

Sistema recomendador de casas basado en los salarios de distintos usuarios

De Gracia González, Jesús Aquiles	Ortega Coronado, Yeny	Obaldia Ruiz, Kaiser Javier
Facultad de Informática, Electrónica y Comunicaciones.	Facultad de Informática, Electrónica y Comunicaciones.	Facultad de Informática, Electrónica y Comunicaciones.
Escuela de Informática.	Escuela de Informática.	Escuela de Informática.
Ciudad de Panamá, Panamá	Ciudad de Panamá, Panamá	Ciudad de Panamá, Panamá
jaquideg08@gmail.com	yennycoronado199727@gmail.com	thekaiser26@gmail.com

Abstract— Recommendation systems are one of the main areas of research today. In fact applications like youtube, netflix, amazon, etc. They use these systems to study the preferences that users have when browsing their content. But, these systems are also applied to solve problems that a person faces in daily life, for example, there is choosing a home that meets the desired expectations of the person. This involves a series of variables that are adapted to a real scenario. That is why this article will be talking about the different results obtained with the creation of a house recommender system in Panama City, applying machine learning techniques that allow the system to be properly trained. Programs were also used to obtain data from the different dwellings (new or second) in the city. Finally, results were achieved that are complemented by user salaries.

RESUMEN

Los sistemas de recomendaciones son una de las principales áreas de investigación actualmente. De hecho aplicaciones como youtube, netflix, amazon, etc. Utilizan estos sistemas para estudiar las preferencias que tienen los usuarios navegando en su contenido. Pero, estos sistemas se aplican también para resolver problemas que afronta una persona en la vida cotidiana, por ejemplo está la de escoger una vivienda que cumpla con las expectativas deseadas de la persona. Esto conlleva una serie de variables que se adapten a un escenario real. Por eso en este artículo se estará conversando acerca de los diferentes resultados obtenidos con la creación de un sistema recomendador de casas en la ciudad de Panamá, aplicando técnicas de machine learning que permita entrenar correctamente el sistema. También se utilizaron programas para obtener los datos de las diferentes

viviendas(nuevas o de segundas) en la ciudad. Finalmente se consiguieron resultados que se complementan con los salarios de los usuarios.

PALABRAS CLAVES

Sistema recomendador, dataset, técnica, machine learning, expectativa, filtrado colaborativo y vivienda.

I - INTRODUCCIÓN

Los sistemas de recomendación están diseñados para proporcionar a los usuarios por medio de un análisis de preferencias pasadas.

Como describiremos más adelante, la idea principal en este proyecto es que las personas a menudo obtienen las mejores recomendaciones. El sistema de recomendación utiliza varias técnicas para hacer coincidir a personas con intereses similares y hacer recomendaciones basadas en intereses compartidos. como dato principal el rango salarial establecido, siendo nuestro dato primordial para llevar a cabo el desarrollo de este trabajo

II - OBJETIVO

Desarrollar e implementar un sistema recomendador aplicando técnicas de machine learning que permita al usuario encontrar una vivienda que cumpla con sus expectativas, tomando en cuenta la ubicación, número de habitaciones, precio de la vivienda, y sobre todo el salario que posee el usuario para poder aplicar a una determinada casa.

III - MATERIALES Y MÉTODOS

A. Dataset

El conjunto de datos utilizado en el estudio se obtuvo desde una página web llamada Grupo Tribaldos, extrayendo los datos por medio de la aplicación

octoparse.

B. Métodos

Filtrado colaborativo: por medio de esta técnica se realizan diferentes investigaciones y procesos colaborativos sobre las calificaciones de los usuarios o elementos para predecir y generar nuevas recomendaciones para los usuarios. Esta técnica destaca 2 tipos de métodos, basado en modelos y basados en memoria. Para efecto del proyecto utilizaremos el método basado en memoria.

Basado en memoria: consta de un métodos basados en el usuario y/o elementos.

1. Basado en el usuario: este consta de 2 tareas
 1. Encontrar vecinos que sean similares al usuario.

$$\cos(x1, x2) = \frac{x1 \cdot x2}{\|x2\| \Delta \|x1\|}$$

2. Predecir el Rating que el usuario a dará a un ítem j

$$p_{aj} = \bar{v}_a + \alpha \sum_{i=1}^n w(a, i)(v_{ij} - \bar{v}_i)$$

IV - Metodologías

1. Lo primero que hacemos es importar la librería pandas para la recopilación de los datos de las viviendas.
2. Después de recopilar y almacenar los datos, filtramos la información más relevante para hacer las recomendaciones.
3. Luego importamos lo que es numpy para crear un vector con los salarios posibles correspondientes de los usuarios para saber donde podía aplicar .
4. Utilizamos la formula `descuento = (data2['salarios'] * 0.4` para poder saber si el usuario dependiendo del interés aplicaba para una vivienda de segunda o nueva.
5. Después realizamos lo que era una condición FOR y IF para que nos diera los resultado para saber qué vivienda podía aplicar el usuario si en una nueva o una de segunda si pasarse del salario de 3000\$.
6. Por último imprimimos los resultados esperados del sistema recomendador.

V - Análisis de Resultados

I. Recolección de datos:

imagen 1: Datos recolectados por la página Tribaldos extraído por la aplicación ocuparse.

imagen 2: Librería panda donde extrae la información necesario de cada vivienda ya sea nueva o vieja

```
[12] import pandas as pd
data = pd.read_csv('datos/legajo.csv')
```

	Numero	Ubicación	Nombre	Disponibilidad	precio de venta	Precio de Alquiler	securasas	Baños	Dimension	Tipo de vivienda	vivienda nueva /vivienda de segunda
0	2	Caldería	WHITE	VENTA	20000	NaN	2	2	119	APARTAMENTOS	2.0
1	2	Caldería	HQO	VENTA	58000	NaN	3	3	405	APARTAMENTOS	1.0
2	2	Caldería	WHITE	VENTA	33000	NaN	3	3	169	APARTAMENTOS	2.0
3	1	Bella Vista	RIVAGE	VENTA	36000	NaN	2	2	177	APARTAMENTOS	2.0
4	1	Bella Vista	WATERS ON THE BAY	VENTALQUILER	37000	NaN	2	2	122	APARTAMENTOS	2.0
...
379	4	San Francisco	PLAZA REAL	VENTA	26000	NaN	3	2	150	APARTAMENTOS	2.0
371	4	San Francisco	PREMIUM TOWER	VENTA	42000	NaN	3	3	179	APARTAMENTOS	1.0
372	4	San Francisco	PACIFIC BAY	VENTALQUILER	63000	NaN	3	2	238	APARTAMENTOS	1.0
373	4	San Francisco	LATORRACA	VENTA	26000	NaN	3	3	113	APARTAMENTOS	2.0
374	4	San Francisco	TERRAZAS DEL PACIFICO	VENTA	97000	NaN	4	4	370	APARTAMENTOS	1.0

375 rows x 12 columns

imagen 2: generación de datos de la vivienda

La imagen3 se muestra nuestro Dataset donde contiene una data limpia de 100 usuarios obtenidos de manera aleatoria, utilizando métodos y algoritmos de inteligencia artificial y basándonos en datos del sistema inmobiliario Nacional.

```
import pandas as pd
import numpy as np

data2 = pd.read_csv('salario2.csv')
data2
```

	usuarios	salarios
0	1	2369
1	2	2308
2	3	2669
3	4	2699
4	5	1851
...
95	96	2540
96	97	2436
97	98	2540
98	99	2803
99	100	2105

Imagen 3: Salarios posibles

II. Fórmula:

Demostración de la fórmula utilizada para saber los salarios aleatoriamente para saber el descuento de cada usuario donde aplicaría si en una vivienda nueva o vieja

```
[17] from random import choice
descuento = (data2['salarios'] * 0.4)
#print (descuento)
u_aleatorio = choice(descuento)
```

```
[28] ICN = 0.055/12
ICV = 0.015/12
```

II. Resultados Finales:

En la **imagen 4** vemos lo que es el código donde hacemos una sentencia for para que recorra el salario del usuario y entre a una sentencia if la cual entrará a un estado para saber las vivienda que podrá aplicar

```
[40] len(data['vivienda nueva /vivienda de segunda'])
import math
import numpy as np
vec = []
for pos, estado in enumerate(data['vivienda nueva /vivienda de segunda']):

    if estado == 2.0:
        de = (ICV * precioVenta[pos])
        nu1 = 1+ICV
        nu = (1 - math.pow((nu1),-360))
        p = round(de/nu)
        vec.append(p)
        # vec=np.array(p)
        # print("Valor 2.0 ==> {}".format(p))
    else:
        de = (ICN * precioVenta[pos])
        nu1 = 1+ICN
        nu = 1 - math.pow((nu1),-360)
        p1 = round(de/nu)
        vec.append(p1)
        # vec=np.array(p1)
        # print("valor 1.0 ==> {}".format(p1))
```

imagen 4: código

En la imagen 5 nos imprime lo que es el usuario seleccionado con su salario

```
[43] print('Usuario seleccionado: ')
print(u_aleatorio)

Usuario seleccionado:
1148.0
```

imagen 5: usuario seleccionado

Por último en la siguiente **imagen 6** tenemos el sistema recomendador donde nos muestra la diferentes opciones que tiene el usuario.

```
df=pd.DataFrame(data)
comparacion = []
for i in range(374):
    if vec[i] <= u_aleatorio:
        print(df.iloc[i])
        print('')
        comparacion.append(vec[i])
```

Numero	2
Ubicación	Calidonia
Nombre	WHITE
Disponibilidad	VENTA
precio de Venta	250000
Precio de Alquiler	NaN
Recamaras	2
Baños	2
Dimensión	119
Tipo de vivienda	APARTAMENTOS
vivienda nueva /vivienda de segunda	2
Name: 0, dtype: object	
Numero	2
Ubicación	Calidonia
Nombre	BAYFRONT
Disponibilidad	VENTA
precio de Venta	215000
Precio de Alquiler	NaN
Recamaras	1
Baños	1
Dimensión	100
Tipo de vivienda	APARTAMENTOS
vivienda nueva /vivienda de segunda	2
Name: 5, dtype: object	

imagen 6: Sistema recomendador

VI - Conclusiones

Como resultado final logramos que el sistema recomendador lograra la función de que se viera el listado de viviendas en la que podía aplicar un usuario según los requisitos pedido en nuestro laboratorio. Aprendimos a cómo construir un sistema recomendador ya que no sabíamos todos los procedimiento que este tenía, pero al investigar más profundo comprendimos cómo manejarlo, y al final concluimos con éxito nuestro trabajo.

VI - Bibliografías

Sistemas de Recomendación. (27 de August de 2019). Obtenido de

<https://www.aprendemachinelearning.com/sistemas-de-recomendacion/>

Tribalgo, G. (2001-2019). Obtenido de

<https://www.grupotribaldos.com/>

Google Colaboratory. (2020). Martha Roja.

https://colab.research.google.com/drive/15_Xa2dJ5_F8-3VQ80UZHVLsXnrzUKs3e

MERLE. (02 de 09 de 2020). Obtenido de

<https://www.merkleinc.com/es/es/blog/sistemas-recomendacion>

JAYWRKR. (01 de 01 de 2029). medium. Obtenido de

<https://medium.com/@eng.saavedra/sistemas-de-recomendaci%C3%B3n-parte-2-b8a5dc9dc730>

Google Colaboratory. (2020). Martha Roja.

https://colab.research.google.com/drive/15_Xa2dJ5_F8-3VQ80UZHVLsXnrzUKs3e

Python - 5 - Matrices, listas y cadenas de texto - AprendeAProgramar.com. (2020). Python.

<https://www.aprendeaprogramar.com/cursos/verApartado.php?id=7006>