电影推荐系统

未来互联网与计算兴趣团队
王少雄

内容概述

- 算法部分
  - 协同过滤 Collaborative Filtering
    - User/Item-base ; Matrix Factorization
- 技术部分
  - python数据处理
  - Flask框架

推荐系统

- 广泛并成功应用于互联网公司
- 经典成功案例有Amazon和Netflix

协同过滤

核心思想

- 利用实体间的相互关联进行预测

1.User-based/Item-based

Rating prediction

$$\hat{r}_{u,i} = \bar{r}_u + \kappa \sum_{v \in N(u)} sim(u,v) \times (r_{v,i} - \bar{r}_v)$$

- $r_{u,i}$: (observed) rating of user *u* for item *i*
- $\bar{r}_u$: mean rating of user *u*
- $\hat{r}_{u,i}$: predicted rating of user *u* for item *i*
- $N(u)$: set of users similar to user *u* (who have rated item *i*)
- $sim(u,v)$: similarity of users *u* and *v*
- κ : normalization factor

1.User-based/Item-based

Item-based

- Rating prediction

$$\hat{r}_{u,i} = \bar{r}_i + \kappa \sum_{j \in N(i)} sim(i, j) \times (r_{u,j} - \bar{r}_j)$$

2.Low-rank Matrix Factorization

- R => X * Θ
- s.t. min (R - X * Θ).^2

python数据获取

- urllib.urlretrieve
- progressbar

pyhton

- from urllib import urlretrieve
- url = "http://a3.att.hudong.com/22/89/300001051406131452898160507_950.jpg"
- urlretrieve(url, "/Users/shawn/Desktop/trash/fengjie.jpg")

python数据获取

- import urllib2
- url = "http://img1.imgtn.bdimg.com/it/u=1155447503,3876658488&fm=21&gp=0.jpg"
- request = urllib2.Request(url)
- request.add_header('User-Agent', 'Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_4) AppleWebKit/537.77.4 (KHTML, like Gecko) Version/7.0.5 Safari/537.77.4')
- res = urllib2.urlopen(request)
- pic = res.read()
- with open("/Users/shawn/Desktop/trash/fengjie2.jpg","w") as f:
- print >> f, pic

python progressbar

- 显示进度条

apt-get/homebrew,pip

- ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"

pip

- curl https://bootstrap.pypa.io/ez_setup.py -o - | sudo python
- sudo easy_install pip
- pip install progressbar

电影推荐

- Data：http://grouplens.org/datasets/movielens/
  - **movies.csv**
  - **ratings.csv**
  - links.csv
  - tags.csv

rating.csv

- userId,movieId,rating,timestamp
- 1,1,5.0,847117005
- 1,2,3.0,847642142
- 1,10,3.0,847641896
- 1,32,4.0,847642008

rating.csv

- userId,movieId,rating,timestamp
- 1,1,5.0,847117005
- with open(rate_path, "r") as f:
- 	f.readline()
- 	for line in f.readlines():
- 		(userId,movieId,rating,timestamp
-) = line.split(',')

Flask

- pip install flask

Getting started

```
from flask import Flask

app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```

路由

@app.route(“/“)

def index():

@app.route(“about”)

def func():

@app.route('/user/<int:username>')

def show_user_profile(username):

MVC 渲染模板

```
from flask import render_template

@app.route('/hello/')

@app.route('/hello/<name>')

def hello(name=None):

    return render_template('hello.html', name=name)
```

hello.html

- <!doctype html>
- <title>Hello from Flask</title>
- {% if name %}
- <h1>Hello {{ name }}!</h1>
- {% else %}
- <h1>Hello World!</h1>
- {% endif %}