sim(x, y) = \frac{\sum_{i \in B_{xy}} (r_{x,i} - \bar{r}_x) \cdot (r_{y,i} - \bar{r}_y)}{\sqrt{\sum_{i \in B_{xy}} (r_{x,i} - \bar{r}_x)^2} \cdot \sqrt{\sum_{i \in B_{xy}} (r_{y,i} - \bar{r}_y)^2}} \in [-1, 1] \quad (4)$$

Similitud de cosenos: Es la distancia resultante de medir dos usuarios en función a su similitud, distancia que forma entre los ángulos.

$$sim(x, y) = \frac{\sum_{i \in B_{xy}} r_{x,i} \cdot r_{y,i}}{\sqrt{\sum_{i \in B_{xy}} r_{x,i}^2} \cdot \sqrt{\sum_{i \in B_{xy}} r_{y,i}^2}} \in [0, 1] \quad (5)$$

- **Filtrado Híbrido:** Un SR híbrido consiste en unir varios algoritmos de las técnicas de memoria y modelo ya sea para aumentar la precisión o para disminuir problemas específicos en el algoritmo. En base a la combinación resultante de estas técnicas el algoritmo proporciona recomendaciones al usuario. Este tipo de algoritmos híbridos pueden entregar predicciones más efectivas pero el coste es más alto, existen varios ejemplos de filtrado Híbrido, entre el más conocido es Amazon, entre ellos podemos ver:

Modelo de vigilancia tecnológica apoyado por recomendaciones basadas en el filtrado colaborativo (Abreu-Lee, Infante-Abreu, Delgado-Fernández y Delgado-Fernández, 2013)

Sistema de recomendación por filtrado colaborativo para el sistema de publicación de contenido multimedia - VideoWeb 1.0 (Castellanos, 2014)

3. Diseño de la Propuesta

3.1. Conocimiento de los datos

Para el desarrollo de la aplicación volveremos al esquema de la metodología CRISP DM, hasta el momento trabajado en la fase del conocimiento del negocio, a partir de este punto comenzaremos a trabajar en las siguientes fases que es el conocimiento de los datos. Se requiere hacer un sistema de recomendación basado en las técnicas de filtrado colaborativo en base a usuarios, las principales fuentes que se requieren son los sitios turísticos e información de usuarios, la información pública no tuvo complejidad en descargarla y almacenar, pero en el caso de la información de usuarios se encontraron algunas limitantes y para cumplir con el objetivo se optó por utilizar datos de la red social Twitter. Para el desarrollo de este trabajo se recompiló información gratuita de entes de control, como el Ministerio de Turismo, Ecuador en cifras, como:

Catastro Turístico:

Descripción geográfica de lugares turísticos de Ecuador, consiste en un conjunto de archivos catalogados como datos abiertos que poseen toda la información catastrada de la planta turística, entre los archivos tenemos:

- Diccionario de datos: la descripción cada campo del archivo principal
- Metadato: archivo con la información descriptiva como fuente, versión, etc.
- Catastro Turístico: base de datos con toda información referente al castro turístico de Ecuador.

	Ministerio de Turismo	Diccionario de Datos No. 01	Institución: Ministerio de Turismo Versión (Actualización): 01 Código Documento: DD-01-2015-V01
Código Documento Referencia (Hoja de Ruta):	HR-01-2015-V01		
Nombre del Conjunto de Datos:	Catastro Turístico		
Nombre del Recurso:	Catastro por provincias a nivel nacional		
Descripción del Recurso:	Contiene información de toda la planta turística del Ecuador		
URI del Recurso:			

Figura 4: Catastro Turístico (Ministerios de Turismo)

La figura nos muestra el formato estándar de la documentación oficial de Ministerio de Turismo Ecuatoriano y una breve descripción informativa correspondiente al Catastro turístico

REGISTRO	NOMBRE	RUC	DIRECCIÓN	CANTÓN	PROVINCIA
101500002	AVILESWORLD TRAVEL		SAN BLAS	CUENCA	AZUAY
101500004	BOONROUTE		SANGURIMA	CUENCA	AZUAY
101500013	HUALAMBARI TOURS		BORRERO	CUENCA	AZUAY
101500020	METRO TOURS		CALLE LARGA	CUENCA	AZUAY

Cuadro 1: Catastro Turístico

El cuadro es una muestra de la información del Catastro Turístico

Entradas y Salidas Internacionales 2018 - Estadísticas Vitales

En un paquete de archivos, catalogados como datos abiertos que posee la información completa de estadísticas de entradas y salidas internacionales en el año 2018. Información de Instituto nacional de Estadísticas y Censos INEC, el paquete contiene

- Diccionario de datos: la descripción cada campo del archivo principal
- Metadato: archivo con la información descriptiva como fuente, versión, etc.
- Entradas y Salidas: Archivo con la base de datos de la información de entradas y salidas internacionales de Ecuador.

	Diccionario de Datos No. 02	Institución: Instituto Nacional de Estadística y Censos Versión (Actualización): 01 Código Documento: DD-01-2018-V01
Código Documento Referencia (Hoja de Ruta):	HR-01-2018-V01	
Nombre del Conjunto de Datos:	Estadísticas Vitales	
Nombre del Recurso:	Entradas y Salidas Internacionales 2018 - Estadísticas Vitales	
Descripción del Recurso:	La información Estadística de Entradas y Salidas Internacionales (ESI) 2018, está constituida de 7.653.258 registros, que corresponde a los movimientos de entradas y salidas de ecuatorianos y extranjeros registradas en las diferentes Jefaturas de Migración que realizan el control migratorio correspondiente en el año de investigación y de 28 variables. El registro contiene la información recolectada en el Sistema de Migración Ecuatoriano, que contiene el formato electrónico de la tarjeta andina de migración la cual es llenada al momento en que las personas ingresan o salen del país, en las diferentes Jefaturas de Control Migratorio	

Figura 5: Entradas y Salidas Internacionales 2018 (INEC 2018)

La figura 5 muestra el formato oficial del Instituto Ecuatoriano de Estadísticas y Censos, y describe la información de entradas y salidas correspondiente al año 2018, describe el contenido de la información del paquete entregado.

Passengers per month (International)			
Month	2017	2018	2019
Jan		178740	189584
Feb	0.068955206	189359	177122
Mar	148691	194938	0

Cuadro 2: Pasajeros por Mes

Lugares turísticos:

Lugares turísticos: Es un archivo que cuenta con dos archivos referentes a catálogo de Sitios de Interés, Reservas y lugares turísticos

- Archivo en Excel y data set de Lugares de Interés

LT01	Nombre	Tipo	TipoLugar	País	Provincia	Ciudad
LT02	Río Santa Bárbara	LugaresTurísticos	Río	Ecuador	Azuay	Gualaceo
LT03	Represa de Paute	LugaresTurísticos	Represa	Ecuador	Azuay	Cuenca
LT04	El Turi	LugaresTurísticos	Iglesia	Ecuador	Azuay	Cuenca
LT05	Cascada Irha	LugaresTurísticos	Casacada	Ecuador	Azuay	Cuenca
LT06	Parque Nacional Cajas	LugaresTurísticos	Parque	Ecuador	Azuay	Cuenca
LT07	Reserva faunística el Chimborazo	LugaresTurísticos	Reserva	Ecuador	Bolívar	Riobamba
LT08	Complejo turístico las cochas	LugaresTurísticos	Atractivo Turístico	Ecuador	Bolívar	Guaranda
LT09	Santuario De La Virgen del Huayco	LugaresTurísticos	Iglesia	Ecuador	Bolívar	San José de Chimbo
LT10	Cuevas de Tiagua	LugaresTurísticos	Montaña	Ecuador	Bolívar	Guaranda
LT11	Volcán Carihuairazo	LugaresTurísticos	Volcán	Ecuador	Bolívar	Riobamba
LT12	Complejo turístico Ingapirca	LugaresTurísticos	Complejo Arqueológico	Ecuador	Cañar	Cañar
LT13	Llamadas de cultos rituales	LugaresTurísticos	Llamina	Ecuador	Cañar	Cañar

Figura 6: Lugares turísticos (Ministerio de Turismo)

La figura 6 se refiere al registro de Lugares turísticos catastrados por el Ministerio de Turismo, cada sitio se encuentra de una categoría llamada Tipo Lugar, esta información es el principal insumo para el modelo de esta investigación.

Tabulaciones y Series históricas 2018

Tabulaciones y Series históricas 2018 Paquete de datos que contiene :

- Índice
- Glosario de términos
- Ficha Metodológica
- Anexo 1
- Series numeradas según el índice
- Archivos Tableau estadísticos

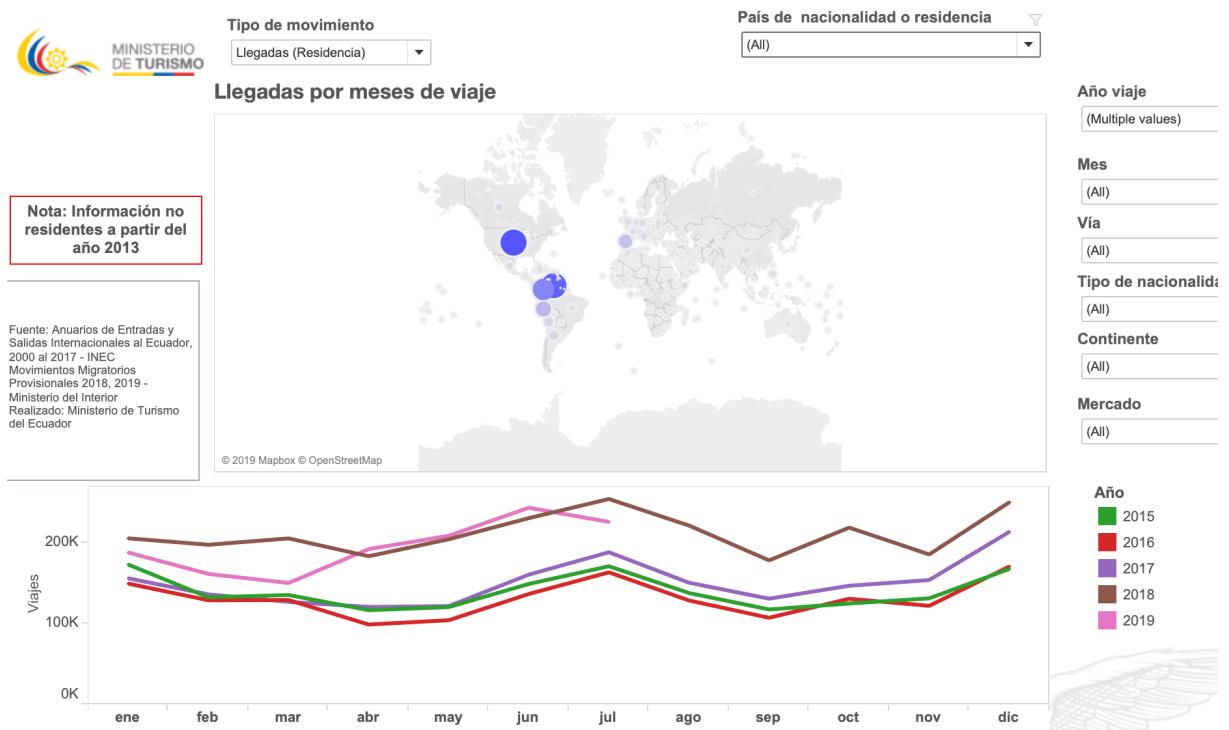


Figura 7: Entradas y Salidas Internacionales - Serie histórica (Ministerio de Turismo)

La figura 7 contiene la captura del tablero principal de Entradas y Salidas con los datos históricos hasta el año 2018

Series No. 1.1.1

Entradas, salidas, movimientos brutos y movimientos netos
(Periodo 1997 - 2018)

Años	Entradas	Salidas	Movimientos Brutos	Movimientos Netos
2012	2,297,211	2,240,008	4,537,219	57,203
2013	2,507,173	2,447,510	4,954,683	59,663
2014	2,826,666	2,759,821	5,586,487	66,845
2015	2,919,356	2,862,444	5,781,800	56,912
2016	2,911,927	2,929,849	5,841,776	-17,922
2017	3,114,763	3,065,412	6,180,175	49,351
2018	3,903,315	3,749,943	7,653,258	153,372

Cuadro 3: Serie histórica de Entradas y Salidas Internacionales

3.1.1. Integración con la red social Twitter

Para obtener información de usuarios, fue necesario manejar acceso a redes sociales, en este caso se generó el clave de acceso gratuito para de desarrollador de Twitter y en base al mismo se realizaron varias conexiones con el fin de obtener el mayor volumen de información para esto se realizaron los siguientes escenarios:

-Cosecha de datos en base a Python

```
UoM-twitter-cosecha-información
VACACIONES ECUADOR
=====
'''
import couchdb
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import json

###API T-for-health-7#####
ckey = "qq2l
csecret = ".
atoken = "2l
asecret = "
#####

class listener(StreamListener):
```

Figura 8: Script para recopilar datos desde Python

La figura 8 muestra un fragmento del script realizado para conectar la base de datos con las librerías de tweepy y descargar la información, la palabra clave fue vacaciones. Este procedimiento descargó mucha información no productiva y la depuración de la misma fue por filtros de ubicación geográfica y palabras claves como Ecuador y Vacaciones en las búsquedas desplegaron información no pertinente para la investigación.

- Cosecha por medio de páginas web

/ Save Twitter mentions to a Google Sheets spreadsheet

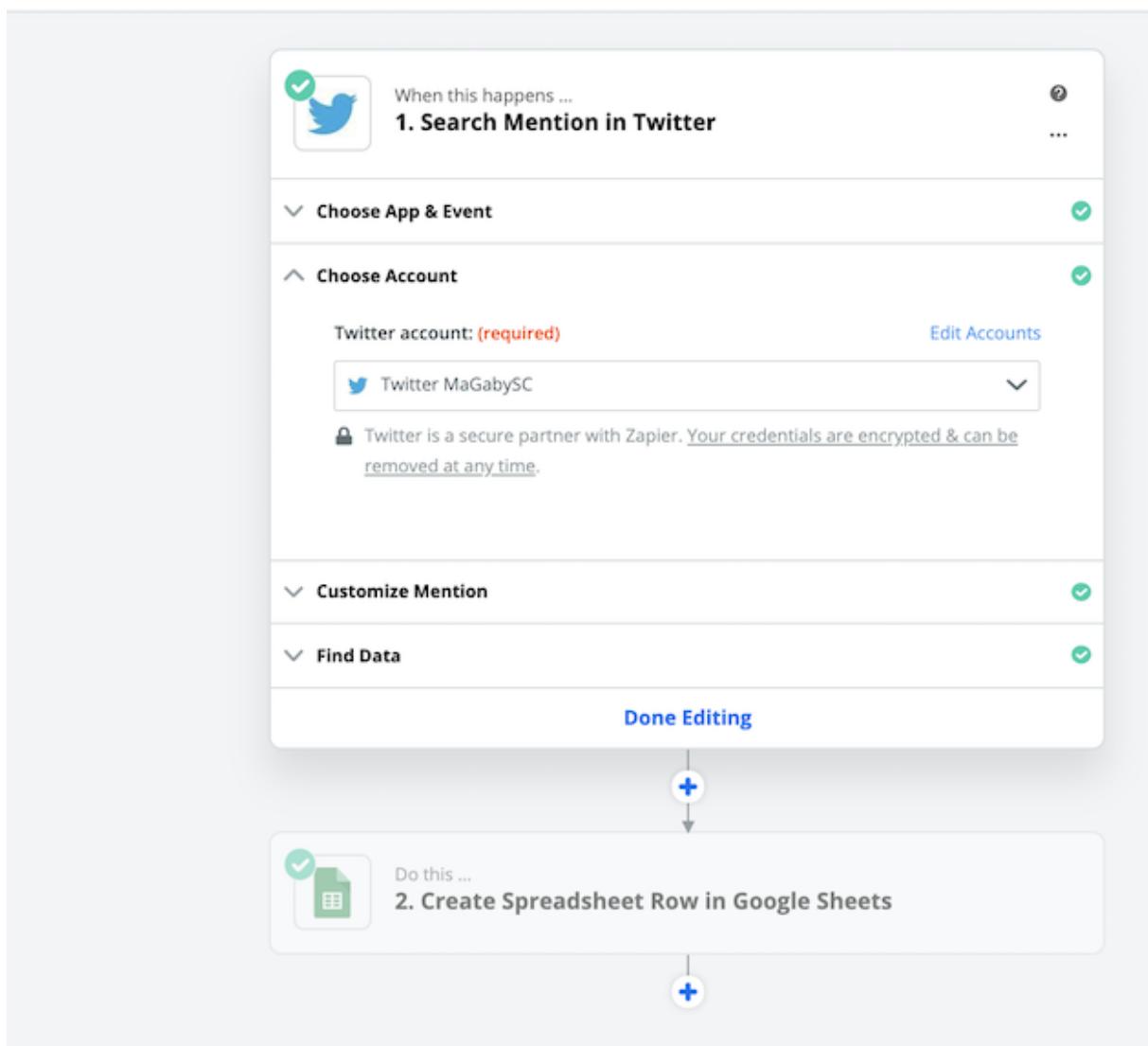


Figura 9: Configuración para descarga de datos desde Zapier

Con herramientas web como Zapier.com y Ifttt.com se realizó la descarga de información más específica con solo los campos necesarios que se requería para una búsqueda más eficiente, los resultados fueron más limpios, pero de igual manera se depuro por medio de base de datos y filtros para limpiar la información Los datos que se descargaron de estas aplicaciones se guardaron en la nube con los siguientes formatos:

RT @RdeRuido: Los #oídos no tienen #vacaciones, ¡cuídalo!	
Entrevista Dr. Joaquín Alacio	
@quironsalud	
#conRderuido NO al #ruido y SÍ a la #salud y a la #convivencia	
Información y formación para decidir	
https://t.co/K8m0wj5RYW https://t.co/9SjDV8ihAK	http://twitter.com/U_Cc August 24, 2019 at 03:00AM
@MrDurden1983	Cuando el levante aprieta, de visita turística por #MedinaSidonia #Cadiz #And
@CARMENCAB	http://twitter.com/MrDu August 24, 2019 at 03:03AM
@SensideCorda	#familytime #vacaciones #dog https://t.co/pSVazqSk3M
@leandrofmach	http://twitter.com/CARI August 24, 2019 at 03:04AM
17 Te explico en 45 sg un juego #Vivir #Alegria #Felicidad #Familia #Amigos #	http://twitter.com/Sens August 24, 2019 at 03:05AM
Bye, bye #vacaciones, #GRACIAS por recargarme las pilas 😊 Volvemos a la	http://twitter.com/leand August 24, 2019 at 03:08AM
DT @TauConsultores_ Los thxvacaciones no se pueden sustituir por una compon	

Figura 10: Datos recopilados de Twitter - Modelo 1

En las capturas de pantallas podemos ver archivos con los datos, se ha ocultado las columnas referentes a nombres de usuarios

usuario	nombre	detalle	datos 1	datos2
KIZILKAYA007	Ramazan Kızılık	Antalya Valiliği İl	Antalya_merhaba: #AntalyaHello...	RT @Antalya_merhaba: 🏙️ 800 yıllık Sarıhacılar Köyü #turist akınına uğruyor https://t.co/hzwsD1mK43 #AntalyaMerhaba
cinkoprusu	Çin Köprüsü	Çin ve Çince ile ilgili sizleri bilgilendirmek	#Çin #çince #sanat #kültürsanat #site #voleybol #turist #turizm #çinköprüsü #he	A Milli Erkek Basketbol Takımı, çarşamba günü Çin yolcusu Detaylar: https://t.co/0f30000000
ahmetselim14	Ahmet Selim	<u>Adiyaman Gergerli - Gazeteci</u>	Tut ilçesinin simgelerinden biri dağ keçisi. #adiyaman #tut #turist #gezi #travelphotography #camping #dağkeçisi #mountain	

Figura 11: Datos recopilados de Twitter - Modelo 2

Figura 8: Datos recopilados de Twitter Modelo 1 En las capturas de pantallas podemos ver archivos con los datos, se ha ocultado las columnas referentes a nombres de usuarios

Figura 9: Datos recopilados de Twitter Modelo 2

3.2. Preparación de los datos

Para poder trabajar con la información recopilada fue necesario depurar la información, segmentar los datos útiles y limpiar la información, en el proceso de depuración se perdió más del 40 por ciento. Para la depuración y limpieza de datos se realizaron filtros aplicados en la librería de tweepy y en las herramientas web con los términos referentes a Vacaciones, Ecuador, Turismo Ecuador. El volumen más alto de información fue con el tag Vacaciones + Ecuador

The screenshot shows a database interface with a code editor at the top containing a SQL query:

```

24
25 select 'US0001',description from Datos2 where description like '%Ecuador%'
26

```

Below the code editor, there is a summary of the data:

- 2,000 rows
- 0 rows
- 448,051 rows
- 910,640 rows

On the right side, there is a small icon labeled 'R'.

The main area displays a table titled "Results" with one row selected, labeled "Result Set 1". The table has two columns: "id" and "description". The data is as follows:

	Results	Result Set 1
'US0001'	description	#Travel #viajeros #turismo #aventuras #vacaciones #Tour #Guayaquil #Ecuador
6 US0001		Ven para viajarte porque para amarte tengo que viajarte #vacaciones #primeroEcuador https://t.co/
7 US0001		#100cosasDelEcuatoriano #primeroEcuador _ _ _ _ _ #Alausí #vacaciones Super Diego on
8 US0001		Cuando algo te llena de alegría 😊 #Quito #Montañita #Viaje #Felicidad #Miércoles #Hoy #vacaciones #paisajes #Ecuador #arcoiris #rainbow #cc

Figura 12: Limpieza y validación de datos

En la figura 12 se muestra el proceso de limpieza y validación de la información, fue necesario reducir filtrando los datos con las palabras claves, al obtener el destino del usuario se codifica en base al catastro de lugares turísticos antes mencionados, los nombres de igual manera se han codificado.

La información que se almacenó en CouchDB de igual manera fue filtrada en base a las palabras claves y el destino Ecuador.

3.3. Definición y Preparación de datos para el modelamiento

En base a la metodología empleada para este desarrollo se procesa a información con la finalidad de obtener 3 conjuntos de datos, mismos que servirán de insumo para el entrenamiento, procesamiento y evaluación del algoritmo de filtrado colaborativo que se desarrollará más adelante.

La información obtenida por el Ministerio de Turismo basada en lugares turísticos y el Tipo de lugar al que pertenece se codifica en base a los requerimientos del modelo, de esta manera se despliega una matriz o tabla cruzada entre cada lugar turístico y el tipo de lugar al que pertenece, como muestra la tabla (4), posteriormente se modificó la codificación de lugar para procesar como tipo numérico.

ID_Lugar	Nombre	Provincia	Ciudad	Acuario
LT01	Río Santa Bárbara	Azuay	Gualaceo	0
LT02	Represa de Paute	Azuay	Cuenca	0
LT03	El Turi	Azuay	Cuenca	0
LT04	Cascada Irha	Azuay	Cuenca	0
LT05	Parque Nacional Cajas	Azuay	Cuenca	0
LT06	Reserva faunística el Chimborazo	Bolívar	Riobamba	0
LT07	Complejo turístico las cochas	Bolívar	Guaranda	0
LT08	Santuario De La Virgen del Huayco	Bolívar	San Jose de Chimbo	0
LT09	Cuevas de Tiagua	Bolívar	Guaranda	0
LT10	Volcan Carihuairazo	Bolívar	Riobamba	0

Cuadro 4: Matriz de lugares turísticos por tipo de lugares

El conjunto de datos de entrenamiento y validación constan del Id de usuario, el id del lugar, la ponderación por destino y la clave única. La información para el modelo se dividió de la siguiente manera: el universo consta de 1'300.000 datos en total, del mismo se divide 30 para pruebas y validaciones y el 70 para entrenamiento del modelo con el siguiente formato:

ID_Usuario	ID_Lugar	Rating	id_pk
US000001	LT33	1	US000001LT33
US000001	LT82	2	US000001LT82
US000001	LT68	5	US000001LT68
US000001	LT78	5	US000001LT78
US000001	LT73	5	US000001LT73
US000001	LT71	3	US000001LT71

Cuadro 5: Datos de Entrenamiento

La tabla (5) nos enseña una muestra pequeña de los datos que utilizaremos en el entrenamiento y validación del modelo

4. Desarrollo del Sistema Web y algoritmo de recomendación

4.1. Desarrollo del SW

En este capítulo se presenta el diseño y desarrollo de las dos partes principales que plantea esta investigación correspondiente al sitio web y el sistema de recomendación por medio de un algoritmo de filtrado colaborativo.

Dentro del desarrollo del sistema web se plantea crear una interfaz que nos permita manejar interactuar con el usuario y con el algoritmo de recomendación, en esta sección se explicara el diseño, modelamiento y desarrollo de dichas partes.

Un sistema web SW, como indica su nombre es una interfaz en una plataforma web que permite gestionar acciones que requiere el usuario y entregar una respuesta para satisfacer una necesidad puntual, un servicio o facilitar el objetivo del mismo en la aplicación web, en este caso la parte web es un complemento de visualización y captura de datos para la gestión de recomendación que se plantea para Lugares turísticos en el Ecuador.

Para el desarrollo de las interfaces se debe tener claro el objetivo y alcance del sitio web para poder modelar la arquitectura necesaria para gestionar su funcionamiento, en este caso las herramientas que se usarán para esta primera parte será la siguiente.

- Servidor Web (MAMP)
- Motor de Base de Datos
- Gestor de Contenidos (WordPress)
- Plantilla o diseño Web del Sitio

Para levantar la plataforma web se usó el servidor Web de Apache MAMP con la base de datos integrada MySQL, con esta configuración básica se realizó la configuración del SW y las personalizaciones necesarias.

El sitio como tal requiere más componentes gráficos e información descriptiva para cada lugar turístico para que sea más amigable con el usuario y permita una mejor experiencia, por

el momento es un sitio informativo y estático con excepción de la integración de Python para los resultados de la recomendación.

Para la construcción del sitio web se tomaron encuentra únicamente las pantallas principales para alimentar el modelo de filtrado colaborativo, con esta premisa se diseñó la pantalla principal con la información básica del sistema y las preguntas de perfilamiento necesarias para empezar con el uso de la herramienta.

La principal función de la plataforma web es facilitar el perfilamiento del usuario, esta opción consiste en un pequeño formulario con preguntas básicas y valoraciones de por medio de preguntas de opción múltiple. Este formulario no debe ocupar mas de un minuto para que el usuario ingrese sus preferencias

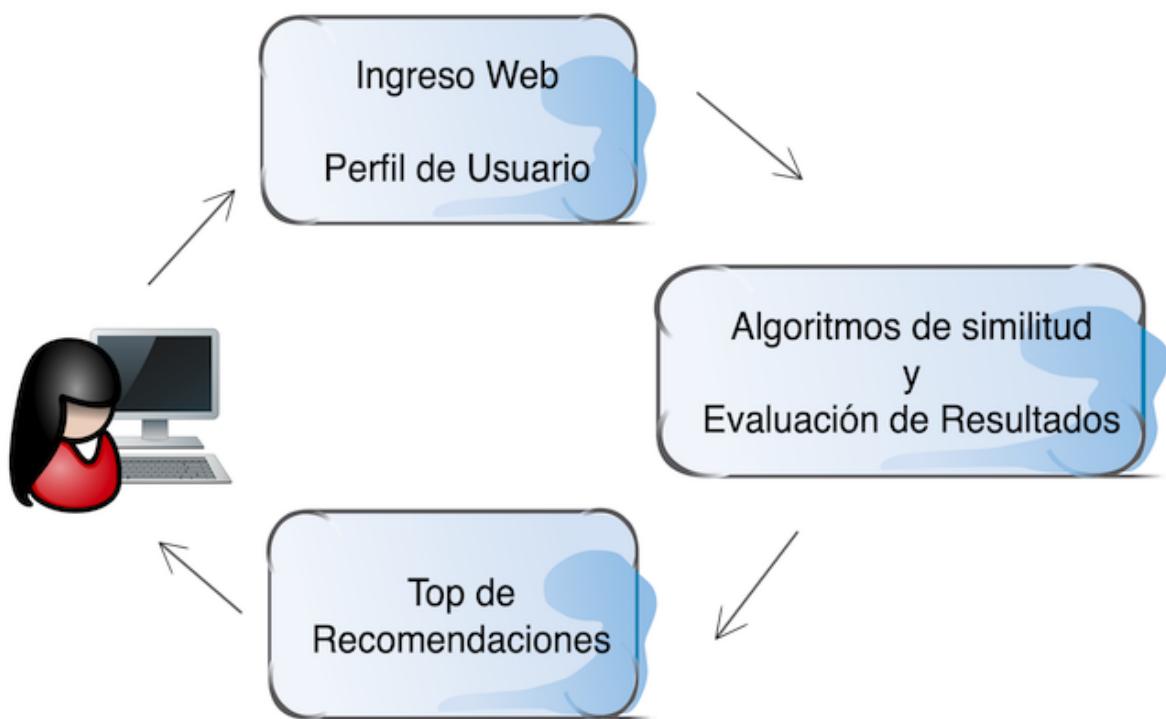


Figura 13: Ciclo de Integración de la solución planteada

Como muestra la figura 13 cuando el usuario ingrese al sitio web podrá ingresar sus principales preferencias y de esta forma el proceso de recomendación se activara, validando las similitudes y usuarios similares, para posteriormente entregar un top de recomendaciones. Para este piloto el sitio web se llamará TRAVEL NOW, se ha diseñado la página principal para su manejo

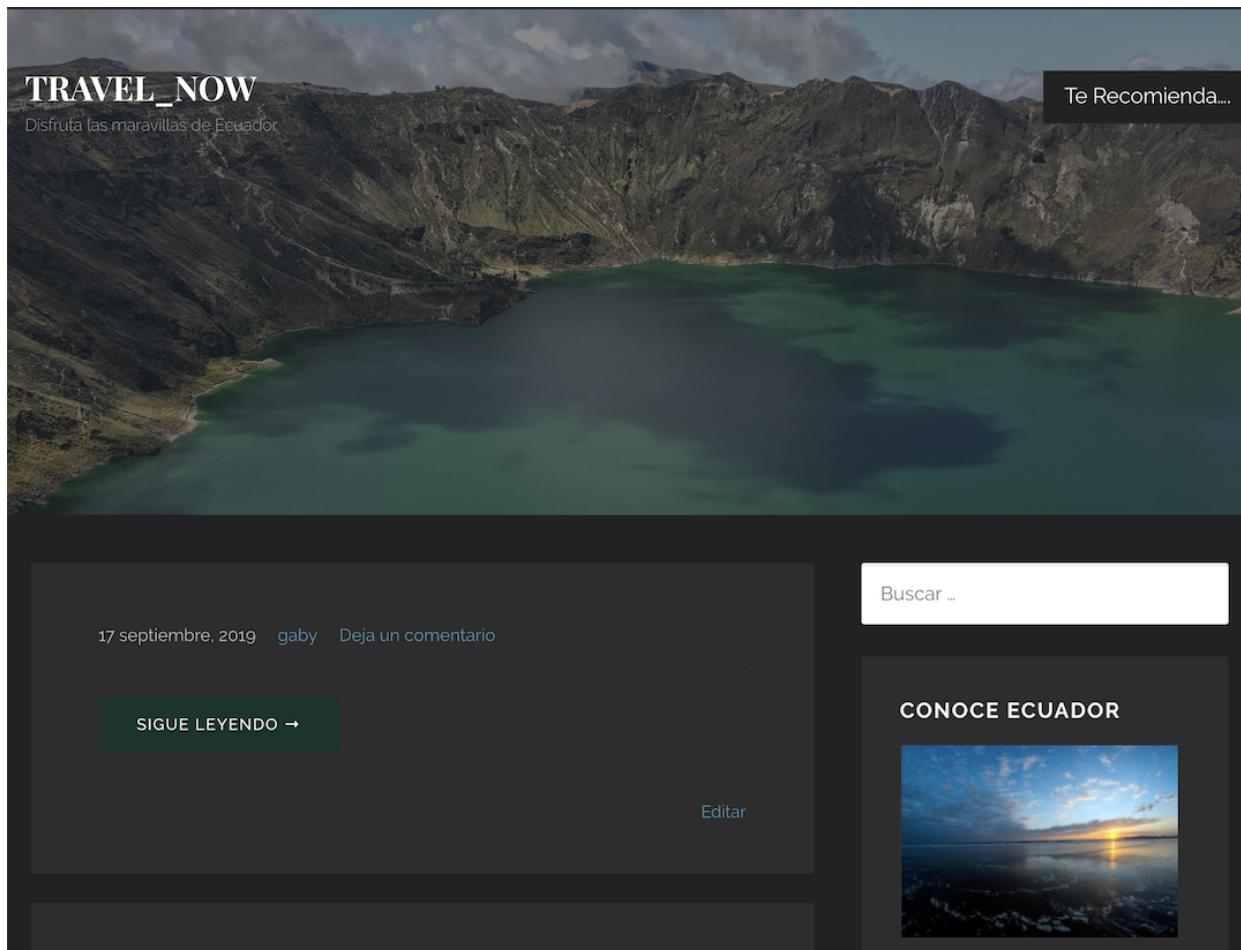


Figura 14: Pantalla Principal del Sitio Web

Para ingresar las preferencias del usuario solo es necesario hacer clic en el menú “Te Recomienda..” Y se desplegará un formulario con campos demográficos básicos y preferencias de propias para cada las categorías de recomendación básica del algoritmo de FC como muestra la imagen

The screenshot shows a dark-themed web application interface. At the top, a banner reads "TE RECOMIENDA...." over a background image of a mountain lake. To the right is a search bar labeled "Buscar ...". On the left, a large form area contains fields for "Nombre" (Name), "Edad" (Age), "Genero" (Gender with options F, M, O), and "Prefieres viajar por:" (Travel preferences with checkboxes for Negocios, Compras, Cultura, Aventura, Turismo, and Ecología). Below these are questions "Conoces Ecuador?" (Do you know Ecuador?) with options SI or No, and a text area for "Cuentanos tu experiencia" (Tell us about your experience) with a placeholder "Cuentanos tu experiencia". At the bottom is a "Enviar" (Send) button. On the right, a sidebar titled "CONOCE ECUADOR" features a photo of Quilotoa lake and the text "Quilotoa".

Figura 15: Formulario de perfilamiento

Para ingresar las preferencias del usuario solo es necesario hacer clic en el menú “Te Recomienda..” Y se desplegará un formulario con campos demográficos básicos y preferencias de propias para cada las categorías de recomendación básica del algoritmo de FC como muestra la imagen

4.2. Algoritmo de recomendación.

En esta sección se realiza el desarrollo y modelamiento de del prototipo del sistema de recomendación basado en las preferencias del usuario, en base a esto se desarrollará un algoritmo correspondiente al FC basado en memoria como ya se describió en el capítulo 2 más a detalle.

Para encontrar la solución más adecuada para la necesidad del algoritmo de filtrado colaborativo que necesitamos se necesita conocer el proceso que se requiere hacer para conseguir una recomendación. Para esto debemos tomar en cuenta todas fuentes que tenemos y el propósito principal que es la recomendación que nos entregue mejor precisión.

Los algoritmos basados en el FC en base a memoria utilizan varias técnicas que alimentan diferentes partes del proceso de recomendación, entre ellos tenemos K-Veinos, MSD (distancia cuadrática media), similitud de cosenos, correlación de Pearson y SVD que puede usarse en cualquier tipo de sistema de recomendación.

Para poder emplear los algoritmos más adecuados se debe tomar en cuenta el esquema con el que se va a trabajar y en base a qué criterios se realizará la recomendación.

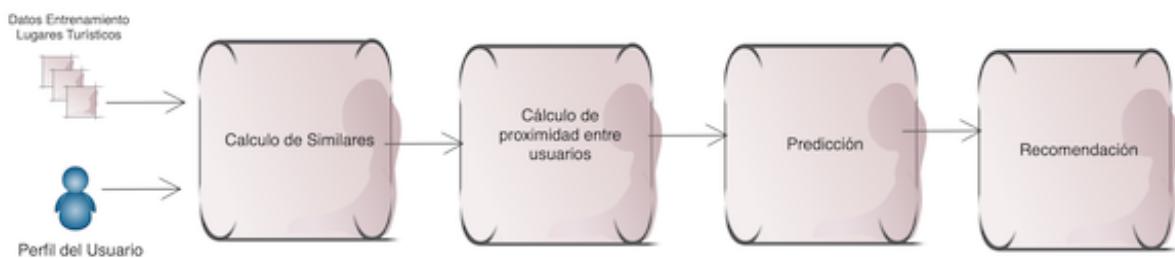


Figura 16: Esquema de funcionamiento de un FC

Figura Esquema Filtrado colaborativo En la figura se puede ver de manera más clara el esquema para generar la recomendación y a continuación se describe el procesado de cada uno de los ítems representados

- Preparación de las fuentes: Consiste en generar las fuentes necesarias para el entrenamiento del modelo que se desea desarrollar, en esta caso, en el capítulo anterior según la metodología con la que estamos trabajando ya se ha preparado las fuentes externas de usuarios y Lugares turísticos. Como ingreso del modelo también se requiere el usuario con el que se va a generar las similitudes y el posterior cálculo de recomendaciones.
- Cálculo de Similares: Con las fuentes listas se puede comenzar con el modelamiento del algoritmo y como punto de partida se realizarán los cálculos para validar las distancias más cortas con las entre el universo que tenemos para entrenar los datos, en este paso se encuentran los algoritmos de similitud por cosenos, MSD.
- Calculo de proximidad entre usuarios: Con los datos de similitud que se ha generado

anteriormente se puede comenzar a probar los algoritmos de proximidad como K-NN y la correlación de Pearson que permiten calcular y ajustar las distancias entre el vector ingresado y los datos de entrenamiento.

- Cálculo de predicciones Con los datos que se ha procesado y ajustado en base a la proximidad entre usuario se puede evaluar y predecir en base a los usuarios que tienen un perfil similar cual podría ser su siguiente compra, iteración o ítem
- Recomendaciones Con la información procesada que se ha generado hasta el momento podemos empezar a ver comportamientos similares, en este punto se debe escoger un número de ítems con los que se va hacer las valoraciones, este número o N se debe evaluar para poder determinar cuál es el que nos generaría un mejor resultado. En este punto se puede generar un top de recomendaciones de acuerdo a la definición que anteriormente se haya hecho para mostrar 5, 10 o el número de sugerencias que el negocio requiera.

4.2.1. Filtrado Colaborativos con retroalimentación Implícita

Una de las principales trabas cuando se realiza un sistema de recomendación en base a usuarios es la dependencia de volúmenes de información generada por el mismo, si no se cuenta con la suficiente data la confianza y precisión del modelo que se esté generando va a bajar considerablemente y no se llegará al propósito. (Liu, Xiang, Zhao y Yang, 2010, B. Wang, Rahimi, Zhou y Wang, 2012)

La retroalimentación explícita está dirigida para este escenario y consiste en recopilar información implícita en la navegación o comportamiento propio del usuario en el sitio, esta información puede ser en base de visitas, compras, iteraciones en el sitio web, toda esta información se guarda como preferencias del usuario pero tiene varios temas que se deben tener presente al momento de implementar este método como es:

- No hay feedback negativo
- Contiene ruido, se requiere mayor depuración para separar movimientos involuntarios del usuario.
- Es complicado cuantificar los cálculos para confianza y precisión del modelo
- No hay muchas métricas de evaluación ya que RMSE y MAE no funcionan bien

Para trabajar con este modelo es necesario hacer referencia al Yifan Hu et al, quien en su paper introduce la variable binaria p_{ui} donde se muestra la preferencia de un usuario sobre un ítem i (<http://yifanhu.net/PUB/cf.pdf>) (Hu, Koren y Volinsky, 2008)

$$p_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases} \quad (6)$$

Lo que indica la fórmula es que si un usuario adquiere un producto es posible que el producto le guste, este concepto no es necesariamente correcta en todos los escenarios y es por eso que Hu et al introduce un indicador de confianza que permita que suba en base a los ítems que asumimos le gusta al usuario. (Hu y col., 2008)

$$c_{ui} = 1 + \alpha r_{ui} \quad (7)$$

Se puede ver en la formula como la Confianza sobre u está regulado por constantemente el incremento de la variable. En caso de que los valores de la variable i sigan subiendo inflando por una acción particular del usuario involuntaria, para esto se preparó una fórmula de confianza logarítmica de la siguiente manera (Hu y col., 2008)

$$c_{ui} = 1 + \alpha \log(1 + r_{ui}/\epsilon) \quad (8)$$

Con los datos que tenemos ya podemos armar las preferencias del usuario, con esta información se puede acceder a la función de costo que se desea minimizar

$$\text{MUII}x^*, B^* \sum_{C^{RS}(B^{RS})} (b^{RS} - x_L^{ST} a^S)_S + y \left(\sum_{s=1}^{st} \|x^{ST}\|_S + \sum_{s=1}^s \|3s^S\|_S \right) \quad (9)$$

Para una investigación más profunda sobre retroalimentación implícita se pueden revisar los siguientes papers Hu y col., 2008, Parra y Amatriain, 2011, Yi, Hong, Zhong, Liu y Rajan, 2014

4.2.2. Análisis Semántico Latente LSA

Es un algoritmo de similitudes que se basa en fórmulas matemáticas para determinar y cuantificar la similitud entre piezas similares, es un método automático de recuperación de información que cuenta con la descomposición de valores singulares SVD como parte de su funcionamiento. Este algoritmo calcula la similitud semántica entre dos elementos analizados de contexto verbal.

La manera que trabaja LSA es sumando las interacciones usadas con las palabras para formar una ecuación que será resuelta por SVD

4.2.3. Descomposición de Valores Singulares SVD

Consiste en utilizar funciones lineales para ir descomponiendo en tres matrices nuevas la matriz de mayos variables y de este modo descubrir los valores singulares. El objetivo principal de esta técnica es ir despejando los valores que considera ruido creando una distribución normal a la frecuencia de datos

$$U = AVS^{-1} \quad (10)$$

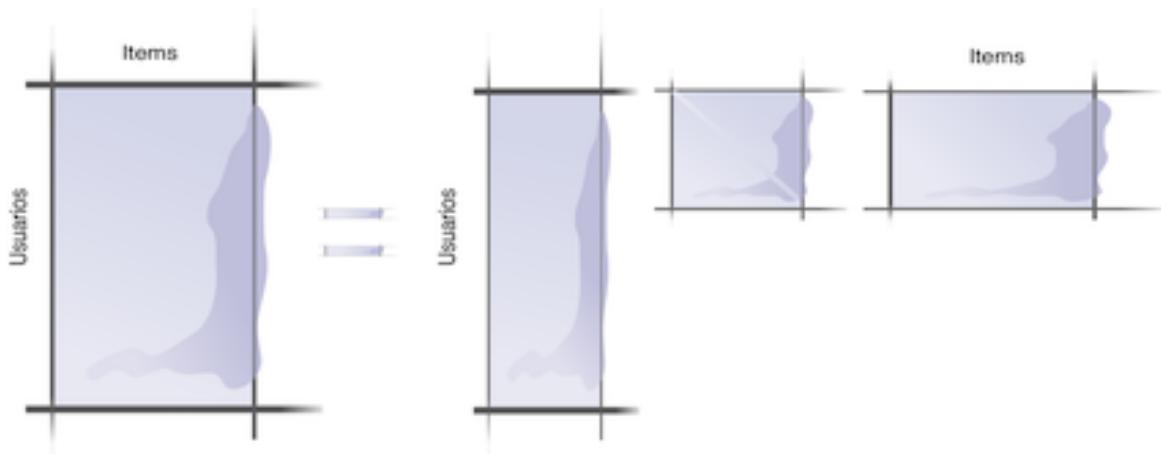


Figura 17: Descomposición de Valores Singulares

La figura 17 indica el proceso de descomposición de valores donde la Matriz [Usuario - Item] se descompone en el Vector[Usuarios] * Matriz[Datos]* Vector[Items]

4.2.4. Redes Bayesianas

Permite identificar patrones por medio de un modelo probabilístico que representa un conjunto de variables aleatorias y dependencias de un nodo también llamado grafo, posee un motor de actualización de probabilidades que hace que sea un potente estimador de probabilidades en base a nuevas evidencias.

$$P(G, S, R) = P(G|S, R)P(S|R)P(R) \quad (11)$$

$$P(R = T|G = T) = \frac{P(G = T, R = T)}{P(G = T)} = \frac{\sum_{S \in \{T, F\}} P(G = T, S, R = T)}{\sum_{S, R \in \{T, F\}} P(G = T, S, R)} \quad (12)$$

4.3. Tecnología utilizada en el desarrollo

A continuación, se presentará como está desarrollado el algoritmo y la tecnología que sustenta su manejo.

Como lenguaje de programación se utiliza Python, que consiste en un lenguaje interpretado de fácil y legible acceso, la versión con la que se realizó el desarrollo es la 2.7, pese a que se encuentra ya desplegada la versión 3.7 no se utilizó por problemas propios con el manejo de los paquetes seleccionados para este desarrollo

Como librería principal para el manejo de la aplicación de recomendación se utiliza Implicit, es una librería de FC del tipo Implicit Feedback que consiste en el análisis de información implícita para validar preferencias de usuario, en base a esta librería utilizaremos dos principales métodos de análisis enfocados en Maquinas factorización y de la misma manera para la predicción de datos y recomendación que se evaluará los resultados en el siguiente capítulo

Para el procesamiento de la evaluación del resultado se utiliza la librería Numpy, esta nos permite manejar vectores y matrices y operaciones del tipo algebraico y de análisis. Este algoritmo se utiliza para la evaluación del modelo.

5. Análisis y Evaluacion de resultados

Este capítulo se realizará pruebas que permiten ver la funcionalidad y precisión del momento al momento de predecir los valores que han sido entrenados con el algoritmo (Bai, Wen, Zhang y Zhao, 2017, Z. Wang, Li, Liu, Liu y Yin, 2015,Zhang y col., 2016, Stadler y Frensch, 1998)

5.1. Algoritmo de recomendación

5.1.1. Ingreso de Conjuntos de datos

Datos para entrenar el modelo

```
df_train = pd.read_csv('/Applications/Documentos/Maestria/TFM/Dataset/Training.csv',
                      sep=',',
                      names=['ID_Usuario','ID_Lugar','Rating','id_pk'],
                      header=None)
df_train.head() #Revisar un resumen de la tabla
```

Figura 18: Entrenamiento: Ingresar datos

La salida de los datos se ve de la siguiente manera

```
Out[64]:
      ID_Usuario  ID_Lugar  Rating      id_pk
0            1        21      2  US000001LT21
1            1        33      1  US000001LT33
2            1        66      2  US000001LT66
3            1        68      5  US000001LT68
4            1        71      3  US000001LT71
```

Figura 19: Entrenamiento: Muestra de Datos

Se realiza una inspección a los datos de entrenamiento se puede ver

```
In [65]: df_train.describe()#Inspección sobre estos
Out[65]:
   ID_Usuario      ID_Lugar     Rating
count  910641.000000  910641.000000  910641.000000
mean   47364.078721    57.506492   3.032718
std    33349.407249    32.930932   1.414016
min    1.000000       1.000000   1.000000
25%   17311.000000    29.000000   2.000000
50%   42328.000000    58.000000   3.000000
75%   76525.000000    86.000000   4.000000
max   110827.000000   114.000000  5.000000
```

Figura 20: Entrenamiento:: Información de los campos de ingresados

Ingreso de matriz Lugares turísticos y tipo de Lugar

```
df_items = pd.read_csv('/Applications/Documentos/Maestria/TFM/Dataset/Lugar_tipo.csv',
                      sep=',',
                      index_col=0,
                      names = columns,
                      header=None,
                      encoding='latin-1')
df_items.head()
```

Figura 21: Items: Ingresar datos de lugar y tipo

La salida de los datos se ven de la siguiente manera:

ID_Lugar		Nombre	Provincia	Ciudad	...	Selva	Volcan	Zoologico
1	Rio Santa Barbara	Azuay	Gualaceo	...	0	0	0	
2	Represa de Paute	Azuay	Cuenca	...	0	0	0	
3	El Turi	Azuay	Cuenca	...	0	0	0	
4	Cascada Irha	Azuay	Cuenca	...	0	0	0	
5	Parque Nacional Cajas	Azuay	Cuenca	...	0	0	0	

[5 rows x 37 columns]

Figura 22: Items: Muestra de Datos de lugar

Se realiza una inspección a los datos Lugares de Interés cargados

```
In [68]: df_items.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 114 entries, 1 to 99
Data columns (total 37 columns):
Nombre           114 non-null object
Provincia        114 non-null object
Ciudad           114 non-null object
Acuario          114 non-null int64
Archipiélago     114 non-null int64
Atractivo Turístico 114 non-null int64
```

Figura 23: Items: Información de los datos cargados

Datos para validar el modelo

```
df_test = pd.read_csv('/Applications/Documentos/Maestria/TFM/Dataset/Test.csv',
                     sep=',',
                     names=['ID_Usuario', 'ID_Lugar', 'Rating', 'id_pk'],
                     header=None)
```

Figura 24: Evaluación: Ingresar datos de validación

La salida de los datos se ven de la siguiente manera:

```
...: df_test.head()
Out[69]:
   ID_Usuario  ID_Lugar  Rating      id_pk
0            1       38      5  US000001LT38
1            1       47      2  US000001LT47
2            1       81      1  US000001LT81
3            1      103      3  US000001LT103
4           16       58      5  US000016LT58
```

Figura 25: Evaluación: Muestra de datos para validación

Se realiza una inspección a los datos de validación que cargamos

```
In [70]: df_test.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 448051 entries, 0 to 448050
Data columns (total 4 columns):
ID_Usuario    448051 non-null int64
ID_Lugar      448051 non-null int64
Rating        448051 non-null int64
id_pk         448051 non-null object
dtypes: int64(3), object(1)
```

Figura 26: Script: Ingresar datos de entrenamiento

Para el desarrollo es necesario que se realice una matriz de Usuarios x Lugares, este proceso se realiza de la siguiente manera:

```
#construcción de la matriz de usuarios x Lugares
user_items_test = {}

for row in test.itertuples():
    if row[1] not in user_items_test:
        user_items_test[row[1]] = []

    user_items_test[row[1]].append(row[2])
```

Figura 27: Matriz: Generar matriz de Usuario y Lugares Turísticos

Muestra de los datos generados.

```
780: [63, 79, 86, 99, 110],
781: [1, 5, 11, 57, 79, 80, 83, 84, 114],
782: [1, 31, 33, 35, 71],
783: [4, 47, 49, 55, 64, 75],
784: [17, 33, 60, 87, 92, 99],
785: [2, 14, 25, 59, 80, 105, 110, 113],
786: [59, 69, 97, 99, 101, 112, 114],
787: [22, 34, 36, 78, 82, 86, 90, 113],
788: [11, 49, 63, 74, 84, 107],
```

Figura 28: Matriz: Muestra de datos desplegados

5.1.2. Métricas

Con los datos cargados se puede empezar a construir las métricas para el funcionamiento de los modelos, a este script no se le ha realizado ningún cambio. En el script se generan los principales cálculos de índices de precisión y evaluación, posteriormente en la ejecución de los modelos será necesario llamar a estos valores

```

def precision_at_k(r, k):
    assert k >= 1
    r = np.asarray(r)[:k] != 0
    if r.size != k:
        raise ValueError('Relevance score length < k')
    return np.mean(r)

def average_precision(r):
    r = np.asarray(r) != 0
    out = [precision_at_k(r, k + 1) for k in range(r.size) if r[k]]
    if not out:
        return 0.
    return np.mean(out)

def mean_average_precision(rs):
    return np.mean([average_precision(r) for r in rs])

def dcg_at_k(r, k):
    r = np.asfarray(r)[:k]
    if r.size:
        return np.sum(np.subtract(np.power(2, r), 1) / np.log2(np.arange(2, r.size + 2)))
    return 0.

def ndcg_at_k(r, k):
    idcg = dcg_at_k(sorted(r, reverse=True), k)

    if not idcg:
        return 0.
    return dcg_at_k(r, k) / idcg

```

Figura 29: Métricas: Definición de las cálculos

5.1.3. Procesamiento de datos

En el procesamiento que se necesita generar se realiza una matriz cruzada resultante de la combinación de la matriz usuario – Lugares turísticos y la matriz de tipo de lugares turísticos

```
#procesamiento de datos
user_items = {}
itemset = set()

for row in train.itertuples():
    if row[1] not in user_items:
        user_items[row[1]] = []

    user_items[row[1]].append(row[2])
    itemset.add(row[2])

itemset = np.sort(list(itemset))

sparse_matrix = np.zeros((len(user_items), len(itemset)))

for i, items in enumerate(user_items.values()):
    sparse_matrix[i] = np.isin(itemset, items, assume_unique=True).astype(int)

matrix = sparse.csr_matrix(sparse_matrix.T)

user_ids = {key: i for i, key in enumerate(user_items.keys())}
user_item_matrix = matrix.T.tocsr()
```

Figura 30: Datos: Definición matriz completa usuarios y Lugares de Interés

La matriz resultante se puede ver resumida en la siguiente captura de pantalla, la matriz completa no es posible visualizar en la consola.

```
In [22]: user_item_matrix
Out[22]:
<110731x114 sparse matrix of type '<type 'numpy.float64'>'>
with 910641 stored elements in Compressed Sparse Row format>

In [23]: sparse_matrix
Out[23]:
array([[0., 0., 0., ..., 0., 0., 1.],
       [0., 0., 0., ..., 0., 1., 0.],
       [1., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [1., 0., 0., ..., 0., 0., 0.]])
```

Figura 31: Datos: Muestra matriz completa usuarios y Lugares de Interés

5.1.4. Evaluación del Modelo

Con las matrices listas para el procesamiento se comienza a definir las clases que se van a utilizar en el modelo, en la captura de la figura xx se puede ver las métricas que estamos definiendo para evaluar el modelo

```

def evaluate(model, n):
    mean_map = 0.
    mean_ndcg = 0.
    for u in user_items_test.keys():
        rec = [t[0] for t in model.recommend(u, user_item_matrix, n)]
        rel_vector = [np.isin(user_items_test[u], rec, assume_unique=True).astype(int)]
        mean_map += mean_average_precision(rel_vector)
        mean_ndcg += ndcg_at_k(rel_vector, n)

    mean_map /= len(user_items_test)
    mean_ndcg /= len(user_items_test)

    return mean_map, mean_ndcg

```

Figura 32: Evaluación: Definición y proceso de evaluación

5.1.5. Modelo de Recomendación

Se definen las clases para el sistema de recomendación que se evaluarán y la clase para el despliegue del resultado del modelo

```

def show_recommendations(model, user, n):
    recommendations = [t[0] for t in model.recommend(user, user_item_matrix, n)]
    return items.loc[recommendations]['Nombre']

def show_similar_places(model, item, n=10):
    sim_items = [t[0] for t in model.similar_items(item, n)]
    return items.loc[sim_items]['Nombre']

```

Figura 33: Recomendación: Definición del proceso de recomendación

Se define el valor del factor e interacciones que el modelo debe tomar y se entrena los datos en el modelo ALS Análisis Semántico Latente

```

# Definimos y entrenamos el modelo ALS
model_als = implicit.als.AlternatingLeastSquares(factors=450, iterations=10)
model_als.fit(matrix)

show_recommendations(model_als, user=40, n=10)

maprec, ndcg = evaluate(model_als, n=10)
print('map: {}\nndcg: {}'.format(maprec, ndcg))

```

Figura 34: Recomendación: Ejecución y entrenamiento con ALS

Muestra de recomendaciones en base a ALS

```
ID_Lugar
53          Valle Del Chota
51          Yahuarcocha
44          El Faro
55          Parque Podocarpus
30          Balneario La Chocha
41          Isla Bartolome
39          Isla Santa Cruz
38          Playa Las Penas
37          Lago De Muisne
36  Reserva Ecologica Cotacachi Cayapas
Name: Nombre, dtype: object
```

Figura 35: Recomendación: Sugerencias en base a ALS

Se define el valor del factor e interacciones que el modelo debe tomas y se entrena los datos en el modelo BPR Redes Bayeristas

```
# Definimos y entrenamos el modelo BPR
model_bpr = implicit.bpr.BayesianPersonalizedRanking(factors = 450,iterations=35)
model_bpr.fit(matrix)

show_recommendations(model_bpr, user=40, n=10)

maprec, ndcg = evaluate(model_bpr, n=10)
print('map: {}\nndcg: {}'.format(maprec, ndcg))
```

Figura 36: Recomendación: Ejecución y entrenamiento con BPR

Muestra de recomendaciones en base a BPR

```
ID_Lugar
85          Dique De Mera
70          Valle Del Rio Upano
68          Cascada Lucy
58          Plaza De La Independencia
82          Parque Acuatico Montero Puyo
62          Zoo el Pantanal
49          Otavalo
97          Santo Domingo
3           El Turi
88          Mital Del Mundo
Name: Nombre, dtype: object
```

Figura 37: Recomendación: Sugerencias en base a BPR

En base a los valores de evaluación que generó el sistema podemos verificar cual es el algoritmo con mayor predicción y fiable y se enviara esa recomendación

```
show_similar_places(model_bpr,item=40,n=10)
```

Figura 38: Recomendación: Ejecución de la Sugerencia al usuario

Recomendación para el usuario

```
ID_Lugar
40           Isla Isabela
89           Parque Nacional Ilinizas
48   Museo Historico De Abdón Calderon
41           Isla Bartolome
79           Rio Napo
49           Otavalo
69           Rio Morona
6   Reserva faunistica el Chimborazo
61           Playas Del Salto
19           Reserva Awa
Name: Nombre, dtype: object
```

Figura 39: Recomendación: Sugerencias para el usuario

5.2. Evaluación de Resultados

5.2.1. Método de evaluación

En esta sección se realizará un análisis de los datos entrenados y los resultados que entrega el algoritmo, previamente se explicará el método de evaluación se está empleando.

Método de evaluación basado en Precisión y recuperación: Son métricas que se utilizan para sacar indicadores que muestran análisis de los resultados obtenidos, en este caso se revisará las métricas de precisión y el alcance o recuperación.

La precisión consiste en tratar de encontrar el valor más bajo, lo que determina que es modelo es preciso, para realizar conseguir este indicador es necesario que analizar los se tome en cuenta los casos que se han clasificado correctamente contra el total de casos bien segmentados.

$$Precision = \frac{TP}{TP + FP} \quad (13)$$

La recuperación, proporciona el valor porcentual correspondiente a el valor de los ítems que fueron correctamente clasificados divididos al total de suma de verdaderos positivos y falsos negativos.

$$Recall = \frac{TP}{TP + FN} \quad (14)$$

5.2.2. Resultados de la evaluación

Se realizaron dos escenarios para la validación de los datos y recomendaciones en base a los dos algoritmos empleados en el desarrollo.

Ejecución de ambos modelos con diferentes valores en interacciones y factores

ALS:

```
map: 0.177079399397
ndcg: 0.321592649311
```

Figura 40: Evaluación: Resulpados con el algoritmo ALS

BPR

```
map: 0.0279031496764
ndcg: 0.243491577335
```

Figura 41: Evaluación: Resulpados con el algoritmo BPR

Para poder ver la precisión de los dos algoritmos se realiza una ejecución con factores e interacciones iguales.

```
In [25]: model_als = implicit.als.AlternatingLeastSquares(factors=450, iterations=35)
....: model_als.fit(matrix)
HBox(children=(IntProgress(value=0, max=35), HTML(value=u'')))

In [26]: maprec, ndcg = evaluate_model(model_als, n=10)
....: print('map: {}\\nndcg: {}'.format(maprec, ndcg))
map: 0.00740326048949
ndcg: 0.169984686064

In [27]: model_bpr = implicit.bpr.BayesianPersonalizedRanking(factors = 450,iterations=35)
....: model_bpr.fit(matrix)
HBox(children=(IntProgress(value=0, max=35), HTML(value=u'')))

In [28]: maprec, ndcg = evaluate_model(model_bpr, n=10)
....: print('map: {}\\nndcg: {}'.format(maprec, ndcg))
map: 0.0272049991418
ndcg: 0.246554364472
```

Figura 42: Evaluación: Ejecución de los algoritmos ALS y BPR

5.2.3. Conclusiones de la evaluación

En base a los datos analizados se puede determinar que mientras más altos son los factores que se toma en cuenta la precisión es mucho más baja

ALS

Nro Factores	100	450
map	0.177	0.007
ndcg	0.321	0.169

BPR

Nro Factores	100	450
map	0.177	0.007
ndcg	0.321	0.169

Cuadro 6: Resultados de ejecución de la evaluación

A pesar de que el algoritmo ALS tiene un menor grado de falla, el modelo se demora mucho más en dar la recomendación que el algoritmo BPR.

6. Conclusiones

Para que el modelo de filtrado colaborativo funcione de una manera optima es necesario aumentar la magnitud de los datos y ejecutar el proceso de evaluación nuevamente para confirmar los indicadores de efectividad

Para poder entregar un sitio web atractivo y comodo para el usuario es necesario crear una estructura más gráfica, para la memoria se desplegaron un banco de imágenes para el sitio que falta incorporar

El presente desarrollo me permitió conocer más fondo la metodología CRISP DM y cada una de sus etapas, implementar la solución en base a la misma fue mucho más dinámico y ordenado.

El desarrollo de todo el proyecto para el fin de master ha sido una experiencia muy enriquecedora, me permitió ver todos los problemas, fases, y soluciones y me preparó para comenzar cosas nuevas con los conocimientos adquiridos.

7. Trabajo futuro

Se considera realizar una segunda versión del desarrollo presentado pero con algoritmos de filtrado colaborativo híbridos que interactúen y de esta manera crear un mejor sistema de recomendaciones de sitios turísticos y de interés

Para el sitio web se considera utilizar un gestor de contenido basado en phyton y cambiar la infraestructura actual para fortalecer el sitio y optimizar los tiempos de respuesta

Gestionar por medio de los entes de control información estadística de hoteles y sitios de interés como hospitales para permitir que el sitio tenga una guía completa para el turista.

Referencias

- Cover, T. y Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1), 21-27.
- Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, (4), 325-327.
- Keller, J. M., Gray, M. R. y Givens, J. A. (1985). A fuzzy k-nearest neighbor algorithm. *IEEE transactions on systems, man, and cybernetics*, (4), 580-585.
- Pearl, J. (1994). A probabilistic calculus of actions. En *Uncertainty Proceedings 1994* (págs. 454-462). Elsevier.
- Denoeux, T. (1995). A k-nearest neighbor classification rule based on Dempster-Shafer theory. *IEEE transactions on systems, man, and cybernetics*, 25(5), 804-813.
- Stadler, M. A. y Frensch, P. A. (1998). *Handbook of implicit learning*. Sage Publications, Inc.
- Baeza-Yates, R., Ribeiro-Neto, B. y col. (1999). *Modern information retrieval*. ACM press New York.
- Pazzani, M. J. (1999). A framework for collaborative, content-based and demographic filtering. *Artificial intelligence review*, 13(5-6), 393-408.
- Burke, R. (2000). Knowledge-based recommender systems. *Encyclopedia of library and information systems*, 69(Supplement 32), 175-186.
- Herlocker, J. L., Konstan, J. A., Terveen, L. G. y Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 5-53.
- Ordóñez, M. (2005). *Políticas de empleo en la planificación turística local de Ecuador: herramientas para su formulación*. United Nations Publications.
- Césari, M. (2006). Nivel de Significación Estadística para el Aprendizaje de una Red Bayesiana. *Trabajo Final de Especialidad en Tecnologías de Explotación de Información. Escuela de Postgrado. Instituto Tecnológico de Buenos Aires*.
- Ballesteros, E. R. y Carrión, D. S. (2007). *Turismo comunitario en Ecuador: desarrollo y sostenibilidad social*. Editorial Abya Yala.
- Nieto, S. M. G. (2007). Filtrado colaborativo y sistemas de recomendación. *Inteligencia en Redes de Comunicaciones. Madrid*.
- Hu, Y., Koren, Y. y Volinsky, C. (2008). Collaborative filtering for implicit feedback datasets. En *2008 Eighth IEEE International Conference on Data Mining* (págs. 263-272). Ieee.

- Kaufman, L. y Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons.
- Pollo Cattaneo, M. F., Britos, P., Pesado, P. M. y García Martínez, R. (2009). Metodología para especificación de requisitos en proyectos de explotación de información. En *XI Workshop de Investigadores en Ciencias de la Computación (San Juan, Argentina)* (Vol. 11).
- Seguido Font, M. (2009). *Sistemas de recomendación para webs de información sobre la salud* (Tesis de mtria., Universitat Politècnica de Catalunya).
- Cobos, C., Zuñiga, J., Guarin, J., León, E. y Mendoza, M. (2010). CMIN-herramienta case basada en CRISP-DM para el soporte de proyectos de minería de datos. *Ingenieria e investigación*, 30(3), 45-56.
- Liu, N. N. [Nathan N], Xiang, E. W., Zhao, M. y Yang, Q. (2010). Unifying explicit and implicit feedback for collaborative filtering. En *Proceedings of the 19th ACM international conference on Information and knowledge management* (págs. 1445-1448). ACM.
- Parra, D. y Amatriain, X. (2011). Walk the talk. En *International Conference on User Modeling, Adaptation, and Personalization* (págs. 255-268). Springer.
- Prieto, M. (2011). Los estudios sobre turismo en Ecuador. *Espacios en disputa: el turismo en Ecuador*, 9-28.
- Wang, B., Rahimi, M., Zhou, D. y Wang, X. (2012). Expectation-Maximization collaborative filtering with explicit and implicit feedback. En *Pacific-Asia Conference on Knowledge Discovery and Data Mining* (págs. 604-616). Springer.
- Abreu-Lee, Y., Infante-Abreu, M. B., Delgado-Fernández, T. y Delgado-Fernández, M. (2013). Modelo de vigilancia tecnológica apoyado por recomendaciones basadas en el filtrado colaborativo. *Ingenieria Industrial*, 34(2), 167-177.
- Castellanos, Y. R. (2014). Sistema de recomendación por filtrado colaborativo para el sistema de publicación de contenido multimedia-VideoWeb 1.0 [Recommender system using collaborative filtering for the publication system of multimedia content-VideoWeb 1.0]. *International Journal of Innovation and Applied Studies*, 6(3), 326-334.
- Yi, X., Hong, L., Zhong, E., Liu, N. N. [Nanthan Nan] y Rajan, S. (2014). Beyond clicks: dwell time for personalization. En *Proceedings of the 8th ACM Conference on Recommender systems* (págs. 113-120). ACM.
- Wang, Z., Li, Q., Liu, Y., Liu, W. y Yin, J. (2015). Online personalized recommendation based on streaming implicit user feedback. En *Asia-Pacific Web Conference* (págs. 720-731). Springer.

- Monge, J. G. y Perales, R. M. Y. (2016). El desarrollo turístico sostenible. Tren Crucero del Ecuador. *Estudios y perspectivas en turismo*, 25(1), 57-72.
- Ramírez, O., Paúl, H., Véliz, N., Tania, I., Roldán Ruenes, A. y Ferrales Arias, Y. (2016). Emprendimiento como factor del desarrollo turístico rural sostenible. *Retos de la Dirección*, 10(1), 71-93.
- Zhang, F., Gong, T., Lee, V. E., Zhao, G., Rong, C. y Qu, G. (2016). Fast algorithms to evaluate collaborative filtering recommender systems. *Knowledge-Based Systems*, 96, 96-103.
- Bai, T., Wen, J.-R., Zhang, J. y Zhao, W. X. (2017). A neural collaborative filtering model with interaction-based neighborhood. En *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (págs. 1979-1982). ACM.

8. Anexos

Algoritmo para bajar información desde Twitter:

```
import couchdb
from tweepy import Stream
from tweepy import OAuthHandler
from tweepy.streaming import StreamListener
import json

#API T-for-health-7#####
ckey = csecret = atoken = asecret =
#####

class listener(StreamListener):

    def on_data(self, data):
        dictTweet = json.loads(data)
        try:
            dictTweet["_id"] = str(dictTweet['id'])
            texto = dictTweet["text"]
            if 'cosecha' in texto:
                doc = db.save(dictTweet)
                print "SAVED- str(doc) +"
                print str(data)
        except:
            print "already exists"
        return True

    def on_error(self, status):
        print status

    auth = OAuthHandler(ckey, csecret)
    auth.set_access_token(atoken, asecret)
    twitterStream = Stream(auth, listener())

    """=====couchdb====="""
    server = couchdb.Server('http://192.168.100.223.:5984/')
    try:
        db = server.create('vacaciones')
```

Algoritmo utilizado para las recomendaciones implicit_als_bpr

```

import pandas as pd
import numpy as np
import implicit
import scipy.sparse as sparse

# Carga el set de datos de entrenamiento df_train = pd.read_csv('/Applications/Documentos/Maestria/TFM/
sep=', names=['ID_Usuario','ID_Lugar','Rating','id_pk'], header=None) df_train.head() #Revi-
sar un resumen de la tabla df_train.describe()#Inspección sobre estos datos (.info())
columns = ['ID_Lugar','Nombre','Provincia','Ciudad','Acuario','Archipielago','Atractivo Turistico',
'Balneario','Bosque','Casacada','Cascada','Caverna','Cementerio','Cerro','Complejo Arqueolo-
gico ','Cueva','Iglesia','Isla','Lago','Laguna','Montaña','Museo','Parque','Parque Acuatico','Parque
Nacional','Playa','Puerto','Represa','Reserva','Reserva Biantropológica','Reserva Ecologica','Reserva
Faunistica', 'Rio','Santuario Nacional','Santuario Nacional','Selva','Volcan','Zoologico']
# Carga el set de datos de de Lugar_tipo df_items = pd.read_csv('/Applications/Documentos/Maestria/TFM/
sep=', index_col=0, names = columns, header=None, encoding='latin-1') df_items.head()
df_items.info()

#Carga el set de datos de entrenamiento pruebas df_test = pd.read_csv('/Applications/Documentos/M
sep=', names=['ID_Usuario','ID_Lugar','Rating','id_pk'], header=None)

df_test.head() df_test.info()

user_items_test = #crear matriz de usuario_test for row in df_test.itertuples(): if row[1]
not in user_items_test: user_items_test[row[1]] = []
user_items_test[row[1]].append(row[2])
# Definicion de métricas (No editar) # Obtenido de https://gist.github.com/bwhite/3726239
def precision_at_k(r, k): assert k >= 1 r = np.asarray(r)[:k] != 0 if r.size != k: raise ValueE-
rror('Relevance score length <k>') return np.mean(r)

def average_precision(r): r = np.asarray(r) != 0 out = [precision_at_k(r, k + 1) for k in
range(r.size) if r[k]] if not out: return 0. return np.mean(out)

def mean_average_precision(rs): return np.mean([average_precision(r) for r in rs])

def dcg_at_k(r, k): r = np.asarray(r)[:k] if r.size: return np.sum(np.subtract(np.power(2, r),

```

```

1) / np.log2(np.arange(2, r.size + 2))) return 0.

def ndcg_at_k(r, k): idcg = dcg_at_k(sorted(r, reverse=True), k)

    if not idcg: return 0. return dcg_at_k(r, k) / idcg

#dProcesamiento de datos user_items = itemset = set()

    for row in df_train.itertuples(): if row[1] not in user_items: user_items[row[1]] = []

        user_items[row[1]].append(row[2]) itemset.add(row[2])

    itemset = np.sort(list(itemset))

    sparse_matrix = np.zeros((len(user_items), len(itemset)))

    for i, items in enumerate(user_items.values()): sparse_matrix[i] = np.isin(itemset, items,
assume_unique=True).astype(int)

    matrix = sparse.csr_matrix(sparse_matrix.T)

    user_ids = key: i for i, key in enumerate(user_items.keys()) user_item_matrix = ma-
trix.T.tocsr()

    #Define modelo de Evaluacion con AP def evaluate_model(model, n): mean_map =
0. mean_ndcg = 0. for u in user_items_test.keys(): rec = [t[0] for t in model.recommend(u,
user_item_matrix, n)] rel_vector = [np.isin(user_items_test[u], rec, assume_unique=True).astype(int)]
mean_map += mean_average_precision(rel_vector) mean_ndcg += ndcg_at_k(rel_vector, n)

    mean_map /= len(user_items_test) mean_ndcg /= len(user_items_test)

    return mean_map, mean_ndcg

#Define modelo de recomendacion def show_recommendations(model, user, n): recommenda-
tions = [t[0] for t in model.recommend(user, user_item_matrix, n)] return df_items.loc[recommendations][['No

#Define datos recomendacion def show_similar_items(model, item, n=10): sim_items = [t[0] for
t in model.similar_items(item, n)] return df_items.loc[sim_items]['Nombre']

# Definimos y entrenamos el modelo ALS model_als = implicit.als.AlternatingLeastSquares(factors=100,
iterations=10) model_als.fit(matrix)

    show_recommendations(model_als, user=40, n=10)

#Evaluar los datos utilizados maprec, ndcg = evaluate_model(model_als, n=10) print('map: :

```

```
'.format(maprec, ndcg))  
# Definimos y entrenamos el modelo BPR model_bpr = implicit.bpr.BayesianPersonalizedRanking(factors  
= 100,iterations=10) model_bpr.fit(matrix)  
  
show_recommendations(model_bpr, user=40, n=10)  
#Evaluar los datos utilizados maprec, ndcg = evaluate_model(model_bpr, n=10) print('map: :  
'.format(maprec, ndcg))  
  
show_similar_items(model_bpr,item=40,n=10)
```