**Mahout code**

**Code1**

import org.apache.mahout.cf.taste.common.TasteException;

import org.apache.mahout.cf.taste.recommender.RecommendedItem;

public class Moviereccom {

public static String[] getMovieList(String userName, Model mod1)

throws TasteException {

List<String> movielist = new ArrayList<String>();

try {

List<RecommItem> movie\_l1 = mod1.model().recommend(model.getStrTolng().toLongID(userName), 10);

for (RecommendedItem movie : movie\_l1) {

movielist.add(model.getStrTolng().toStringID(

movie.getItemID()));

}

} catch (TasteException e) {

throw e;

}

return movielist.toArray(new String[movielist.size()]);

}

public static void main(String[] args) {

String userName = "mkkzr6";

Model mod1= new Model();

try {

String[] list = getMovieList(userName, mod1);

System.out.println("Recommended movie for " + userName

+ " \n\n");

for (String result : list) {

System.out.println(result);

}

} catch (TasteException e) {

e.printStackTrace();

}

}

}

**Code2**

package edu.mkk.moviereccom;

import java.io.File;

import java.io.FileInputStream;

import java.io.FileReader;

import java.io.InputStreamReader;

import java.io.Reader;

import java.util.ArrayList;

import java.util.HashMap;

import java.util.List;

import java.util.Map;

import java.util.Map.Entry;

import org.apache.commons.csv.CSVParser;

import org.apache.mahout.cf.taste.impl.common.FastByIDMap;

import org.apache.mahout.cf.taste.impl.model.GenericDataModel;

import org.apache.mahout.cf.taste.impl.model.GenericPreference;

import org.apache.mahout.cf.taste.impl.model.GenericUserPreferenceArray;

import org.apache.mahout.cf.taste.impl.model.MemoryIDMigrator;

import org.apache.mahout.cf.taste.impl.recommender.GenericBooleanPrefItemBasedRecommender;

import org.apache.mahout.cf.taste.impl.similarity.LogLikelihoodSimilarity;

import org.apache.mahout.cf.taste.model.DataModel;

import org.apache.mahout.cf.taste.model.Preference;

import org.apache.mahout.cf.taste.model.PreferenceArray;

import org.apache.mahout.cf.taste.recommender.Recommender;

public class Model {

private Recommender recom = null;

private Mem strTolng = new MemoryIDMigrator();

private DataModel model;

public Mem getStrTolng() {

return strTolng;

}

public Recommender model()

{

String[] line;

try {

Map<Long, List<Preference>> userPrefLists = new HashMap<Long, List<Preference>>();

CSVParser csv = new CSVParser(new InputStreamReader(

new FileInputStream("/home/murali/Desktop/KDM/datasets/midterm/Book\_Ratings2.csv"),

"UTF-8"));

csv.getLine();

List<Preference> userPrefrence;

while ((line = csv.getLine()) != null) {

String user\_id = line[0];

String movie\_id = line[1];

if(Integer.parseInt(line[2])>=4)

{

long userLong = strTolng.toLongID(user\_id);

strTolng.storeMapping(userLong, user\_id);

long movieLong = strTolng.toLongID(movie\_id);

strTolng.storeMapping(movieLong, movie\_id);

if ((userPrefrence = userPrefLists.get(userLong)) == null) {

userPrefrence = new ArrayList<Preference>();

userPrefLists.put(userLong, userPrefrence);

}

userPrefrence.add(new GenericPreference(userLong, movieLong, 1));

}

}

FastByIDMap<PreferenceArray> prefrenceHashMap = new FastByIDMap<PreferenceArray>();

for (Entry<Long, List<Preference>> entry : userPrefLists.entrySet()) {

prefrenceHashMap.put(entry.getKey(),new GenericUserPreferenceArray(entry.getValue()));

}

model = new GenericDataModel(prefrenceHashMap);

recommender = new GenericBooleanPrefItemBasedRecommender(model,new LogLikelihoodSimilarity(model));

} catch (Exception e) {

e.printStackTrace();

}

return recommender;

}

}

**Android web app**

Xml

<?xml version="1.0" encoding="UTF-8"?>

[<manifest android:versionName="1.0" android:versionCode="1" package="com.example.weatherapp" xmlns:android="http://schemas.android.com/apk/res/android">](file:///C:\Users\murali\workspace\weatherapp\AndroidManifest.xml)<uses-sdk android:targetSdkVersion="18" android:minSdkVersion="10"/><uses-permission android:name="android.permission.INTERNET"/>[<application android:theme="@style/AppTheme" android:label="@string/app\_name" android:icon="@drawable/ic\_launcher" android:allowBackup="true"><activity android:name="com.example.weatherapp.MainActivity" android:label="@string/app\_name"><intent-filter>](file:///C:\Users\murali\workspace\weatherapp\AndroidManifest.xml)<action android:name="android.intent.action.MAIN"/><category android:name="android.intent.category.LAUNCHER"/></intent-filter></activity></application></manifest>

Main activity.java

package com.example.weatherapp;

import java.io.IOException;

import java.io.InputStream;

import org.apache.http.HttpEntity;

import org.apache.http.HttpResponse;

import org.apache.http.client.ClientProtocolException;

import org.apache.http.client.HttpClient;

import org.apache.http.client.methods.HttpPost;

import org.apache.http.impl.client.DefaultHttpClient;

import org.json.JSONArray;

import org.json.JSONException;

import org.json.JSONObject;

import android.os.AsyncTask;

import android.os.Bundle;

import android.app.Activity;

import android.view.Menu;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends Activity {

Button getlatlon,getweather;

TextView lat,lon,temp2;

EditText addr1,addr2;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

addr1 = (EditText) findViewById(R.id.editText1);

addr2 = (EditText) findViewById(R.id.editText2);

getlatlon = (Button) findViewById(R.id.button1);

lat = (TextView) findViewById(R.id.textView1);

lon = (TextView) findViewById(R.id.textView2);

temp2=(TextView) findViewById(R.id.textView4);

getweather = (Button) findViewById(R.id.button2);

getlatlon.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

String add = addr1.getText()+"";

add = add + "" + addr2.getText();

GetLatLong gll = new GetLatLong();

gll.execute(new String[]{add});

}

});

getweather.setOnClickListener(new View.OnClickListener(){

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

String latln = lat.getText()+"";

latln = latln + " " + lon.getText();

GetWeather gw = new GetWeather();

gw.execute(new String[]{latln});

}

});

}

class GetLatLong extends AsyncTask<String,Void,JSONObject>{

@Override

protected JSONObject doInBackground(String... arg0) {

// TODO Auto-generated method stub

StringBuilder sb = new StringBuilder();

try{

String address = arg0[0];

address = address.replaceAll(" ", "%20");

HttpPost httppost = new HttpPost("http://maps.google.com/maps/api/geocode/json?address="+address+"&sensor=false");

HttpClient client = new DefaultHttpClient();

HttpResponse response;

sb = new StringBuilder();

response = client.execute(httppost);

HttpEntity entity = response.getEntity();

InputStream stream = entity.getContent();

int b;

while((b = stream.read())!=-1){

sb.append((char)b);

}

}catch(ClientProtocolException e){

} catch(IOException e){

}

JSONObject jsonObject = new JSONObject();

try{

jsonObject = new JSONObject(sb.toString());

}catch(JSONException e){

e.printStackTrace();

}

return jsonObject;

}

protected void onPostExecute(JSONObject jsonObject){

super.onPostExecute(jsonObject);

try{

//String location = "";

double longitude = ((JSONArray)jsonObject.get("results")).getJSONObject(0)

.getJSONObject("geometry").getJSONObject("location")

.getDouble("lng");

double latitude = ((JSONArray)jsonObject.get("results")).getJSONObject(0)

.getJSONObject("geometry").getJSONObject("location")

.getDouble("lat");

MainActivity.this.lat.setText(""+latitude);

MainActivity.this.lon.setText(""+longitude);

}catch(JSONException e){

e.printStackTrace();

}

}

}

class GetWeather extends AsyncTask<String,Void,JSONObject>{

@Override

protected JSONObject doInBackground(String... arg0) {

// TODO Auto-generated method stub

StringBuilder sb1 = new StringBuilder();

try{

String lat = arg0[0].split(" ")[0];

String longg = arg0[0].split(" ")[1];

lat = lat.replaceAll(" ", "%20");

longg = longg.replaceAll(" ", "%20");

HttpPost httppost1 = new HttpPost("http://api.openweathermap.org/data/2.5/weather?lat="+lat+"&lon="+longg+"&sensor=false");

HttpClient client1 = new DefaultHttpClient();

HttpResponse response1;

sb1 = new StringBuilder();

response1 = client1.execute(httppost1);

HttpEntity entity1 = response1.getEntity();

InputStream stream1 = entity1.getContent();

int b1;

while((b1 = stream1.read())!=-1){

sb1.append((char)b1);

}

}catch(ClientProtocolException e){

} catch(IOException e){

}

JSONObject jsonObject = new JSONObject();

try{

jsonObject = new JSONObject(sb1.toString());

}catch(JSONException e){

e.printStackTrace();

}

return jsonObject;

}

protected void onPostExecute(JSONObject jsonObject){

super.onPostExecute(jsonObject);

try{

//String location = "";

double temp = jsonObject.getJSONObject("main").getDouble("temp");

MainActivity.this.temp2.setText(""+temp);

}catch(JSONException e){

e.printStackTrace();

}

}

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.main, menu);

return true;

}

}