May-June 2020
ISSN: 0193-4120 Page No. 5340-5345

Retail Store - Product Recommendation System

[1]Neha Kumari Singh, [2]Prabhakar.M
[1]MTech Student, [2]Associate Professor, School of Computing and Information Technology,
Reva University, Bangalore, Karnataka, India
[1]kneha9404@gmail.com, [2]prabhakar.m@reva.edu.in

Article Info
Volume 83
Page Number: 5340-5345
Publication Issue:
May-June 2020

Article History
Article Received:*19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *16 May 2020*

Abstract

Usually there are masses of information created every day more over, the nearness of a ton of information makes it scarcely to mining the required information. Personalized recommendation is the strategy to mitigate the issue. Collaborative filtering is one of the most notable advances in the customized recommendation system. As the customer rating matrix ending up being incredibly sparsity, traditional collaborative filtering recommendation figuring registers likeness between things using the rating data, and it doesn't consider the semantic association between different things, thusly recommendation quality is particularly poor. To deal with this issue, this paper combines the thing semantic closeness and the thing rating similarity, which considers the effect of thing semantic and customer rating to improve the things based collaborative filtering.

Keywords: *Collaborative Filtering, Content based Filtering, Recommendation System.*

1. Introduction

A Recommender system or a recommendation system (now and then supplanting 'system' with an equivalent word, for example, stage or motor) is a subclass of data shifting framework that looks to foresee the 'rating' or 'inclination' a client would provide for a thing. They are essentially utilized in business applications.

Recommender system are used in an assortment of territories, and are most normally perceived as playlist generators for video furthermore, music administrations like Netflix, YouTube and Spotify, item recommenders for administrations, for example, Amazon, or substance recommenders for online life stages, for example, Facebook furthermore, Twitter. These system can work utilizing a solitary information, like music, or different contributions inside and across stages like news, books, and search questions. There are additionally well known recommender system for explicit points like eateries and web based dating. Recommender system have been created to investigate look into articles and specialists, partners, monetary administrations and life coverage.

2. Overview

Ordinarily there are masses of data created and the presence of a lot of data makes it barely to mining the needed data. Customized suggestion is the procedure to alleviative the issue. Collaborative filtering is one of the most well known innovations in the individual suggestion system. As the client rating network turning out to be incredibly sparsity, conventional community separating suggestion calculation ascertains closeness between things utilizing the rating information, and it doesn't think about the semantic connection between various things, along these lines proposal quality is poor.

Recommender systems typically utilize either or both collective filtering and substance based separating (otherwise called the character based methodology), just as different frame- works, for example, information based frameworks. Collaborative filtering approaches construct a model from a client's past conduct (things recently bought or chose as well as numerical appraisals given to those things) just as comparable choices made by different clients. This model is then used to anticipate things (or appraisals for things) that the client may have an enthusiasm for. Content-based filtering approaches use a progression of discrete, pre-labeled qualities of a thing so as to suggest extra things with comparative properties. Current recommender system commonly consolidate at least one methodologies into a cross breed system.

A. Collaborative Filtering

Collaborative filtering depends on the supposition that individuals who concurred in the past will concur later on, and that they will like comparable sorts of things as they loved previously. The system produces suggestions utilizing just data about rating profiles for various clients or things. By finding peer clients/things with a rating history like the present client or thing, they create suggestions utilizing this area. The client and thing based closest neighbor calculations can be joined to manage the virus start issue and improve suggestion results utilizing this information. Collaborative filtering strategies are named memory-based and model-based.

B. Content based filtering

Content- based filtering strategies are based with respect to a portrayal of the thing and a profile of the client's inclinations. These techniques are most appropriate to circumstances where there is known information on a thing (name, area, portrayal, and so on.), yet not on the client. Content-based recommenders treat proposal as a client explicit order issue and get familiar with a classifier for the client's preferences dependent on item includes.

To make a client profile, the system for the most part centers around two kinds of data:

- A model of the client's inclination.
- A background marked by the client's communication with the recommender system.

3. Literature Survey

[1]F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh "Recommendation system: Principles, methods and evaluation"

On the Internet, where the quantity of decisions is overpowering, there is have to channel, organize and proficiently convey applicable data so as to lighten the issue of data over-burden, which has made a potential issue to numerous Internet clients. Recommender system take care of this issue via looking through enormous volume of powerfully produced data to give clients customized substance and administrations. This paper investigates the various qualities and possibilities of various forecast procedures in suggestion system so as to fill in as a compass for research and practice in the field of proposal system.

[2]ShadiAlian, Juan Li and Vikrampandey "A Personalized Recommendation system to support Diabetes Self- Management for American Indians

The plague of diabetes in American Indian (AI) people group is a genuine general wellbeing challenge. The occurrence and commonness of diabetes have expanded drastically with going with increments in body weight and reduced physical movement. In this paper, we know about propose a proactive diabetes self-care recommendation system explicitly for AI patients. It prescribes sound way of life to clients to battle for their diabetes. Because of the semi pervasive utilization of cellphones in most AI clans, we pick cellphones as the stage to give keen individual consideration to AI patients. By coordinating the AI clients' ontological profile with general clinical diabetes suggestion and rules, the system can make customized proposals (e.g., nourishment consumption and physical exercise) in view of the exceptional financial, social, and land status especially to AI patients. The proposed system was actualized as versatile applications. Assessments performed by use contextual investigations and human master confirmation show the adequacy of the system.

4. Problem Statement

The world is moving towards web based shopping because of the advantages it gives to the clients. Large online business destinations like Amazon, Flipkart and so on give customized suggestions to their clients dependent on their past conduct. These guide the clients during their shopping and give palatable shopping experience. These suggestions are given by recommender system utilized by the destinations. As these system develop and turn out to be progressively perplexing, increasingly exact forecasts and suggestions are given to the clients. In the event that this proceeds, retail business may wind down later on. Consequently, a system which uses money receipt information to give personalized recommendation to clients can help give business favorable position to the retail locations.

5. Existing Work

A. Existing Work on Personalized Recommendation System

The information caught is utilized by the recommender system to in the long run give the proposals to the client. Recommender system can be available in a wide range of systems and circumstances, and in this way can be actualized from multiple points of view.

There are five categories of recommendation approaches:

1) Non-personalized
2) Demographic-based
3) Collaborative filtering 4)Content-based 5)Knowledge-based

B. Examination of Hybrid Online Music Recommender system

A hybrid recommender system which gives music suggestions dependent on content and collective filtering and further- more utilizing setting for better proposals. The philosophy to suggest music has the accompanying strides with clarification:

1) At first the important highlights must be accumulated utilized in music recommendation.

May-June 2020
ISSN: 0193-4120 Page No. 5340-5345

2) In the wake of gathering information additionally including rating information and logical data, we apply advance algorithm for significant recommendation.

Step 1: Gather client data

For the new client, the system solicitations to enroll him a record to assemble his own data and furthermore the evaluations of a lot of songs. New User sign in Store client data Fetch evaluations of songs. Song Database Gather song data Recommendations Neighbor set Songs recommender User database.

Step 2: Create song database

Collect the song information such as title, singer, director, composer, release date, ratings etc. for recommendations.

Step 3: Song Recommender

This phase recommend songs to the user.

C. Execution Analysis of Recommendation System Based On Collaborative Filtering and Demographics

The proposed hybrid recommender system that consolidate forecast utilizing item based collaborative filtering and demo- graphic based client bunch in weighted plan is utilized. To make system adaptable, thing closeness and client bunch are registered disconnected.

D. Grouping huge Data Set with Mixed Numeric and Categorical Values

To conquer the issues in K-means while protecting its productivity, K-models were created. Here a grouping calculation is introduced to take care of information parcel issues in information mining. The algorithm depends on the k- means worldview yet evacuates the numeric information just impediment while safeguarding its proficiency. The algorithm bunches objects with numeric and straight out properties in a manner like k-means.

E. Drawbacks

- All the current strategies target online locales and clients on the web.
- No Definite System is created for Customers who want to shop in physical Retails stores face to face.
- K-means, while proficient at bunching, it works just for Numerical data. Therefore, rendering it pointless for categorical data or for mixed data types i.e., a mixture of both Numerical and Categorical data.
-

6. Proposed Model

The system proposed, consists of three modules.

- A device which acts as a hotspot and is used to create a private wireless network to its users. It also queries the database and pushes out specific offers to the user.
- An app installed on the user's smartphone, connects to the store's private network via hotspot.
- A simple database which stores the user's info connected to the store's network.

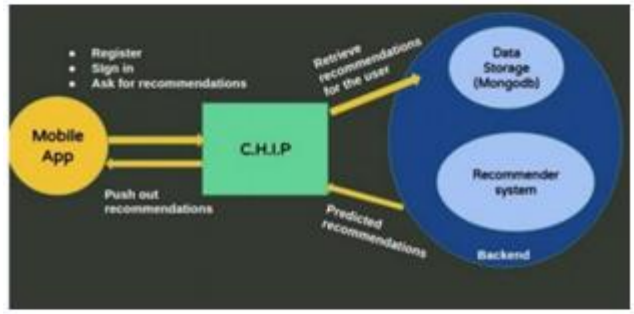

Figure 1: Architecture of proposed system

A. Backend process

A Recommender Engine/ System is developed to generate recommendations specific to an individual user and to store them in the Database.

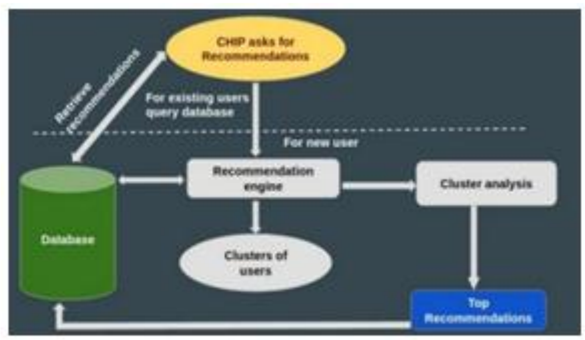

Figure 2: Flow of process in Recommendation system

B. K-prototypes Algorithm

The k-prototypes algorithm can be portrayed as follows.

1) Select k initial prototypes from a informational collection X, one for each cluster.

2) Apportion each article in X to a cluster whose model is the closest to it as indicated by cost function. Update the model of the group after every portion.

3) After the sum total of what articles have been dispensed to a cluster, retest the closeness of items against the present models. On the off chance that an article is discovered with the end goal that its closest model has a place with another cluster as opposed to its present one, reallocate the item to that group and update the models of the two clusters.

4) Rehash (3) until no item has changed clusters after a full cycle trial of X. The algorithm is based upon three procedures, beginning models choice, introductory portion, and reallocation.

May-June 2020
ISSN: 0193-4120 Page No. 5340-5345

C. Taking care of Missing Data

- The algorithm treats any missing/concealed information as coordinating with one another however jumbling with non-missing/seen information while deciding similitude between focuses.
- While foreseeing, the model treats any qualities in grid that

– It has not seen before during preparing

– Are missing, similar to an individual from the "obscure qualities" class.

- K-models acknowledges np.NaN values as missing qualities in the Matrix for all out qualities and for numerical qualities, it doesn't acknowledge missing qualities.
- K-modes algorithm acknowledges np.NaN values as missing qualities in the system, fitting them in their own class.

D. Cluster Selection for Target user

To appoint target client p to a cluster, we figure the Euclidean separation among p and every one of the cluster centroids.

We pick that cluster whose centroid has the least Euclidean separation, and we add p to that cluster gave the Euclidean separation is not exactly a limit.

7. Implementation

A. Import modules

- pandas and numpy for information control.
- turicreate for performing model determination and assessment.
- sklearn for parting the information into train and test set.

```
%load_ext autoreload
%autoreload 2

import pandas as pd
import numpy as np
import time
import turicreate as tc
from sklearn.cross_validation import train_test_split

import sys
sys.path.append("..")
```

B. Burden information

Two datasets in .csv position are utilized underneath, which can be found in information organizer:

- recommend1.csv comprising of a rundown of 1000 client IDs to suggest as yield.
- trxdata.csv comprising of client exchanges.

```
customers = pd.read_csv('../data/recommend_1.csv')
transactions = pd.read_csv('../data/trx_data.csv')
```

C. Data Preparation

We will probably separate each rundown of things in the items section into lines and check the quantity of items purchased by a client.

1) Make Data with user, item and Target field.

- This table will be a contribution for our modeling.
- For this situation, our client is customerID,productID and purchase count.

```
data = pd.melt(transactions.set_index('customerId')
['products'].apply(pd.Series).reset_index(),
             id_vars=['customerId'],
             value_name='products') \
    .dropna().drop(['variable'], axis=1) \
    .groupby(['customerId', 'products']) \
    .agg({'products': 'count'}) \
    .rename(columns={'products': 'purchase_count'}) \
    .reset_index() \
    .rename(columns={'products': 'productId'})
data['productId'] = data['productId'].astype(np.int64)
```

2) Create Dummy

- Dummy is for making whether a client purchased that thing or not.
- In the event that one purchases a thing, at that point buy dummy are set apart as 1.
- Normalizing the purchase count, say by every client, would not work since clients may have diverse purchasing recurrence don't have a similar taste. Be that as it may, we can standardize things by buy recurrence over all clients.

```
def create_data_dummy(data):
    data_dummy = data.copy()
    data_dummy['purchase_dummy'] = 1
    return data_dummy

data_dummy = create_data_dummy(data)
```

3) Standardize thing esteems across clients

In Retail store, we have n number of thing to buy, sell and monitor all exchange with it. In this way, We standardize buy recurrence of every thing across client by first making a client thing lattice as follows:

```
df_matrix = pd.pivot_table(data, values='purchase_count',
index='customerId', columns='productId')

df_matrix_norm = (df_matrix-df_matrix.min())/(df_matrix.max()-
df_matrix.min())
```

```
# create a table for input to the modeling

d = df_matrix_norm.reset_index()
d.index.names = ['scaled_purchase_freq']
data_norm = pd.melt(d, id_vars=['customerId'],
value_name='scaled_purchase_freq').dropna()

print(data_norm.shape)
data_norm.head()
```

The above steps can be combined to a function below:

```
def normalize_data(data):
    df_matrix = pd.pivot_table(data, values='purchase_count',
index='customerId', columns='productId')
    df_matrix_norm = (df_matrix-df_matrix.min())/(df_matrix.max()-
df_matrix.min())
    d = df_matrix_norm.reset_index()
    d.index.names = ['scaled_purchase_freq']
    return pd.melt(d, id_vars=['customerId'],
value_name='scaled_purchase_freq').dropna()
```

May-June 2020
ISSN: 0193-4120 Page No. 5340-5345

4) Split Train and Test Set

- Parting the information into preparing and testing sets is a significant piece of assessing prescient displaying, for this situation a collaborative filtering model. Commonly, we utilize a bigger bit of the information for preparing and a littler bit for testing.
- We utilize 80:20 proportion for train-test set size.
- Our preparation segment will be utilized to build up a prescient model, while the other to assess the model's presentation.

```
def split_data(data):
    '''
    Splits dataset into training and test set.

    Args:
        data (pandas.DataFrame)

    Returns
        train_data (tc.SFrame)
        test_data (tc.SFrame)
    '''
    train, test = train_test_split(data, test_size = .2)
    train_data = tc.SFrame(train)
    test_data = tc.SFrame(test)
    return train_data, test_data
```

5) Characterize Models utilizing Turicreate library

Before running a progressively entangled methodology, for example, collaborative filtering, we should run a benchmark model to analyze and assess models. Since standard regularly utilizes a straightforward methodology, systems utilized past this methodology ought to be picked on the off chance that they show moderately better precision and unpredictability. For this situation, we will utilize popularity model.

```
# constant variables to define field names include:

user_id = 'customerId'
item_id = 'productId'
users_to_recommend = list(customers[user_id])
n_rec = 10 # number of items to recommend
n_display = 30 # to display the first few rows in an output dataset

def model(train_data, name, user_id, item_id, target,
users_to_recommend, n_rec, n_display):
    if name == 'popularity':
        model = tc.popularity_recommender.create(train_data,
                                                    user_id=user_id,
                                                    item_id=item_id,
                                                    target=target)
    elif name == 'cosine':
        model = tc.item_similarity_recommender.create(train_data,
                                                    user_id=user_id,
                                                    item_id=item_id,
                                                    target=target,

similarity_type='cosine')

elif name == 'pearson':
        model = tc.item_similarity_recommender.create(train_data,
                                                    user_id=user_id,
                                                    item_id=item_id,
                                                    target=target,

similarity_type='pearson')

    recom = model.recommend(users=users_to_recommend, k=n_rec)
    recom.print_rows(n_display)
    return model
```

8. Result

The recommendation system is a significant instrument to offer personalized recommendation. Current recommendation system infrequently utilize intuitive strategies to progressively change loads of recommendation algorithms, in order to accomplish precise recommendation. We know that genuinely effective recommendation system ought to be founded on a personalized recommendation algorithms, to shape long term stable relationship with buyers. The fundamental commitments of this research are twofold. Firstly, we focus on that in some cases it is difficult for purchasers to communicate their inclinations. Purchasers need to communicate with the system and assess the recommendation items given by the system. At that point, the system runs rehashed patterns of intuitive recommendation to guarantee the precision of proposals in information mining. Exactness is significant for the recommendation items. Precise suggestions can improve consumer loyalty, yet in addition improve corporate deals. Secondly, to accomplish exact recommendation, we propose an increasingly powerful model utilizing a blend of recommendation algorithm in information mining. The hybrid recommendation method, supplements the benefits of the current recommendation algorithm, acquires unique outcomes with every calculation, and afterward sets up the loads as indicated by the intuitive out- come with consumer. our model offers significant implications for online business stage suppliers with respect to the design of product recommendation system. Web based business stage administrators could redesign the recommendation algorithm of recommendation system dependent on the above model, to choose increasingly fitting items to suggest.

References

[1] Pengbo Li, Guisong Zhang, Li Chao, Zhifeng Xie, "Personalized Recommendation system for Offline Shopping," International Conference on Audio, Language and Image Processing (ICALIP), Shanghai, China, July 2018.

[2] F.O. Isinkaye, Y.O. Folajimi, B.A. Ojokoh "Recommendation sys- tem: Principles, methods and evaluation," Egyptian Informatics Journal, November 2015.

[3] Shadi Alian, Juan Li and Vikram pandey "A Personalized Recommendation system to support Diabetes Self- Management for American Indians," IEEE publication, USA, December 2018.

[4] Yulong Ying "The Personalized Recommendation Algorithm Based on item Semantic Similarity," chaina, January 2011.

[5] Yeounoh Chung, Hye-Wuk Jung, Jaekwang Kim, Jee-Hyong Lee "Personalized Expert Based Recommender System: Training C-SVM for

Personalized Expert Identification," Korra, 2013.
[6] Xing Xie, Jianxun Lian, Zheng Liu, Xiting Wang, Fangzhao Wu, Hongwei Wang, and Zhongxia Chen " Personalized Recommendation Systems: Five Hot Research", China, November 2018.
[7] Yufeng Wang "A Beginner Overview of Personalized Recommendation", Machine learning Deep learning blogger, Hong Kong, January 2019.