Game Recommendation

Xiaonan Duan

Project Overview

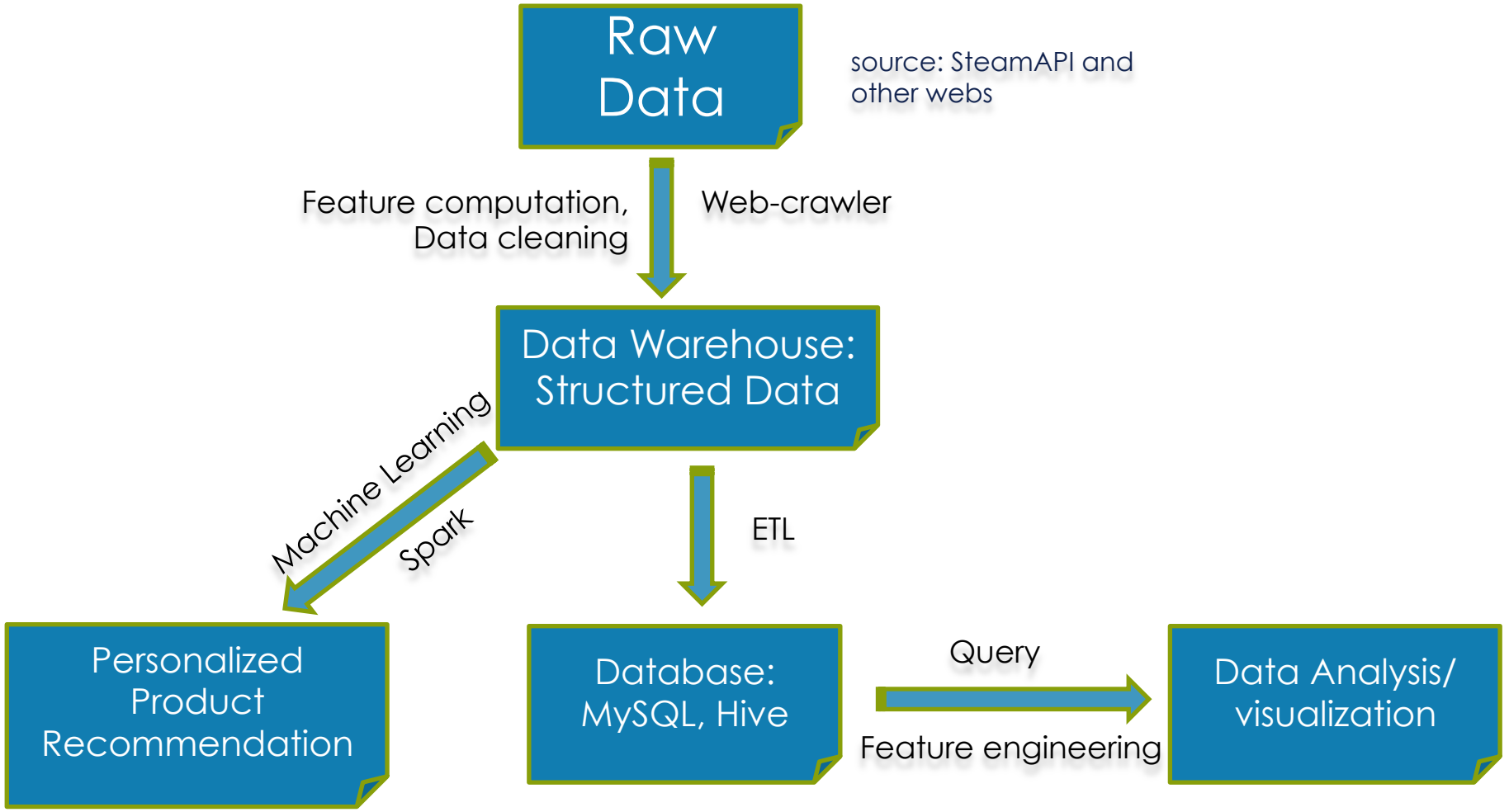
Objective:

- Obtain structured user and product data from internet and ETL to database
- Implement a personalized game recommendation engine to each user
- Provide business insights to game industry

Skills and tools:

- Machine learning, ETL, web-crawling, data analysis, feature selection and engineering
- Python(pandas, numpy, sklearn, matplotlib, requests), MySQL, Spark and Hive

Project Flow



ETL-part I

- Web-crawler: extracted raw data from SteamAPI and other webs using python's requests and beautiful soup.
- Loaded data(user and product) into MySQL and Hive.

Code examples :

```
import requests
import json
my_key ='XXXXXXXXXX'

def steamAPI(query):
    load = {'key': my_key,
            'steamid': query,
            'format': json}
    r = requests.get('http://
api.steampowered.com/IPlayerService/
GetOwnedGames/v0001/',
                     params = load)
    return r.json()
```

```
from sqlalchemy import create_engine
import mysql.connector

engine = create_engine('mysql
+mysqlconnector://root:XXXXXX@127.0.0.1/
gameRecommendation')

df_simple_product.to_sql('product_table',con=e
ngine,if_exists='replace',index = False)
```

Project –ETL-part I

Data examples :

user_new.txt

{"[redacted]": [{"playtime_forever": 370, "playtime_2weeks": 49, "appid": 4000}, {"playtime_forever": 9, "appid": 110800}, {"playtime_forever": 0, "appid": 226320}, {"playtime_forever": 507, "appid": 250320}, {"playtime_forever": 78, "playtime_2weeks": 62, "appid": 296470}, {"playtime_forever": 20, "appid": 301520}, {"playtime_forever": 0, "appid": 205790}, {"playtime_forever": 5973, "playtime_2weeks": 64, "appid": 730}, {"playtime_forever": 54, "appid": 218620}, {"playtime_forever": 36, "appid": 352460}, {"playtime_forever": 0, "appid": 34270}, {"playtime_forever": 1, "appid": 205230}, {"playtime_forever": 0, "appid": 205950}]}
{"[redacted]": [{"playtime_forever": 2389, "playtime_2weeks": 52, "appid": 4000}, {"playtime_forever": 14306, "appid": 34030}, {"playtime_forever": 4602, "appid": 42680}, {"playtime_forever": 11055, "appid": 42690}, {"playtime_forever": 625, "appid": 50300}, {"playtime_forever": 173, "appid": 104900}, {"playtime_forever": 1120, "appid": 113400}, {"playtime_forever": 3621, "appid": 203290}, {"playtime_forever": 114, "appid": 206210}, {"playtime_forever": 11, "playtime_2weeks": 11, "appid": 211500}, {"playtime_forever": 274, "playtime_2weeks": 178, "appid": 218230}, {"playtime_forever": 1005, "playtime_2weeks": 617, "appid": 236390}, {"playtime_forever": 0, "appid": 107400}, {"playtime_forever": 1283, "appid": 224260}, {"playtime_forever": 1933, "appid": 233450}, {"playtime_forever": 1532, "appid": 242760}, {"playtime_forever": 19646, "appid": 107410}, {"playtime_forever": 988, "playtime_2weeks": 239, "appid": 218620}, {"playtime_forever": 4322, "playtime_2weeks": 38, "appid": 252950}, {"playtime_forever": 35, "appid": 301520}, {"playtime_forever": 102, "appid": 304050}, {"playtime_forever": 1923, "appid": 304930}, {"playtime_forever": 102, "appid":

all_appids.txt

{"221540": {"data": {"steam_appid": 221540, "achievements": {"highlighted": [{"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/719caffcf213b00c2d97cec58d4bed97c241d095.jpg", "name": "Eradicator"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/2f00fa7093f387bf6fc7445a6e547b547ae44840.jpg", "name": "First Blood"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/702f988cbcbb468f0b0089b8591ebe038965a888.jpg", "name": "Annihilator"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/b627f12c4aa9e4a55747437ed44beaca4f5e2ec8.jpg", "name": "Exterminator"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/a5dd80cb51d5630fb2e31b378cb682ff01cbafee.jpg", "name": "Xenocide"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/52270f8d6cdfcce9d27d74f58efce21c4a1957ae.jpg", "name": "Field Promotion"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/d2bf88a851368235b7541d8c8cfb429405ca81b0.jpg", "name": "Flawless Victory"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/50cfc5f0ade9797fafdb5cb5653df56d966148dd.jpg", "name": "Salvage Rights"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/4f2f96d50d19f8a041a2e36f401313bb31a37a15.jpg", "name": "Liquidator"}, {"path": "http://cdn.akamai.steamstatic.com/steamcommunity/public/images/apps/221540/54c64fa7f65fca798e7bf82ed9e2a84434a00452.jpg", "name": "Surplus"}], "total": 65}, "price_overview": {"currency": "USD", "initial": 1499, "final": 1499, "discount_percent": 0}, "platforms": {"windows": true, "mac": true, "linux": true}, "detailed_description": "<h1>Special Edition</h1><p>Special Edition purchasers will receive the digital book, The Art of Defense Grid 2, plus the ebook, The Making of Defense Grid 2: The Complete Story Behind the Game by Russ Pitts, and "A Matter of Endurance" audiobook written by Hugo award-winning author Mary Robinette Kowal and performed by the English cast.\r
\r
THE ART OF DEFENSE GRID 2 digital art book celebrates the talents and efforts of the people responsible for creating the visuals of the game. It is an insight of the concepts and variable directions that the team explored during the game's creation. The journey in visual exploration is a winding and strange path at times. This book revisits that journey, come along and enjoy it. \r
\r
THE MAKING OF DEFENSE GRID 2 brings you behind-the-scenes of the creation of a modern video game. \r
Over nearly two years and hundreds of

Database-part II

- Created tables in MySQL database and data analysis by queries.
- Loaded data(user and product) into MySQL and Hive.

Code examples :

```
Create table product_2 as
(SELECT app_id, COUNT(user_id)
AS num_players,
AVG(playtime_forever) AS
avg_playtime FROM
user_app_pair GROUP BY
app_id);
```

```
CREATE table final_product as
(SELECT p.app_id, p.num_players,
p.avg_playtime,
a.initial_price,a.score FROM
product_table AS a JOIN product_2
AS p ON a.steam_appid =
p.app_id);
```

Database-part II

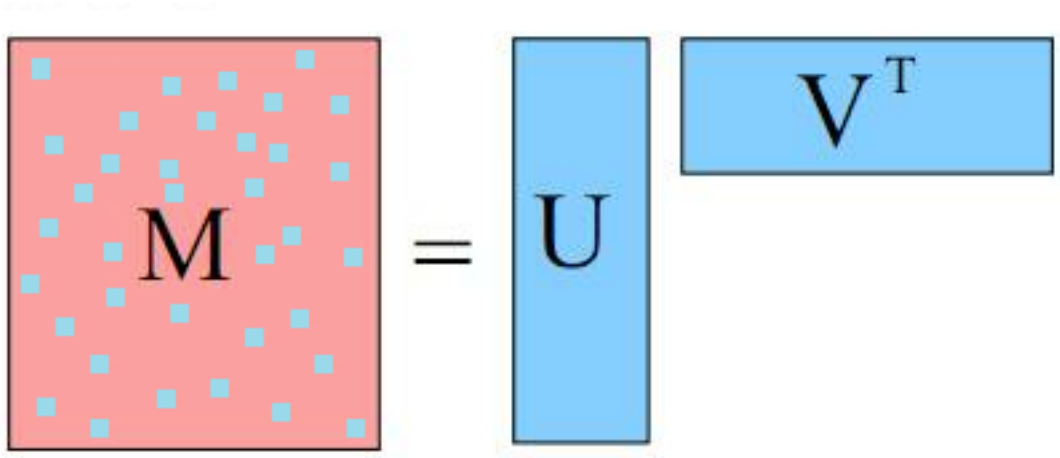
- **Data analysis example:**

Top 10 popular games (by num_players)

	app_id	num_players	avg_playtime	initial_price	score	name
0	730	3480	28745.1224	1499	83	Counter-Strike: Global Offensive
1	4000	3290	13024.1447	999		Garry's Mod
2	352460	3162	276.7995	1499		Dead Realm
3	304930	3104	1231.0264			Unturned
4	550	2515	2617.4962	1999	89	Left 4 Dead 2
5	218620	2186	3326.6331	1999	79	PAYDAY 2
6	230410	1949	2695.6814		68	Warframe
7	105600	1755	4057.3704	999	83	Terraria
8	240	1733	5586.9919	1999	88	Counter-Strike: Source
9	301520	1727	688.1257			Robocraft

Recommendation-ML-Part III

- Applied collaborative filtering using matrix factorization (Alternating Least Square) on data by PySpark(Spark MLlib and Spark SQL) for product recommendation.



http://www.cs.cmu.edu/~yuxiangw/research.html

Recommendation-ML-Part III

- Applied collaborative filtering using matrix factorization on data by PySpark(Spark MLlib and Spark SQL) for product recommendation.

Code examples :

```
def parse_user(raw_string):
    user_inventory = json.loads(raw_string)
    user_id,lst_user_inventory = user_inventory.items()[0]
    if not lst_user_inventory == None:
        try:
            return [(user_id, i.get('appid'),i.get('playtime_forever')) for i in lst_user_inventory]
        except:
            return []
    else:
        return []
user_rdd = sc.textFile('user_new.txt').flatMap(parse_user)


from pyspark.mllib.recommendation import ALS
model = ALS.train(data, rank=2, seed=0)
#get each user's top 10 game recommendation
model.recommendProducts(index,10)
```