Capstone Design 1 Final Report

캡스톤 디자인 최종 보고서

과제명	**Capstone Design 1**			
프로젝트 설명	데이터분석 &코디 추천 프로그램			
팀명	패피(패션피플)			
수행기간	2019.03.04 ~ 2019.06.13			
지도교수	**성명**	**소속**	**연락처**	**이메일**
	정현숙	IT융합대학	010-2508-9231	hsch@chosun.ac.kr
팀원	**학과**	**학번**	**성명**	**이메일**
	컴퓨터공학과	20144800	천승현	ehwjsx@gmail.com
	컴퓨터공학과	20165172	이유겸	qkl23@naver.com
	컴퓨터공학과	20164216	박지혜	aldzl3701@naver.com
	컴퓨터공학과	20164228	최예솜	choip1221@naver.com

목차

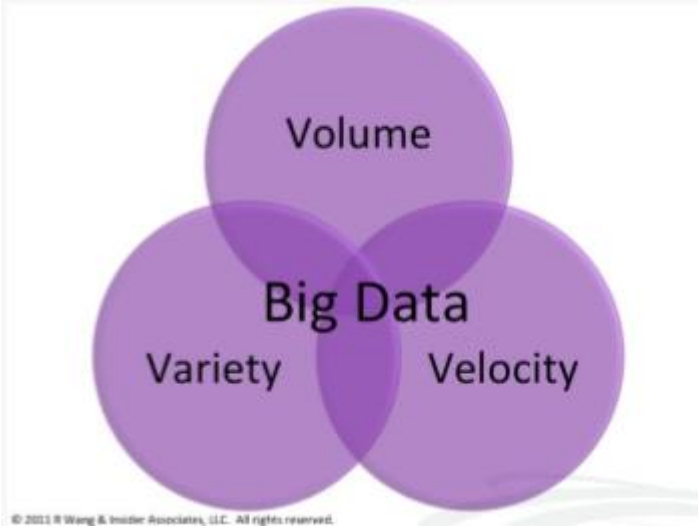
English part

Understanding of Big Data

Big data is the most important IT industry keyword in the modern 4th industrial Revolution. Big data can be defined in many defines by many viewpoint IDC said in June 2011 that big data extracted value from a wide variety of large data types at a low cost, Next-generation technologies and architectures designed to support the rapid collection, discovery and analysis of data have been defined that focus on how the task is performed. McKinsey defined big data as data beyond the data collection, storage, management and analysis capabilities of existing database management tools when it focused on the size of the data in May 2011. Collectively, the data generated in the digital environment can be defined as large-scale, short-lived, and large-scale data containing not only numerical data but also character and image data. With the exponential growth of digital information volume, the types of data vary, forming a predictable environment and analysis of thoughts and opinions through a combination of sensors and embedded systems, formal, unstructured, novel and real-time data.

Concept of Big data

The most important feature of Big data is 3V. 3V means the most important 3 words in Big data. BI/DW research organization TDWI has illustrated the top three elements of big data as shown below.



Volume

Big data has a large capacity. Generally, tens of terabytes or more than a few dozen petabytes fall within the scope of big data. One petabyte is the capacity to hold about 174,000 5 Giga byte DVD movies. Big data is not only difficult to store in existing file systems, but it is also increasing in volume to the point where it is difficult to digest in solutions such as traditional data warehouse used for data analysis. To overcome these problems, a distributed computer approach is needed to store and analyze data in a scalable manner. Hadoop of Apache is a typical example of distributed computing.

Velocity

Velocity can be interpreted two ways. The first is the rapid rate of generation of data and the second is the concept of the processing speed of the data. Object data, web search data, real-time transmission data, and data generated from mobile are often generated at a constant and rapid pace. These big data must be analyzed and processed quickly. Because digital data is generated at a very fast rate today, the collection, storage, and analysis of data must be done in real time.

Variety

The last concept in 3V of big data is the diversity of data. Data are classified according to its characteristics into formal, semi-regular and unstructured formats. Static Data means data stored in a fixed field and is in a constant format. Semi-static data is not a fixed field but includes metadata or schema. Unstructured data means data that is not stored in a fixed field. They include photos, videos, location information, and phone conversations.

static data	unstructured data	semi static data
data stored in a fixed field	data complex in form and structure	Data with somewhat inconsistent values and formats
-relational type R B (R D B) -spreadsheet	-social data (S N S) -file -image -video -movie	-HTML -X M L -Web File -Web Logs -Sensor data

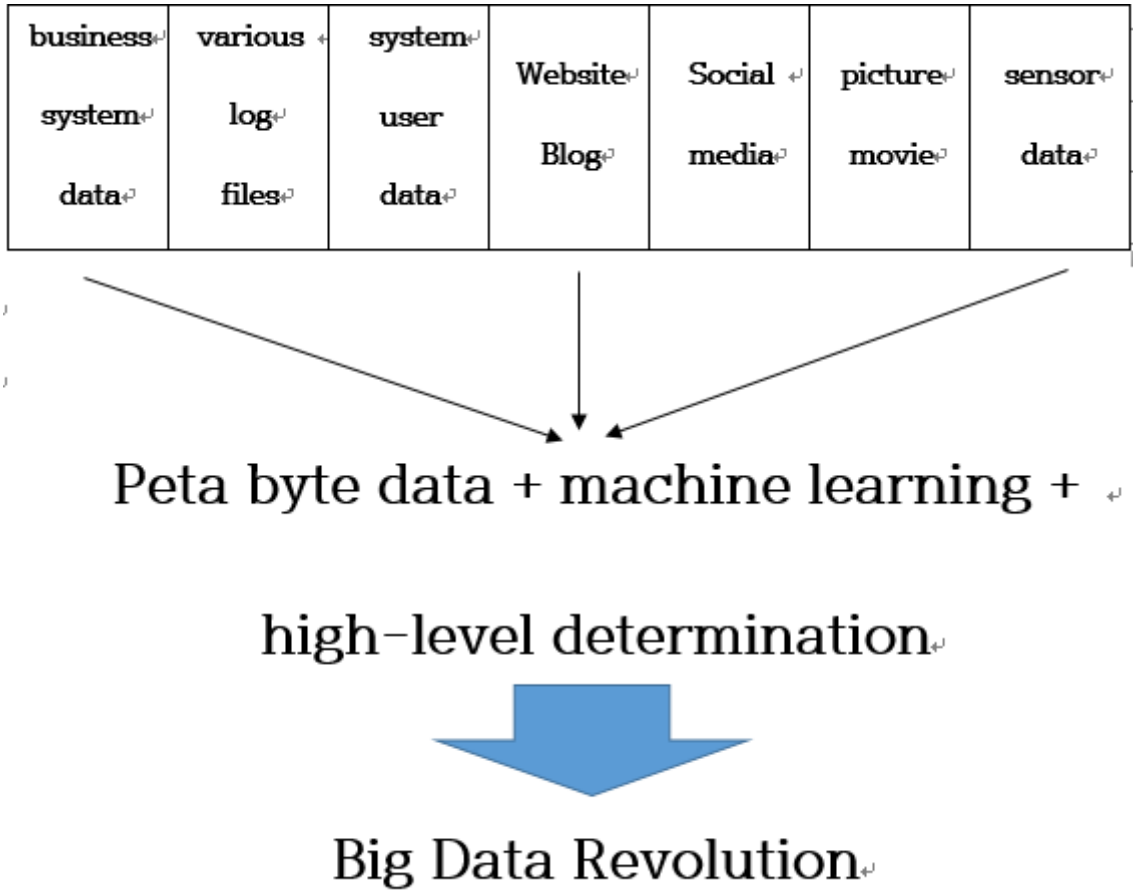
In addition, the new property of value is also called 4V.

Application of Big Data

There are four major steps in the technology of big data. Collection, storage and management, processing and analysis, visualization and utilization stage. First, create new data, collect external data, integrate data, or obtain data regardless of the type and material of the data.

Next, it needs distributed computing storage technology that can store and manage large forms of data in real time. To analyze the value of stored data, analysis methods such as large-scale statistics, predictive processing, and data mining are needed, and in-depth analysis technology using machine learning and artificial intelligence is needed.

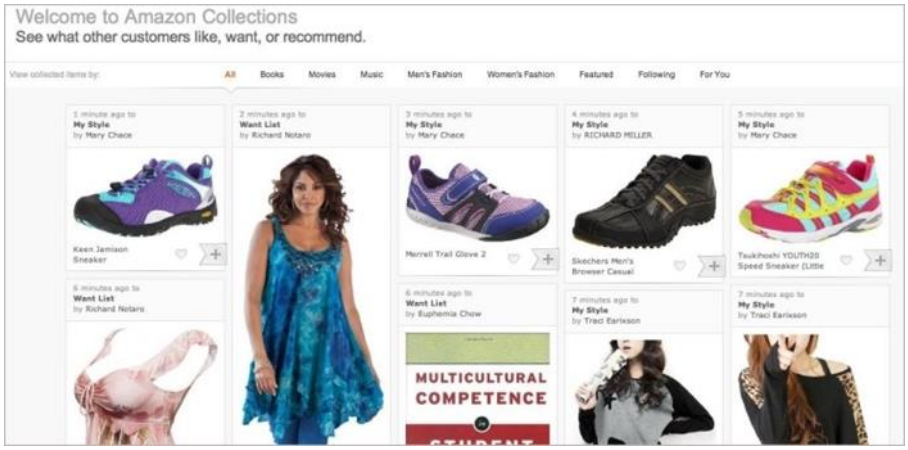
Finally, visualization is needed that consists of graphic technology to develop an analysis tool module that provides an environment for non-specialists to perform data analysis, or to display analysis results and provide intuitive information. There are many reasons behind this sudden emergence of big data. The biggest impact is the spread of mobile devices such as smartphones and the prevalence of social network services, which are producing huge amounts of data. Big data is proving its value in the fields of business, government, education and culture, and companies that actively utilize big data are already demonstrating the effectiveness of big data with business results



The most notable technology in this project is to deeply observe consumers' consciousness or behavior within big data and derive meaningful interpretation by penetrating the innermost feelings, trends and trends that consumers are not self-aware.

Recommendation System

The recommendation system is one of the most powerful and essential strategies in the modern commercial market. Users don't have time to look around all the items and want to recommend the products they need most. Web shopping sites that apply the recommendation system use their algorithms to recommend the right items for users. Users analyze the attributes or ratings of previously purchased products and present relevant products on a recommendation list.



The most famous company with the recommendation system is Amazon by far. Amazon analyzes its members' consumption patterns and recommends products that are available for purchase, with about 35 percent of Amazon's sales coming from recommended products as the No. 1 contributor to Amazon's growth. Amazon has registered patents for its unique item-based collaborative filtering algorithm called 'A9'. Integrate elements in the big data field to develop a web system that applies these recommended systems. It will collect data through the big data collection stage, parse and regularize data through the analysis stage, and use data collected in the utilization stage. It is going to implement its website and algorithm and learn elements of big data in person.

Development motive

As the online content market grows, consumers have more choices. However, consumers are forced to choose content on a limited basis to spend less time efficiently because they do not have the time to consume all of it. Most multimedia service websites use content recommendation algorithms to strategically use such consumer sentiment. Netflix, which has been growing the most recently, is a leading multimedia content service site. Netflix uses strong content recommendation algorithm to analyze user-viewed images to identify trends and recommend content based on your preference. Or, it also carries out a marketing campaign to expose content enjoyed by other users as a list.

☞ Netflix: 66% of the movies are rented by recommendation

☞ Google News: 38% or more views are generated by recommendation

☞ Amazon: 35% of sales come from recommendations

☞ Netflix Prize (2009) It is a competition organized by Netflix that was awarded to a collaborative filtering algorithm that best predicts movie preferences (US$1,000,000)

Content recommendation algorithms have become the most essential and basic marketing technique in the consumer industry. It can be used not only in the media industry, but also in shopping, sports and real estate, and it can be used to collect new data by analyzing the needs of users as well as the consumer industry. By improving and developing these recommended algorithms in the future, they can ultimately be used in the field of artificial intelligence that reads consumers' minds and recommends only the elements they need perfectly.

Introduction of existing system, analysis

-My Closet(application)

어플명:

내옷장

✓ 앱상에서 코디 가능

✓ 유저들 중 전문 코디네이터에게 스타일 연출가능

disadvantages

1. Among the tops, clothes that are not classified as outer or inner clothes are classified as those of the top. You can't style such as layered.

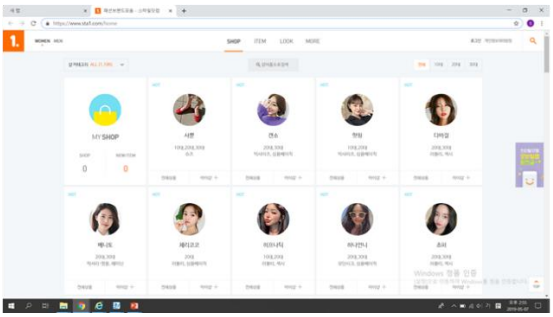
2. Increased user inconvenience due to errors in the app itself

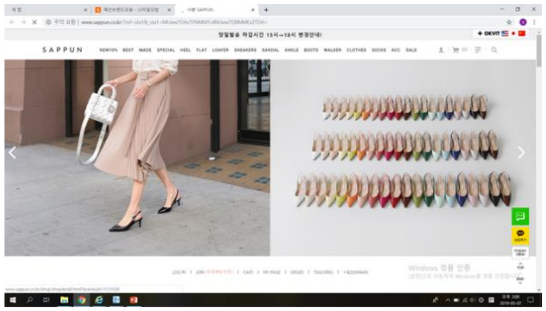
.https://play.google.com/store/apps/details?id=com.cubelab.owncloset

-Style.com(Website)

어플명:

스타일닷컴





advantages

1. There are the largest number of clothing brands among all your apps.
2. Users can analyze clothes most popular among the public by providing the most brand ranking, male shopping mall popularity

ranking, and female shopping mall popularity ranking based on the number of searches

disadvantages

1. Recommend your favorite clothes in color, not in the shape of your clothes

https://www.sta1.com/home

Improvement measures

A key focus of the project is the improvement of the recommended system. There are two major problems that need to be improved now.

Cold Start Problem

First-time visitors are not recommended because there are no similar users
Therefore, on the first visit of a web page, it is recommended that a certain number of codes be graded.

User-Rating Problem

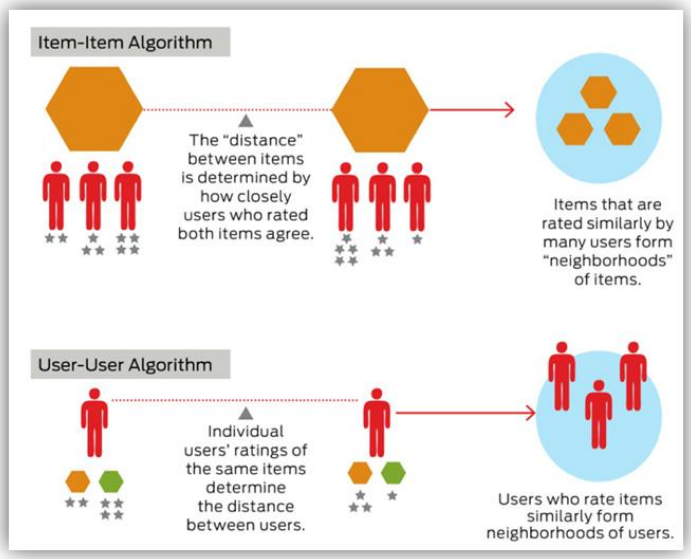
Most users tend to have poor ratings
Personalization is difficult for these people, and Popularity based recommendations.

The recommended algorithm steps are divided into three stages. The first is the simplest step. As the easiest algorithm, it recommends items with the highest popularity, or ratings. I recommend the same item to all users. The next step is to find similarities between users and items using the evaluation between them. Predicting the user's ratings based on the ratings left by users similar to those of a particular user. Numerical prediction of how much the user will like the item.
The last step is personalization, where everyone recommends

items that fit each individual's preferences rather than the same item. User-item evaluation, collaborative filtering algorithm, SVD, etc. are used.

The most commonly used recommend algorithm is collaborative filtering. Collaborative filtering is a filtering technique that receives input from users and automatically predicts and recommends interests. Collaborative filtering is classified by user based collaborative filtering, item based filtering. User based filtering is re classified by active and manual filtering. These filtering algorithms are analysis the data by similarity calculate and then recommend the item.

User based collaborative filtering is use the score by quantifying how much preferences match based on common items among users. For an example user A give the 5score to Z (Blockboster) movie and user B give 5 score to Z movie that means their preference distance is 0. If user B give 3 score their preference distance will be increase. UBCF make this digitization and use the closest distance of similarity based recommendation method. But UBCF has disadvantage, if user use the system at first, there are ant data to use at first so it is hard to recommend, and it takes long time to collect these data. Item based filtering is the way to make better this disadvantages.



IBCF uses valued item object instead of user data into recommend algorithm. In these days almost recommend system use IBCF, the representative company is Netflix and Amazon. Amazon uses the prediction that user will prefer the similar product that they already purchased. IBCF is differ from UBCF depends on initially registered data, when the first time users easily recommended that they wanted product and similar product. But IBCF also consider the similarity with goods,Because preferences among users have not been considered at all, very different preferences from specific customers can result in poor recommendation accuracy among products and poor recommendation ability of the recommendation system.

	Advantage	Disadvantage
User based filtering	Simple algorithm, Recommend without item data	More items more harder, first users data blank space
Item based filtering	Recommendations can be made without the information of the item itself, new user recommend possibility	Bigger data calculate uncomfortable, initial service recommendation accuracy lower
Active filtering	Higher reliability	Narrow affinity, little feedback
Manual filtering	Active filtering variable remove	User reliability

In addition, there are active filtering, which has recently been growing in popularity, or passive filtering, which is considered to have the most potential in the future. The project aims to improve the recommended system by implementing these collaborative filtering directly

Development Title

Fashion recommend system using Big Data

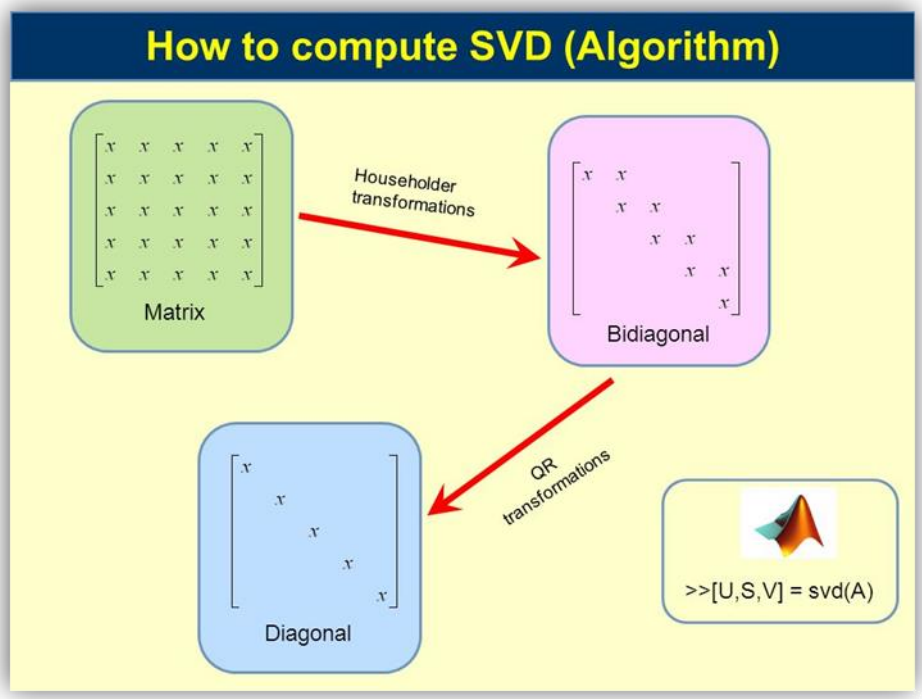
It is meant to develop a system that recommends fashion items by applying various applications of big data.

Project introduction

-theory

Dimension-reduction algorithms are receiving the most attention for complementing the shortcomings of collaborative filtering. There are two notable drawbacks to collaborative filtering. The first is how to respond to massive data, or the nature of big data. In order to analyze and predict massive data as quickly as possible, we need to shrink it. Second is the actual measurement. In organizing collaborative filtering, there may be items in the matrix that the user has not yet evaluated. If these values exist, errors can occur in making predictions using data.

And then how to compensate the disadvantages of this collaborative filtering? Dimension-reduction algorithms have many, but most typically complement the recommended system using an algorithm called SVD. The SVD is the most representative dimension reduction algorithm used in the recommended system, and uses the matrix to perform dimension reduction.



First, it organizes preferences between users and products into numerous datasets and presents them as a matrix. The SVD can then be used to reduce the amount of overall data that needs to be processed by shrinking the diagonal matrix, and the estimated score can be obtained using the average value for data that has not yet been evaluated. The recommendation system uses these scores to recommend items with high recommendation scores to users. Dimension reduction algorithm is generalized by applying dimension reduction to various products, and the more dimension reduction is applied, the more efficient the recommended system works.

In FRUB project our goal is to implement the prototype as a system. Create a sample CSV file with the user's gender, scores evaluated for each item, and calculate the forecast using the SVD supported by the Surprise package of python. Finally, Google web crawling open source is added to search for the item and output the image.

Korean part

서론

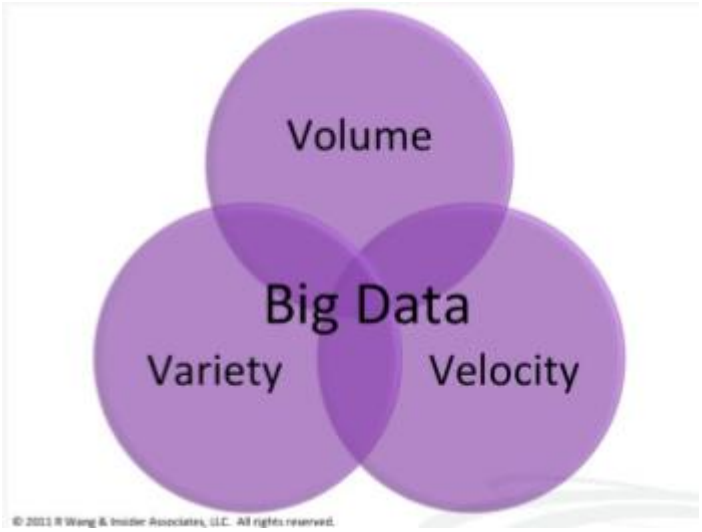
빅데이터에 대한 이해

빅데이터는 현대 4차산업혁명에서 가장 주목받는 중요한 IT산업 키워드이다. 빅데이터는 여러가지 관점에 따라 다양한 정의를 내릴 수 있는데, IDC에서는 2011년 6월 빅데이터가 다양한 종류의 대규모 데이터로부터 저렴한 비용으로 가치를 추출하고, 데이터의 빠른 수집, 발굴, 분석을 지원하도록 고안된 차세대 기술 및 아키텍처라고 업무수행 방식에 초점을 맞춘 정의를 내렸다. 맥킨지에서는 2011년 5월 빅데이터는 데이터의 규모에 초점을 맞췄을 때 기존 데이터베이스 관리도구의 데이터 수집, 저장, 관리, 분석 역량을 넘어서는 데이터라고 정의 했다. 종합하여 평가하자면 디지털 환경에서 생성되는 데이터로 그 규모가 방대하고, 생명주기가 짧고, 형태도 수치 데이터 뿐만 아니라 문자와 영상 데이터를 포함하는 대규모 데이터라고 정의 할 수 있다.

디지털 정보량의 기하급수적인 증가에 따라 데이터의 종류가 다양해져 도로, 건축물 등에 내장된 센서 및 임베디드 시스템, 정형, 비정형, 소설, 실시간 데이터 등의 복합적인 구성을 통해 생각과 의견까지 분석 및 예측 가능한 환경이 구성되고 있다.

빅데이터의 개념

빅데이터의 개념에서 가장 중요한 특징은 3V이다. 3V는 빅데이터의 3가지 가장 주요한 특징의 단어들을 뜻한다. BI/DW 리서치 기관인 TDWI가 빅데이터의 3대 요소를 아래와 같이 그림으로 표현 했다.



Volume(용량)

빅데이터는 대용량의 크기를 가진다.일반적으로 수십 테라바이트(terabyte) 혹인 수십 페타바이트(petabyte) 이상이 빅데이터의 범위에 해당한다. 1 페타바이트는 5기가바이트 DVD 영화를 약 17만 4천편 담을 수 있는 용량이다. 빅데이터는 기존 파일 시스템에 저장하기 어려울 뿐만 아니라 데이터 분석을 위해 사용하는 기존 DW(데이터 웨어하우스) 같은 솔루션에서 소화하기 어려울 정도로 급격하게 데이터의 양이 증가하고 있다. 이러한 문제를 극복하기 위해 확장 가능한 방식으로 데이터를 저장하고 분석하는 분산 컴퓨터 방식이 필요하다. 현재 사용 되는 분산 컴퓨팅 방식에는 아파치의 하둡이 대표적이다.

Velocity(속도)

속도는 두가지의 해석이 가능하다. 첫번째는 데이터의 빠른 생성 속도를 뜻하고 두번째는 데이터의 처리 속도의 개념이다. 사물 데이터, 웹 검색 데이터, 실시간 전송 데이터, 모바일에서 생성되는 데이터들은 지속적이고 빠른 속도로 생성되는 경우가 많다. 이러한 빅데이터들은 빠르게 분석되고 처리 되어야 한다. 오늘날 디지털 데이터는 매우 빠른 속도로 생성 되기 때문에 데이터의 수집, 저장, 분석 등이 실시간으로 처리돼야 한다.

Variety(다양성)

빅데이터의 3V중 마지막 개념은 데이터의 다양성이다. 데이터는 그 특징에 따라

정형, 반정형, 비정형으로 구분 된다. 정형 데이터는 고정된 필드에 저장되는 데이터를 의미하며 일정한 형식을 갖추고 있다. 반정형 데이터는 고정된 필드는 아니지만 메타데이터나 스키마등을 포함한다. 비정형 데이터는 고정된 필드에 저장되지 않는 데이터를 뜻한다. 사진, 동영상, 위치 정보, 통화 내용 등이 해당된다.

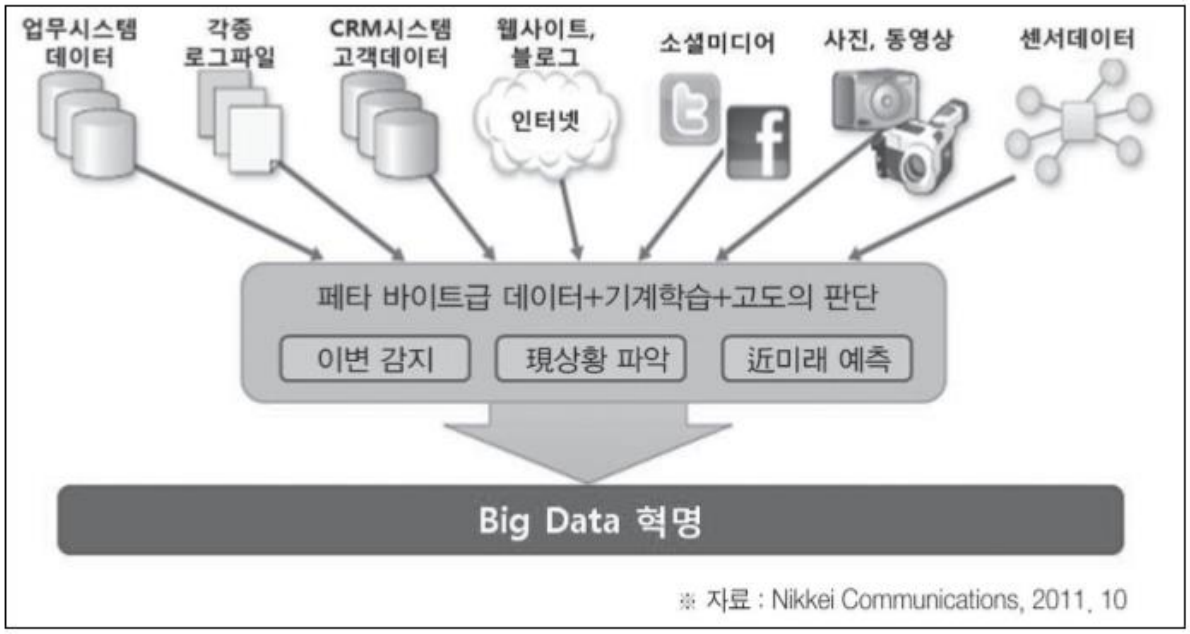
정형 데이터	비정형데이터	반정형데이터
고정된 필드에 저장된 데이터 (형식이 있음)	형태와 구조가 복잡한 데이터	값과 형식이 다소 일관성이 없는 데이터
.관계형 DB(RDB) .스프레드시트	.소셜데이터(SNS) .문서 .이미지 .비디오 .동영상	.HTML .XML .웹문서 .웹로그 .센서 데이터

이외에도 Value(가치)라는 새로운 속성이 더해져서 4V라고 불리기도 한다.

빅데이터의 적용

빅데이터의 기술에는 크게 4가지의 단계가 있다. 수집, 저장 및 관리, 처리 및 분석, 시각화 및 활용단계이다. 먼저 새로운 데이터를 생성 및 외부 데이터를 수집 하고 데이터를 통합 하거나 데이터의 형태와 소재에 관련없이 데이터를 확보한다. 다음으로 거대한 형식의 데이터를 실시간 저장, 관리할 수 있는 분산 컴퓨팅 저장 기술이 필요하다. 저장 된 데이터의 가치를 분석하기 위해 대규모 통계, 예측 처리, 데이터 마이닝등의 분석 방법이 필요하고 머신러닝 및 인공지능을 활용한 심층 분석기술이 필요하다. 마지막으로 비전문가가 데이터 분석을 수행할 수 있게 환경을 제공하는 분석 도구 모듈을 개발하거나, 분석 결과를 표시하고 직관적인 정보를 제공하기 위해 그래픽적 기술로 구성하는 시각화가 필요하다.

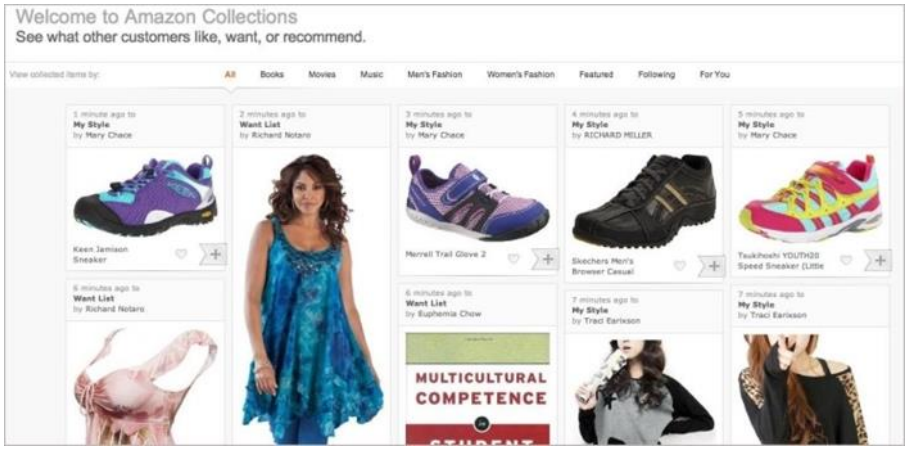
이렇게 빅데이터가 출현하고 갑자기 주목받는 배경에는 여러가지 이유가 있다. 가장 큰 영향은 스마트폰과 같은 모바일 기기의 보급과 소셜네트워크 서비스(SNS)의 유행으로 엄청난 양의 데이터가 생산되고 있다. 빅데이터는 기업,정부, 교육,문화 분야에서 그 가치가 입증되고 있으며 빅데이터를 적극적으로 활용하는 기업은 이미 비즈니스적 성과를 보이며 빅데이터의 효과를 입증하고 있다.



빅데이터 안에서 소비자의 의식이나 행동을 깊이 있게 관찰하고 소비자의 스스로 자각하지 못하는 속마음, 경향, 트렌드를 꿰뚫어보아서 의미 있는 해석을 도출하는 것이 본 프로젝트에서 가장 주목하는 기술이다.

추천 시스템

추천 시스템(recommend system)은 현대 상업시장에서 가장 강력하고 필수적인 전략 중 하나이다. 사용자(User)는 모든 아이템들을 다 둘러볼 시간이 없고, 자신에게 가장 필요로 하는 상품을 추천 해 주길 원한다. 추천 시스템을 적용한 웹 쇼핑사이트들은 자신들의 알고리즘을 이용하여 사용자들에게 적합한 아이템을 추천해준다. 사용자들이 기존에 구매했던 상품들의 속성이나, 평점을 분석하여 관련 있는 상품들을 추천 리스트로 작성해 보여준다.



추천시스템을 가진 회사중 가장 유명한 회사는 단연 아마존이다. 아마존은 회원들의 소비 패턴을 분석해 구매 가능한 상품을 추천하는데, 아마존 성장의 일등공신으로 아마존 매출의 약35%가 추천 상품에서 발생한다. 아마존은 고유의 아이템기반 협업필터링 알고리즘을 'A9'이라 부르고 특허를 등록하였다.

이러한 추천 시스템을 적용한 웹시스템을 개발하기 위해 빅데이터 분야의 요소를 접목시킨다. 빅데이터 수집단계를 통해 데이터를 수집하고, 분석단계를 통해 데이터를 파싱, 정규화 하며 활용단계에서 수집한 데이터들을 이용하기로 한다. 이렇게 빅데이터의 요소를 웹 사이트와 알고리즘을 구현하며 직접 학습해보기로 한다.

본론

개발동기

온라인 컨텐츠 시장이 성장함에 따라 소비자들은 더욱 많은 선택권을 가지게 되었다. 하지만 그만큼 모든 컨텐츠를 소모하기엔 시간적 여유가 따르지 않기 때문에 소비자들은 제한된 시간을 효율적으로 소모하기 위해 제한적으로 컨텐츠를 고를 수 밖에 없다. 대부분의 멀티미디어 서비스 웹사이트들은 이런 소비심리를 전략적으로 이용하기 위해 대부분 컨텐츠 추천 알고리즘을 사용하고 있다.

최근 가장 높은 성장률을 보이고 있는 넷플릭스(Netflix)는 대표적인 멀티미디어 컨텐츠 서비스 사이트이다. 넷플릭스는 강력한 컨텐츠 추천알고리즘을 사용하여 사용자(User)가 시청한 영상물들을 분석하여 트렌드를 파악하고, 취향에 맞는 컨텐츠를 추천해준다. 또는 다른 유저들이 많이 즐긴 컨텐츠들을 리스트로 상단에 노출시키는 마케팅을 하기도 한다

- **Netflix :** 대여되는 영화의 **2/3** 가 추천으로부터 발생
- **Google News : 38%** 이상의 조회가 추천에 의해 발생
- **Amazon :** 판매의 **35%** 가 추천으로부터 발생
- **Netflix Prize (~2009) Netflix** 에서 주관하는 경연대회로, 영화 선호도를 가장 잘 예측하는 협업 필터링 알고리즘에 수상 **(US$1,000,000)**

컨텐츠 추천 알고리즘은 소비산업에서 가장 필수적이면서 기본적인 마케팅 기법이 되었다. 미디어 산업뿐만 아니라 쇼핑, 스포츠, 부동산등 여러 분야에서 활발히 사용되고 있고 소비산업 뿐만 아니라 사용자들의 니즈를 분석하여 새로운 데이터를 수집할 수 있다는 점에서 다양하게 사용될 수 있다. 앞으로 이런 추천 알고리즘을 개량 시키고 발전시킴으로써 궁극적으로 소비자의 마음을 읽어 완벽하게 필요로 하는 요소들만 추천하는 인공지능 분야에도 사용될 수 있다.

기존 시스템 소개,분석

-내옷장(어플)

어플명:

내옷장

✓ 앱상에서 코디 가능

✓ 유저들 중 전문 코디네이터에게 스타일 연출가능

단점

1.상의 중에서도 아우터, 이너 이런 것으로 분류되는 것이 아닌 상의 옷이면 그대로 상의 옷으로 분류되어 레이어드 등 코디 불가

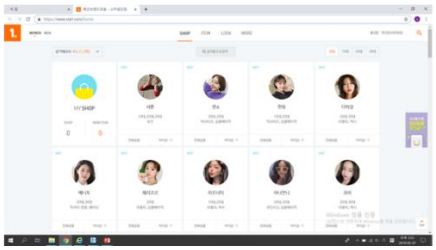
2.앱 자체에서의 오류로 인한 user 등의 불편 증가

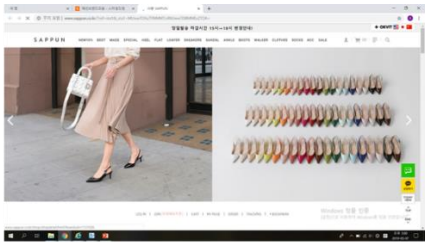
https://play.google.com/store/apps/details?id=com.cubelab.owncloset

-스타일닷컴(웹사이트)

어플명:

스타일닷컴





장점

1.유사 앱 중에서 가장 많은 의류 브랜드가 있다.

2.검색 수 기준으로 가장 브랜드 순위, 남자 쇼핑몰 인기 순위, 여자 쇼핑몰 인기 순위 등을 제공함으로써 user가 가장 대중들이 선호하는 의류 분석 가능

단점

1.자신이 선호하는 의류들을 의류의 모양 등이 아닌 색상 위주로 추천

https://www.stal.com/home

개선방안

프로젝트에서 핵심으로 주목한 부분은 추천 시스템의 개선이다. 현재 개선해야 할 문제점은 크게 2가지로 볼 수 있다.

Cold Start Problem

첫 방문한 사용자는 유사한 사용자가 없기 때문에 추천이 되질 않음

->따라서, 웹페이지 첫 방문 시, 일정 개수 이상의 코디에 평점을 남기는 것을 유도

User-Rating sparsity Problem

대부분의 사용자는 평점을 잘 남기지 않는 경향이 있음

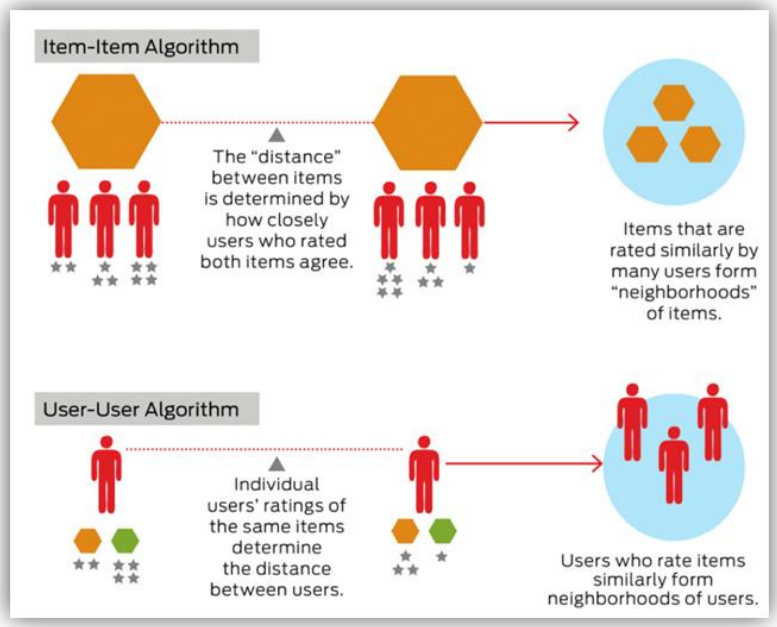
->이들에게는 개인화가 어려움, Popularity based 추천

추천 알고리즘의 단계는 3가지로 나누어 진다. 첫번째는 가장 단순 단계(Popularity)이다. 가장 쉬운 알고리즘으로, 제일 높은 인기도, 즉 평점을 갖는 아이템을 추천해준다. 사용자 모두에게 동일한 item을 추천한다. 그 다음으로 중간단계(Collaborative Filtering)는 사용자와 아이템간의 평가를 이용하여 사용자끼리의 유사도를 찾는다.특정 사용자와 유사한 사용자들이 남긴 평점을 기반으로 해당 사용자의 평점을 예측한다. 해당 사용자가 해당 아이템을 얼마나 좋아할 것인지 수치적으로 예측하는 것을 의미한다.

가장 마지막 단계는 Personalization(개인화)단계로 모두에게 동일한 아이템이 아닌 각 개인의 성향에 맞는 아이템을 추천한다. user-item평가, 협업 필터링 알고리즘, SVD등이 사용된다.

현재 가장 많이 사용되고 있는 추천 알고리즘은 협업 필터링이다. 협업 필터링은 사용자들로부터 정보를 입력 받아 관심사들을 자동으로 예측하여 추천하는 필터링 기법이다. 협업 필터링은 유저 기반 협업 필터링(User-based-CF), 아이템 기반 필터링(Item-based-CF)로 구분된다. 유저 기반 협업 필터링은 다시 능동적, 수동적 필터링으로 분류된다. 이러한 필터링 알고리즘들은 데이터를 분석하여 유사도 계산을 통해 아이템을 추천하게 된다.

유저 기반 필터링(UBCF)는 사용자들간 공통된 아이템을 기준으로 얼마나 선호도가 일치하는지를 수치화 하여 그 점수를 사용한다. 예를 들어 A가 Z라는 블록버스터 영화에 평점5점을 주고, B가 5점을 준다면 둘의 선호도도 점수의 거리는 0이라고 할 수 있다. 만약 B가 3점을 줬다면, 두 사용자의 거리는 증가하게 된다. UBCF는 이것을 수치화 하여 더 가까운 유사도를 활용한 추천 방식이라고 볼 수 있다. 그러나 UBCF는 단점을 가지고 있는데, 만약 사용자가 최초로 시스템을 이용하는 거라면, 필요한 초기 데이터가 없기 때문에 추천을 하기가 어렵고, 또한 이런 데이터를 수집하는데 많은 시간이 소요 될 수 있다. 이러한 단점을 보완하는 것이 아이템 기반 필터링 이다.



아이템 기반 필터링(IBCF)는 추천 알고리즘에 사용자 데이터 대신 평가된 아이템 객체들이 데이터로 사용된다. 오늘날 대부분의 추천 시스템은 IBCF를 사용하고 있는데, 가장 대표적인 곳이 넷플릭스와 아마존이다. 아마존에서는 사용자가 구매한 상품들과 유사한 상품들을 선호할 것이라는 예측을 기반으로 하고 있다. IBCF는 UBCF와 달리 초기에 등록한 객체들의 데이터에 의존하기 때문에, 처음 시스템을 접하는 사용자라도 원하는 상품과 유사한 상품들을 쉽게 추천 받을 수 있다. 그러나 IBCF역시 상품들과의 유사도를 고려하였지만, 사용자들간의 선호도가 전혀 고려되지 않았기 때문에 특정 고객과 선호도가 매우 다르다면 상품들간의 추천 정확도가 떨어지고, 추천 시스템의 추천 능력이 떨어질 수 있다

	장점	단점
유저 기반 필터링	알고리즘이 간단함, 아이템 정보 없이 추천이 가능	데이터가 많아질수록 연산이 복잡해짐, 신규 사용자의 데이터 공백
아이템 기반 필터링	아이템 자체의 정보 없이 추천 가능, 신규 사용자 추천 가능	데이터가 커질수록 연산이 복잡해짐, 초기 서비스 추천 정확도 떨어짐
능동적 필터링	신뢰성이 높음	편협 적인 유사도, 적은 피드백
수동적 필터링	능동적 필터링의 변수 제거	사용자 의존적

이외에도 최근에 인기가 증가하고 있는 능동적 필터링이나, 미래에 가장 잠재력이 있다고 여겨지는 수동적 필터링 방법들도 존재한다. 본프로젝트에서는 이러한 협업 필터링을 직접 구현하는 방식으로 추천 시스템을 개선시키기로 한다.

개발 제목

Fashion recommend system using BigData

빅데이터를 이용한 패션 아이템 추천 시스템

빅데이터의 여러가지 적용분야를 접목하여 패션 아이템을 추천해주는 시스템을 개발한다는 뜻을 가지고 있다.

프로젝트 소개

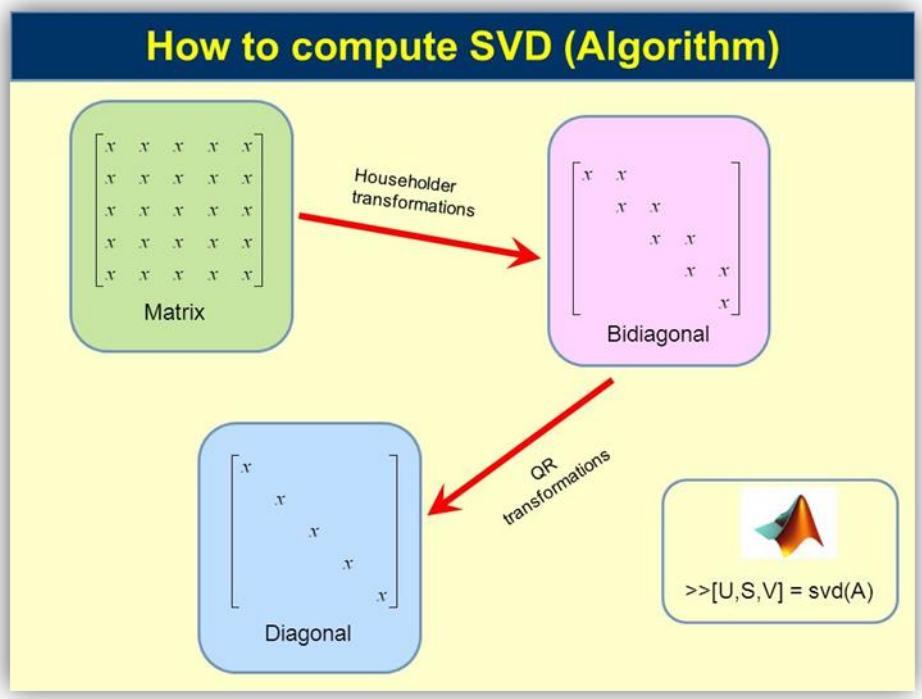
프로젝트 세부내용

이론

협업 필터링들의 단점들을 보완 하는 것에는 차원 축소 알고리즘이 가장 주목받고 있다. 협업 필터링의 주목할 단점은 2가지가 존재한다. 첫번째로 거대한 데

이터 즉, 빅데이터의 성질에 대해 어떻게 대응하냐이다. 거대한 데이터를 최대한 빨리 분석하고 예측 하기 위해서는 데이터를 축소할 필요가 있다. 두번째로는 결측치이다. 협업필터링을 구성하면서 매트릭스에는 사용자가 아직 평가 못한 아이템이 존재할 수 있다. 이러한 결측값들이 존재하면 데이터를 사용하여 예측하는데 오차가 발생할 수 있다.

그렇다면 이러한 협업 필터링들의 단점들을 어떻게 보완할 수 있을까? 차원 축소 알고리즘들은 여러가지가 있지만 가장 대표적으로 SVD(특이값분해)라는 알고리즘을 사용하여 추천 시스템을 보완하도록 한다. SVD는 추천 시스템에 쓰이는 가장 대표적인 차원 축소 알고리즘으로, 매트릭스를 이용하여 차원 축소를 수행한다.



먼저 사용자와 상품 간의 선호도를 수많은 데이터셋으로 정리하고 매트릭스로 표현한다. 이때 SVD를 사용하여 대각 행렬을 축소하여 처리해야 하는 전체 데이터의 양을 줄일 수 있고, 아직 평가하지 않은 데이터에 대해서 평균 값을 이용하여 예상점수를 구할 수 있다. 추천 시스템은 이 점수를 이용하여 추천 점수가 높은 아이템을 사용자에게 추천하게 된다. 차원 축소 알고리즘은 다양한 상품에 대

해 차원 축소를 적용함으로써 일반화 현상이 발생되고, 차원 축소가 많이 적용될 수록 추천시스템은 더 효율적으로 작동한다.

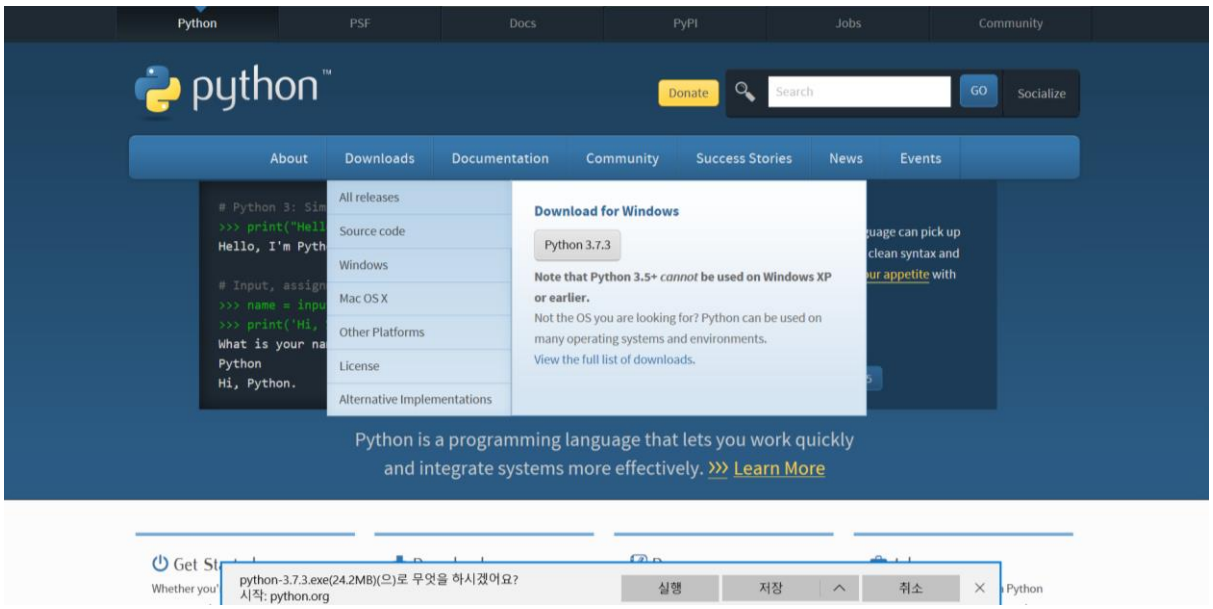
FRUB프로젝트에서는 위 시스템을 프로토타입으로 구현하는 것을 목표로 한다. 사용자의 성별, 각 아이템에 대해 평가한 점수를 가지고 있는 샘플 CSV파일을 생성하고, 파이썬의 서프라이즈패키지에서 지원하는 SVD를 사용하여 예측치를 계산한다. 마지막으로 구글 웹 크롤링 오픈소스를 추가하여 해당 아이템을 검색하여 이미지를 출력한다.

일정표

	1주	2주	3주	4주	5주	6주	7주	8주	9주	10주	11주	12주	13주	14주	15주
자료 조사															
정규 토의															
제안서,보고서 작성															
개념 설계															
상세 설계															
1차 구현															
2차 구현															
테스트 및 보완															
리허설 및 유지보수															
비고			주제 발표					중간 발표		1차 구현 발표		2차 구현 발표			최종 발표

개발 환경

파이썬 -> 플랫폼이 독립적이고 인터프리터식으로 만들어져 있으며, 객체지향적, 동적 타이핑 대화형 언어로 사용된다.데이터분석에 특화된 언어이다.

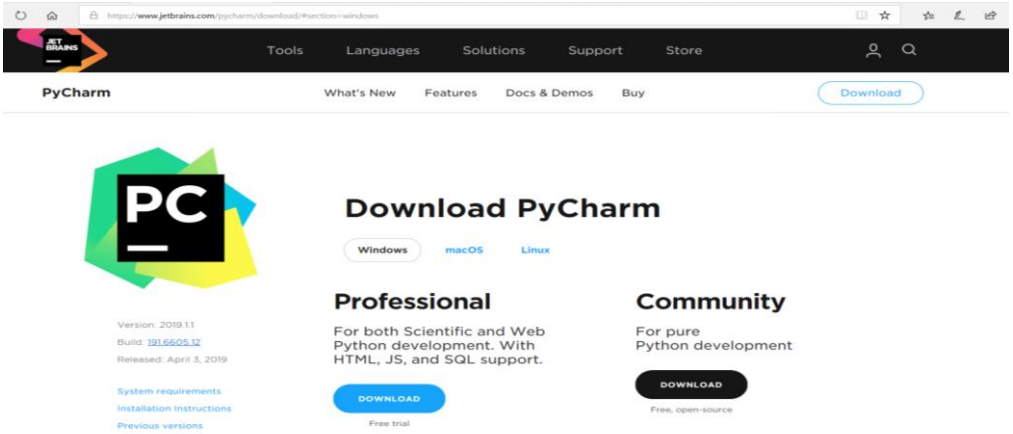


설치 방법은 파이썬 다운로드링크는 https://www.python.org/ 이고,링크에 들어가면 위 그림과 같은 페이지에 접속된다.접속후 downloads카테고리를 선택한후 자신의 환경을 선택하여 python 3.7.3 을 클릭하여 파이썬을 실행시켜준다.

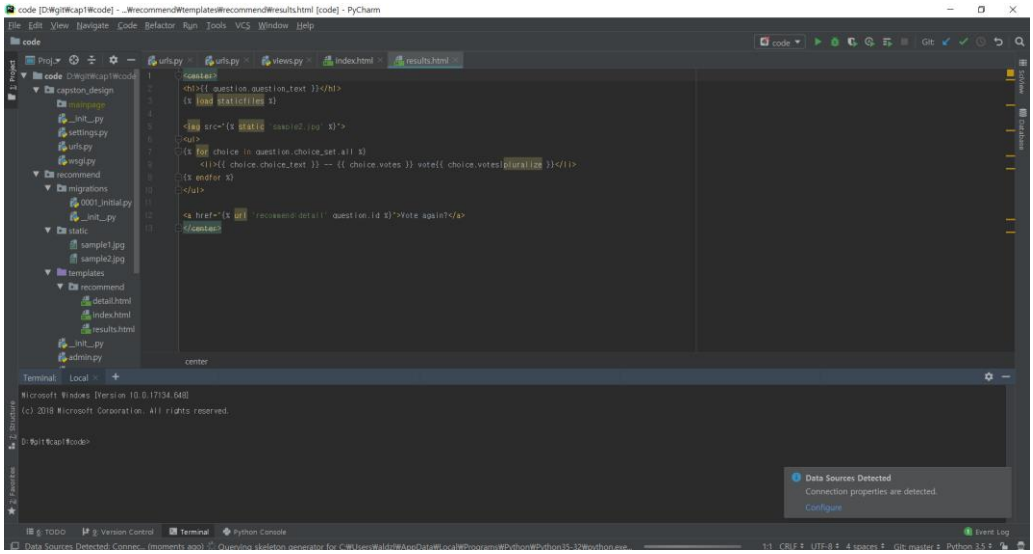
- 파이참 설치

파이참은 Python 개발에 필요한 모든 도구를 제공해 주는 역할을 한다.파이참은 다운로드시 별도의 제약 없으며 다운로드후 ->power shell 에서 장고 설치하기위해 pip install Django 를 입력하며 개발환경을 만들어준다.

다운로드 방법은 https://www.jetbrains.com/pycharm/download/#section=windows 링크에 접속하여 화면에 보이는Download를 클릭하여 Community Download를 눌러준다. Professional Download를 선택시 시간이 흐른 뒤 유료화 되기 때문에 community를 다운받아준다.

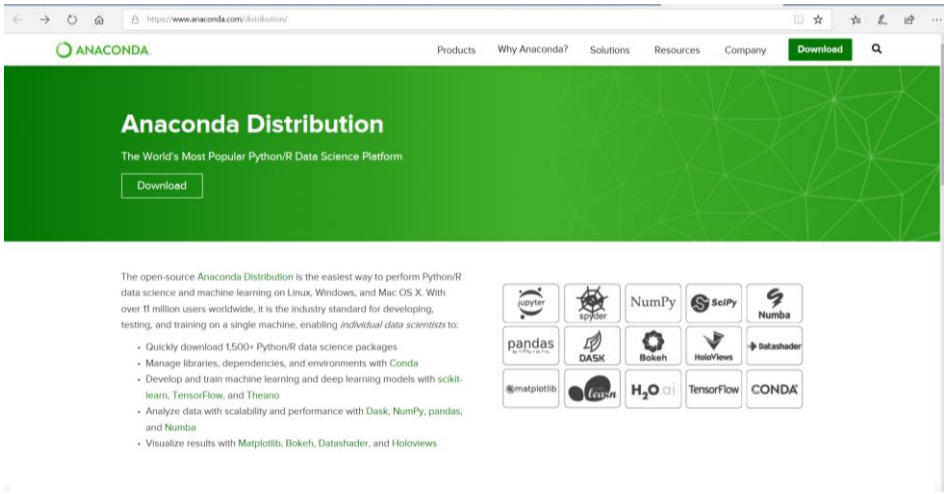


다운로드가 완료되고 실행시키면 pycharm 실행화면이 나온다.

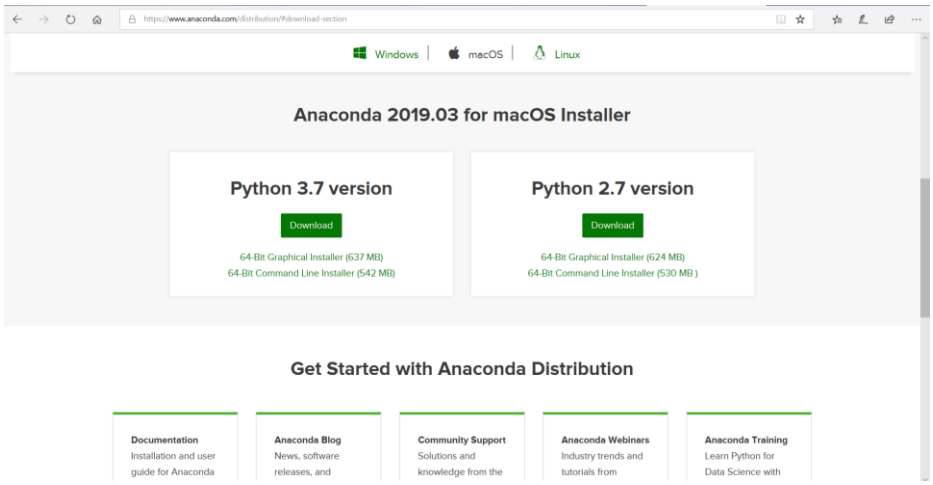


- 아나콘다 네비게이터 ,

Anaconda 는 데이터 분석용으로 사용되며, 패키지관리를 단순화하고 과학 컴퓨팅을 위한 Python및 R 프로그래밍 언어를 위한 무료 및 오픈소스 배포판이다. **아나콘다 설치를 위해** https://www.anaconda.com/distribution/ **링크에 클릭해주면,** 위와 같은 페이지에 접속이 된다.



여기서 Download버튼을 클릭 하고 각자 필요한 version을 클릭하여 다운로드하여 설치해준다



개발 현황

프로젝트를 진행하기 위해 필요한 프로그램을 설치하고 환경을 구성한다. 이후에 데이터를 수집하고 활용하기 위해 파이썬을 선택한다. 파이썬은 데이터를 분석하는데 특화된 프로그래밍 언어로 다양한 패키지를 제공하고 있다. 우선 아나콘다 네비게이터에서 제공하는 주피터 노트북으로 데이터를 관리하고 필요한 패키지를 설치해서 함수를 작성한다. 코디 사진을 수집하기 위한 웹 크롤러 패키지를 설치하고, SVD를 사용하기위해 서프라이즈 패키지를 import 한다.

코드 해석

```
'''
#itemid
1 = shorts
2 = caps
3 = T-shirt
4 = sandals
5 = sweater
6 = cardigan
7 = raincoat
8 = coat
9 = gloves
10 = jeans
11 = sneakers
12 = dress
13 = suit
14 = pants
15 = vest
16 = blouse
17 = jacket
18 = sunglasses
19 = bag
20 = shoes
'''
```

	userId	gender	itemId	rating
2	1	2	16	3
3	1	2	11	1
4	1	2	8	1
5	1	2	15	1
6	1	2	19	5
7	1	2	20	3
8	1	2	12	4
9	1	2	4	2
10	1	2	13	4
11	1	2	1	4
12	1	2	3	3
13	1	2	9	1
14	2	1	1	4
15	2	1	10	3
16	2	1	13	3
17	2	1	16	3
18	2	1	11	1
19	2	1	12	3
20	2	1	3	4
21	2	1	6	2
22	2	1	5	2
23	2	1	8	4
24	2	1	9	2
25	3	2	3	4
26	3	2	7	5
27	3	2	9	2
28	3	2	8	2
29	3	2	13	1
30	3	2	16	4
31	3	2	18	1

사용되는 데이터는 미리 준비한 사용자-아이템 평가 데이터셋이다. 데이터는 userId, gender, itemId, rating속성을 가지고 있고 csv형식으로 저장 되어있다. 각 아이템 아이디는 각자 정해진 정수형 코드를 가지고 있고 사용자들은 그 아이템에 대하여 평가를 최소 1점에서 최대 5점까지 줄 수 있다. 또한 사용자들의 성별이 남자일경우 1, 여자일경우 2를 가지게 된다. 프로그램이 실행됨에 따라 파일에 새로운 사용자와 아이템에 대한 평가데이터가 추가된다.

```
# package import
from google_images_download import google_images_download
from imageai.Prediction import ImagePrediction
from surprise import Reader, Dataset, SVD, evaluate
from PIL import Image
from io import BytesIO
import pandas as pd
import csv
import random
import requests
import warnings

warnings.filterwarnings(action='ignore')

# 아이템 키워드 리스트 작성
item_keyword = ['shorts','caps','T-shirt','sandals','sweater',
                'cardigan','raincoat','coat','gloves','jeans',
                'sneakers','dress','suit','pants','vest',
                'blouse','jacket','sunglasses','bag','shoes']
# 사용자-아이템 평가 csv 파일
ratings = pd.read_csv('item_rating.csv')
```

필요한 모듈들을 import한다. itme_keyword변수에 기존에 정한 아이템들의 키워드들을 리스트형식으로 순서대로 저장한다. pandas에서 제공하는 read_csv() 함수로 기존 csv파일을 rating변수에 바인딩한다.

```
def login():
    global gender

    # 성별입력, 새로운 유저 아이디 생성
    gender = input('성별을 입력해주세요: ')
    new_userId = int(ratings.loc[len(ratings.index) - 1]['userId']) + 1
    print("userId: " + str(new_userId) + "₩n성별: " +gender)

    #rand_recommend(gender,new_userId)
```

login 함수는 새로운 사용자의 정보를 저장한다. 성별을 입력받고 기존 csv파일의 맨 아래의 인덱스의 userId의 값에 +1 하여 새로운 userId를 부여한다. (ex: 맨 마지막 인덱스의 UserID가 30이면 30+1=31을 부여한다.)

```
def rand_recommend(gender,new_userId):
    global ratings

    # 반복문으로 원하는 만큼 아이템을 추천 받는다.
    r_num = int(input('추천 받을 횟수를 입력 하세요: '))
    for a in range(1,r_num+1):
        # 아이템 리스트중에서 랜덤하게 하나를 추천
        item_Id = random.randint(1,20)
        keywords = str("20대 " + gender + " " + item_keyword[(item_Id)-1] + " 추천")
        response = google_images_download.googleimagesdownload()
        arguments = {"keywords":keywords,"limit":1,"output_directory":'image',"no_download":True, "safe_search":True}
        paths = response.download(arguments)

        # 추천받은 아이템에 대한 점수 평가
        while True:
            itemscore = int(input('점수를 평가해주세요(1~5점): '))
            if 1 <= itemscore and itemscore <= 5:
                break
            else:
                print('warning: 1점에서 5점사이의 값을 입력하세요')
        if gender == '남자':
            new_gender = 1
        else: new_gender = 2
        new_itemId = item_Id
        new_rating = itemscore

        # 평가한 점수를 기존 csv파일에 추가한다.
        ratings.loc[len(ratings.index)] = [new_userId, new_gender, new_itemId, new_rating]
        ratings = ratings.set_index("userId")
        ratings.to_csv('item_rating.csv')
        ratings = pd.read_csv('item_rating.csv')

    #svd_ratings(ratings,new_userId)
```

rand_recommend 함수는 무작위로 사용자에게 아이템을 추천해준다. 사용자가 원하는 추천 횟수를 입력 받는다. 위코드에서 입력했던 아이템 리스트에서 무작위로 하나를 추출하여 키워드를 만들고 검색하여 이미지 url을 사용자에게 보여준다. 사용자는 해당 링크에 대해 점수를 입력하고 평가점수를 csv파일에 새로운 레코드로 저장한다.그리고 반복문을 종료하고 루프 처음으로 돌아간다.

```
def svd_ratings(ratings,new_userId):
    reader = Reader()
    svd = SVD()

    # 데이터셋 학습시키기
    data = Dataset.load_from_df(ratings[['userId', 'itemId', 'rating']], reader)
    trainset = data.build_full_trainset()
    svd.fit(trainset)

    # 사용자가 평가를 안내린 아이템들 중 무작위로 3개의 점수를 예측한다.
    lst_itemId = list(range(1,21))
    lst_ratings = list(ratings[ratings.userId == new_userId]['itemId'].values)
    notrating_itemId =list(set(lst_itemId)-set(lst_ratings))
    rand_itemId = random.sample(notrating_itemId,3)
    a = 0
    svd_rating = []

    # 평가를 안내린 아이템, 그 아이템의 예측 점수를 각각 key와 value로 dic생성
    for i in range(1,4):
        svd_rating.append(round((svd.predict(new_userId,rand_itemId[a]).est), 1))
        a += 1
    rating_dic = dict(zip(rand_itemId,svd_rating))

    # 3개의 항목중에 가장 높은 예측 값을 가진 아이템의 코드를 추천한다
    recommend_itemId = max(rating_dic, key=rating_dic.get)

    #final_recommend(recommend_itemId)
```

svd_ratings함수는 svd라이브러리에서 제공하는 예측 계산 함수를 사용한다. 우선 ratings변수를 학습시키고 svd메서드에 fit시킨다.기존에 설정한 아이템리스트에서 사용자가 평가한 항목들을 빼면 사용자가 평가 안한 아이템의 리스트만 남게 된다. 그리고 그 리스트를 svd.predict함수로 예측값을 구성하고, 각각 아이템과 예측값을 key와 value로 묶어 dic자료형 변수로 생성하고 max()로 최댓값을 추출한다. 즉 '사용자가 평가를 아직 안내린(결측치) 아이템에 대한 예측값중에 가장 큰값을 가진 아이템 코드가 recommend_itemId에 담기게 된다.

```
def final_recommend(recommend_itemId):

    # 추천 받은 코드로 키워드를 생성하여 아이템을 추천한다.
    keywords = str("20대 " + gender + " " + item_keyword[(recommend_itemId)-1] + " 추천")
    response = google_images_download.googleimagesdownload()
    arguments = {"keywords":keywords,"limit":1,"output_directory":'image',"no_download":True, "safe_search":True}

    print('사용자에게 추천는 아이템입니다')
    paths = response.download(arguments)

    # 추천받은 아이템의 url주소를 파싱하여 img변수에 바인딩
    '''
    url = paths[keywords][0]
    resp = requests.get(url)
    img = Image.open(BytesIO(resp.content))

    '''
```

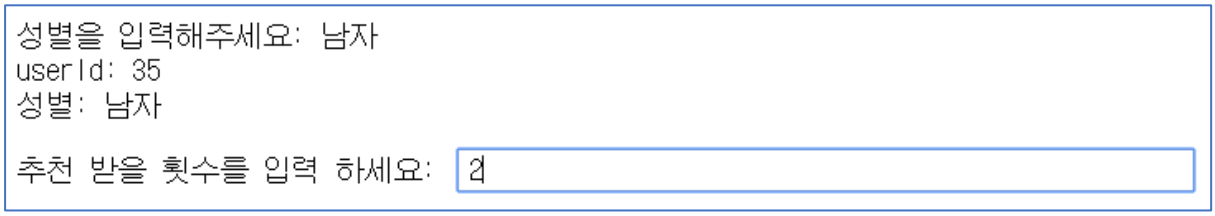
최종적으로 사용자가 아직 평가를 내리지 않은 아이템들의 예측값들 중에 가장 큰 값의 키워드를 검색한 결과를 사용자에게 추천해주게 된다.

실행 화면

251	31	1	8	1	
252	32	2	17	1	
253	33	2	17	1	
254	33	2	3	1	
255	34	1	15	2	
256	34	1	2	1	
257					

프로그램을 실행하기전 CSV파일의 마지막 레코드의 정보. 유저34가 15번과 2번 아이템에 대해 평가를 내린 점수를 보여주고 있다.

새로운 사용자를 추가해보도록 하자. 먼저 성별은 남자로 설정한다.



그러면 새로운 userID 35를 부여해주고 추천 받을 횟수를 요구한다. 2회를 입력하게 되면 2번아이템을 무작위로 추천해준다.

```
Item no.: 1 --> Item name = 20₩ub300 ₩ub0a8₩uc790 pants ₩ucd94₩ucc9c
Evaluating...
Getting URLs without downloading images...
Image URL: https://post-phinf.pstatic.net/MjAxNzA0MjBfMjQ1/MDAxNDkyNjc1MDMzOTEx.AiolneeG1j8RWkiMploy8D1-BRYadH_Fgrmcg.JPEG/10%EB%8C%8020%EB%8C%80%EB%82%A8%EC%9E%90%EC%87%BC%ED5%98%EC%87%BC%ED%95%91%EB%AA%B0_%2810%29.JPG?type=w1200
Printed url without downloading

Errors: 0

점수를 평가해주세요(1~5점): 4

Item no.: 1 --> Item name = 20₩ub300 ₩ub0a8₩uc790 shoes ₩ucd94₩ucc9c
Evaluating...
Getting URLs without downloading images...
Image URL: https://mblogthumb-phinf.pstatic.net/MjAxODA0MTFfNTMg/MDAxNTIzNDI1Q25m2_IJsLmJPbouuVmeaFJJc4hHNGdogUAg.JPEG.grayhomme_/IMG_1824.jpg?type=w800
Printed url without downloading

Errors: 0

점수를 평가해주세요(1~5점): 3
```

각각 pants와 shoes아이템에 대해 4점과 3점을 평가한다.

```
사용자에게 추천는 아이템입니다

Item no.: 1 --> Item name = 20₩ub300 ₩ub0a8₩uc790 shorts ₩ucd94₩ucc9c
Evaluating...
Getting URLs without downloading images...
Image URL: http://www.lemans.co.kr/shopimages/lemans/0050020000243.jpg?1501781116
Printed url without downloading

Errors: 0

프로그램을 종료합니다
```

그러면 프로그램은 최종적으로 사용자의 평점을 분석해서 shorts아이템을 추천해주게 된다.

55	34	1	15	2
56	34	1	2	1
57	35	1	14	4
58	35	1	20	3
59				

CSV파일(DB)을 확인해본 결과 정상적으로 유저35가 각 아이템에 4점과 3점 평점을 준 것이 저장 된 것을 확인할 수 있다. 이제 이 데이터를 이용하여 새로운 유저가 추가되면, 유저35가 아이템에 내린 평가가 SVD 평점 예측계산에 쓰이게 되고 또 새로운 유저가 데이터에 추가되는 식으로 빅데이터의 저장과 처리 분석이 서로 영향을 주며 작동하게 된다.

결론

프로젝트를 진행하며 빅데이터의 개념과 활용 순서를 적용할 수 있었다. 빅데이터는 저장, 처리, 분석, 시각화의 순서로 처리되며,각 단계에서 사용되는 기술이 있다. 추천 시스템은 빅데이터의 기술의 집합체로서 다양한 기술이 적재적소에 활용된다. FRUB프로젝트에서는 빅데이터에서 쓰이는 DB를 샘플로 생성한 csv파일로 예시를 들어 사용하고 SVD기술을 사용하여 처리하고, 결측치를 분석하였다. 또한 데이터의 이미지를 시각화 하여 사용자에게 추천했다. 기존 추천 시스템의 단점이었던 단순추천을 협업 필터링을 사용하여 사용자들의 취향을 예측하여 점수를 부여 하는 방법으로 보완했다. 개선된 시스템을 파이썬으로 프로토타입 코딩으로 작성하여 실제로 동작하는 샘플시스템을 구현해볼 수 있었다.

피드백

FRUB프로젝트에서는 사용자의 취향을 단순 평점(1~5점)으로 분석 하고 있다. 실제 상업에서 쓰이는 추천알고리즘은 더욱 다양한 속성을 사용한다. 본 프로젝트의 패션추천을 예시로 든다면 사용하는 아이템의 색상, 가격, 브랜드, 디자인등의 다양한 속성을 테이블에 추가 함으로서 추천 시스템의 알고리즘을 더 고도화 할 수 있을 것이다.

소감문

20144800 천승현

캡스톤디자인 과목을 시작하며 처음에는 지금까지 해왔던 조별과제와 똑같이 진행하면 될거라고 생각했었다. 하지만 조별과제와는 달리 캡스톤 디자인은 대주제만 주어지고 모든 것을 혼자서 해결해야했다. 자료조사, 프로그램 구현등 지난 4학년 학부동안 학습했던 모든 지식들을 활용해야 한다. 프로젝트 초기에는 시각적인 결과물에 집중해서 자신의 역량을 이해하지 못하고 무리해서 계획을 추진했다. 그러나 학기가 진행되면서 다른 팀들의 발표를 듣고, 교수님의 피드백을 들으며 생각이 바뀌게 됐다. 결과물에 중시하는 것보다 정한 주제의 문제점을 파악하고 어떻게 개선시킬지를 생각하는 과정에 중점을 두기로 하였다. 프로그램을 구현할 때 원하는 요소를 적재적소에 활용하고 설계단계에서 정했던 알고리즘들을 차례대로 구현하는 것이 우선이라 판단하였다. 결국 팀원 모두 만족할 만한 최종 결과물을 얻게 되었고 모든 팀원이 다들 전체 프로젝트를 이해할 수 있었다. 각자 자신 있는 분야에 역할분담을 맡고 서로를 이해하는 과정을 습득하는 것이 캡스톤 디자인의 궁극적인 학습 결과물이라는 것을 알게 되었다.

20165172 이유겸

산학 캡스톤 디자인1 과목을 처음 수강할 때 이 과목은 지금까지 배워왔던 어느 과목과는 다른 모든 것을 교수님이 알려주시는 것이 아닌 학생들이 스스로 또는 학생들끼리 협업을 해서 프로젝트를 진행해야 한다는 점이다. 이러한 이유로 많은 부담감을 느끼게 되었지만 조원들과 같이 프로젝트를 진행해보니 혼자였다면 하지 못할 부분들을 협업을 통해 원활히 진행하게 된 것 같다. FRUB 구현을 하던 중 파이썬 언어로 간단한 코드로 구글에서 원하는 이미지 컨텐츠의 이름을 입력하면 자신이 지정한 폴더에 이미지 파일들이 자동으로 받아지는 크롤링 기술이 가장 흥미로웠던 것 같다. 산학 캡스톤 디자인1 수강을 하기 전에는 과목들은 사실 결과물만 내려고 노력했던 것 같다. 하지만 항상 어떤 과제나 프로젝트를 진행 하려고 하면 결과물만 생각하고 진행하였는데 그럴수록 어떻게 진행을 해야하고 어디서부터 시작해야 하는지를 잘 인지하지 못하고 진행을 하게 되어 과정들이 많이 꼬이거나 잘못된 결과물이 도출하고는 했다. 하지만 이번 수업을 통해 알게 된 balsamiq등을 사용하여 미리 만들어 내야하는 결과물의 프로토타입등을 작성하고 매주 팀원들과 정규 미팅을 통해 만나서 프로젝트 진행에 관한 상황들을 토의하고 매주 수업시간을 통한 발표로 교수님의 피드백을 통해 피드백 받은 부분 등을 개선하고 프로젝트를 진행하게 되니 큰 도움이 됐던 것 같다. 하지만 중간 프로토타입등을 사용을 하거나 구글링을 해봐도 해결이 되지 않던 부분들이 있었지만 조장과 팀원 분들이 많이 도와준 덕분에 FRUB 프로젝트를 진행을 하는데 있어서 큰 무리가 없었던 것 같다. 산학 캡스톤 디자인1 과목을 통해 프로젝트를 진행함에 있어서 자신의 프로그래밍 관련 부족한 부분이 어느 쪽이고 어느 부분에 조금 자신이 있는지를 알게 된 것 같다.

20164216 박지혜

이번 산학 캡스톤디자인 과목에서 프로젝트를 처음 시작했을 때, 가장 관심 있는 주제를 선택하게 되었고 재미있게 참여할 수 있었습니다. 서로 각자 맡은 부분에 대하여 최선을 다하여 참여했고, 선배들의 말을 들으며 엄청난 걱정을 했던 것에 비하면 생각보다 순조롭게 진행되었던 것 같습니다. 프로젝트 진행 초반에는 아무런 결과물 없이 흰 바탕에서 시작하는 것이라 막막했습니다. 그렇기에 매 수업 교수님의 피드백을 집중해서 들었고, 하나하나 다 적어가며 그대로 고치면서 프로젝트를 진행하였습니다. 그렇게 중간발표 때는 웹 페이지를 만들 수 있었고, 최종 발표 때는 완성된 웹 페이지를 만들 수 있다고 생각했습니다. 하지만 매번 발표 때마다 저희 조 뿐만 아니라 다른 조에 대한 교수님의 피드백을 들으며 생각해 본 결과, 너무 결과만을 위해 진행하는 것 같았습니다. 교수님의 말씀을 토대로 너무 결과에 집중하지 않고, 결과보다는 과정에 조금 더 집중하기로 결정을 하였습니다. 그래서 최종 발표를 앞두고 과감히 웹 페이지에 적용하기는 하지 않기로 결정하였습니다. 대신 소스코드를 이용해 저희가 진행하려는 프로젝트를 좀 더 보기 쉽게 설명해드리며, 그것을 결과로 보여드리기로 하게 되었습니다. 그로 인해 프로젝트의 제목도 FCMW에서 FRUB라고 변경하게 되었고, 처음에는 당황스러웠지만 하루 이틀 프로젝트를 고치며 진행하다 보니 전 보다 더 제목에 맞는 결과물을 낼 수 있었던 것 같습니다.

20164228 최예솜

처음 조원들과 패션 코디 추천 웹 사이트를 만들겠다는 목표로 한 학기를 달려왔습니다. 많은 지식을 가지고 있지도 않고 조원들 과도 어색했던 날들이 있었지만 매주 같이 공부하며 서로 도움을 주며 FCMW(Fashion coordy for 20's man&woman inWebsite) 웹사이트 구축에 노력해왔습니다.파이썬을 이용한 파이참 프로그램에 대해 더 심도 있게 공부도 해보고 빅데이터 시대에 맞게 개인화를 중점으로 사회에 이용이 될 수 있고 누군가에게 필요한 결과물을 만들어 내고 싶었습니다. 그래서 더욱 열정적으로 공부하고 참여했던 거 같습니다. 중간발표 이후 시각화된 웹페이지를 보며 뿌듯함도 느끼고 정말 한 페이지에 여러 기능들과 코드들이 들어가고 부족한 부분도 많다고 느끼며 발전해 나가야 겠다는 생각으로 열심히 수정해가며 매시간 교수님의 말씀을 하나하나 새겨들며 결과보단 과정이 더 중요하며 작은 것 하나 놓치지 말고 단단하게 해 나가기로 마음을 먹었습니다. 그래서 빅데이터 처리과정을 중점으로 분석은 svd알고리즘을 사용하여 데이터를 분해하는 데에 중점을 두고, 처리는 csv샘플 파일로 분석한 자료로 추천 시스템까지 가며 개인의 취향에 맞게 추천하는 과정을 정확히 인지하였습니다. 원리 이해에 맞는 제목도 FRUB(Fashion Recommend system Using BigData)로 변경하며 저희 방향을 확실하게 정해 가며 나아갔습니다. 한 학기 동안 빅데이터를 가지고 코디 추천을 원리 이해와 개선방법 모색이라고 마무리를 지었지만 이 마무리가 있기까지는 많은 언어 공부와, 설계 방법, 빅데이터 활용 방법 등을 배워 더 성장할 수 있는 한 학기였습니다.

부록

전체코드

전체 코드

```
from google_images_download import google_images_download
from imageai.Prediction import ImagePrediction
from surprise import Reader, Dataset, SVD, evaluate
from PIL import Image
from io import BytesIO
import pandas as pd
import csv
import random
import requests
import warnings

warnings.filterwarnings(action='ignore')



item_keyword = ['shorts','caps','T-shirt','sandals','sweater',
                'cardigan','raincoat','coat','gloves','jeans',
                'sneakers','dress','suit','pants','vest',
                'blouse','jacket','sunglasses','bag','shoes']

ratings = pd.read_csv('item_rating.csv')



def login():
    global gender

    gender = input('성별을 입력해주세요: ')
    new_userId = int(ratings.loc[len(ratings.index) - 1]['userId']) + 1
```

```
    print("userId: " + str(new_userId) + "\n성별: " +gender)
    rand_recommend(gender,new_userId)

def rand_recommend(gender,new_userId):
    global ratings
    r_num = int(input('추천 받을 횟수를 입력 하세요: '))
    for a in range(1,r_num+1):
        item_Id = random.randint(1,20)
        keywords = str("20대 " + gender + " " + item_keyword[(item_Id)-1] + " 추천")
        response = google_images_download.googleimagesdownload()
        arguments =
{"keywords":keywords,"limit":1,"output_directory":'image',"no_download":True,
"safe_search":True}
        paths = response.download(arguments)

        while True:
            itemscore = int(input('점수를 평가해주세요(1~5점): '))
            if 1 <= itemscore and itemscore <= 5:
                break
            else:
                print('warning: 1점에서 5점사이의 값을 입력하세요')
        if gender == '남자':
            new_gender = 1
        else: new_gender = 2
        new_itemId = item_Id
        new_rating = itemscore

        ratings.loc[len(ratings.index)] = [new_userId, new_gender, new_itemId, new_rating]
        ratings = ratings.set_index("userId")
        ratings.to_csv('item_rating.csv')
```

```
        ratings = pd.read_csv('item_rating.csv')
    svd_ratings(ratings,new_userId)


def svd_ratings(ratings,new_userId):
    reader = Reader()
    svd = SVD()

    data = Dataset.load_from_df(ratings[['userId', 'itemId', 'rating']], reader)
    trainset = data.build_full_trainset()

    svd.fit(trainset)
    lst_itemId = list(range(1,21))
    lst_ratings = list(ratings[ratings.userId == new_userId]['itemId'].values)
    notrating_itemId =list(set(lst_itemId)-set(lst_ratings))
    rand_itemId = random.sample(notrating_itemId,3)
    a = 0
    svd_rating = []

    for i in range(1,4):
        svd_rating.append(round((svd.predict(new_userId,rand_itemId[a]).est), 1))
        a += 1

    rating_dic = dict(zip(rand_itemId,svd_rating))
    recommend_itemId = max(rating_dic, key=rating_dic.get)
    final_recommend(recommend_itemId)


def final_recommend(recommend_itemId):

    keywords = str("20대 " + gender + " " + item_keyword[(recommend_itemId)-1] + " 추천")
```

```
    response = google_images_download.googleimagesdownload()
    arguments =
{"keywords":keywords,"limit":1,"output_directory":'image',"no_download":True,
"safe_search":True}


    print('사용자에게 추천하는 아이템입니다')
    paths = response.download(arguments)


    '''
    url = paths[keywords][0]
    resp = requests.get(url)
    img = Image.open(BytesIO(resp.content))


    '''



if __name__ == '__main__':
    login()
    print('프로그램을 종료합니다')
```

개발환경

-소프트웨어

윈도우10 ver.edu

파이썬 3.7.2

Anaconda navigator verPython3.7

Pycharm ver2018.3.5

-하드웨어

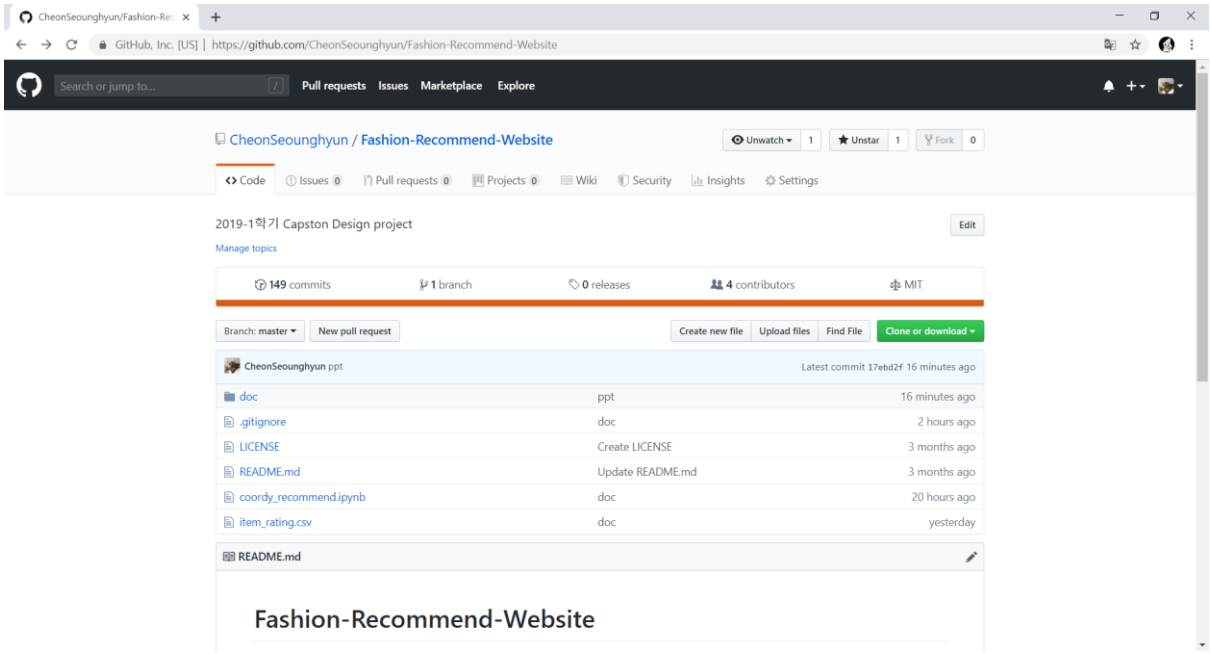
Cpu:i7-8700k

Gpu:Gtx1060 6G

Ram:8G

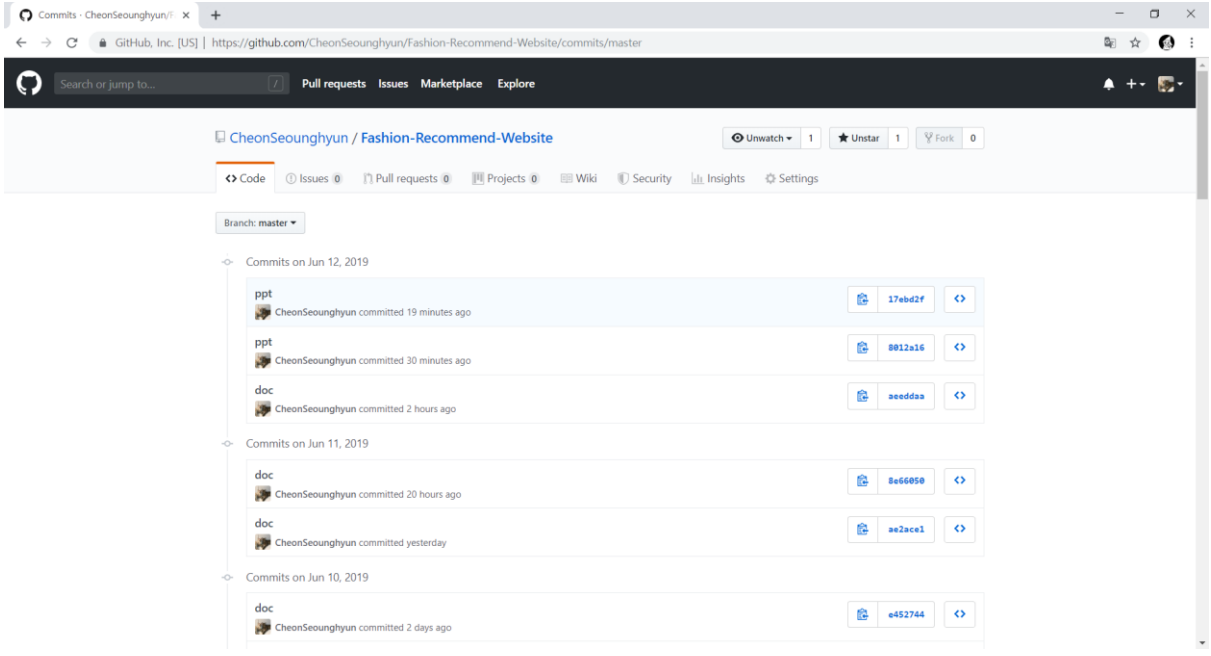
Storage:2TB

Github

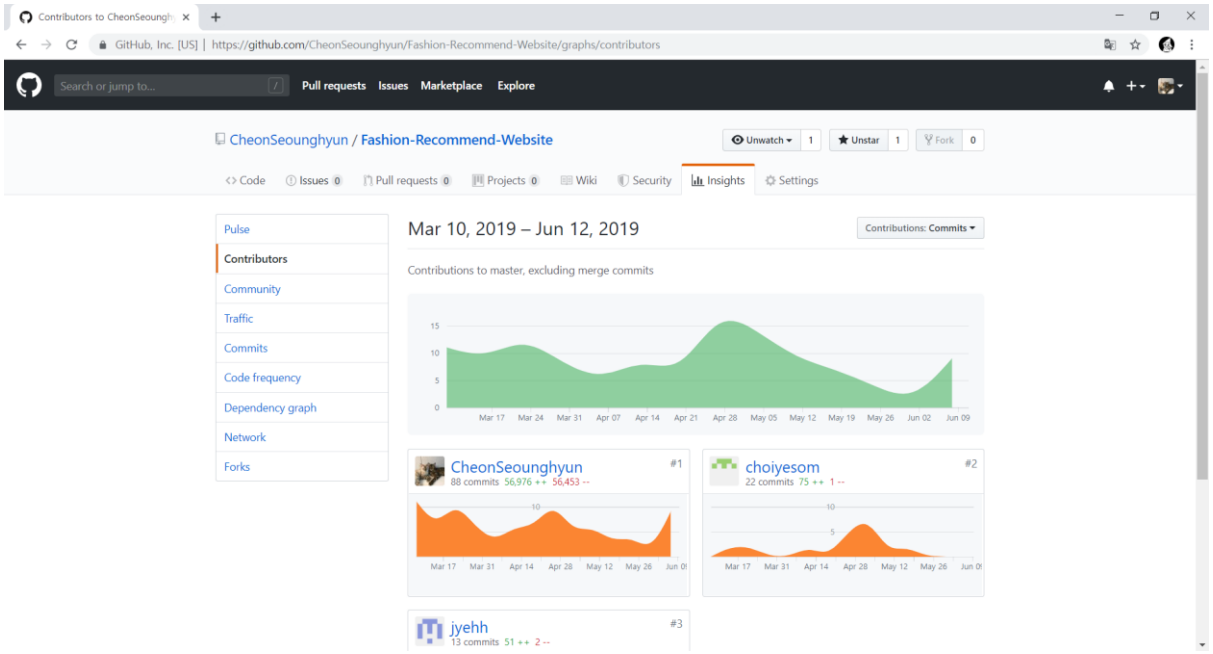


https://github.com/CheonSeounghyun/Fashion-Recommend-Website

프로젝트를 진행 시 협업방식은 GIthub를 사용하기로 하였다. Github 저장소 링크에서 본프로젝트에 사용된 문서파일, 프로그램코드, 사용 데이터 등을 다운로드 받을 수 있다.



https://github.com/CheonSeounghyun/Fashion-Recommend-Website/commits/master



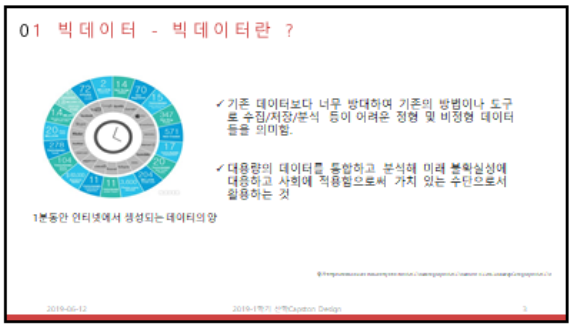
https://github.com/CheonSeounghyun/Fashion-Recommend-Website/graphs/contributors

또한 github 저장소 탭들을 이용하여 지금까지 저장소에 commit한 history, Traffic등을 확인 할 수 있다.

발표자료

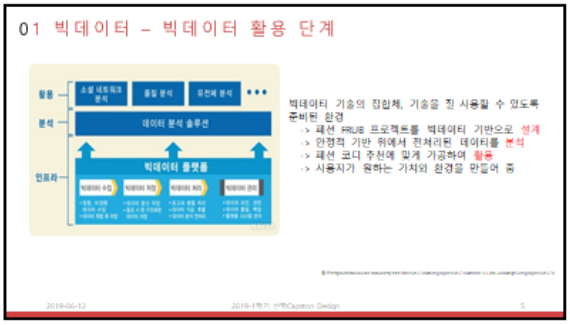
1

2



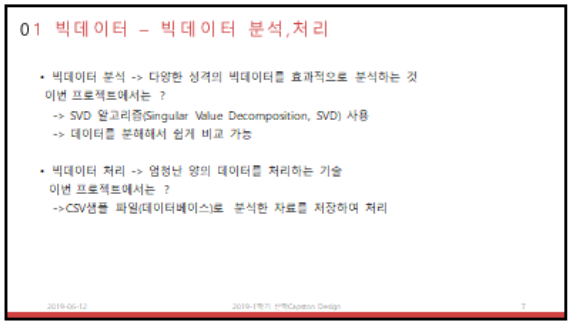
3

4

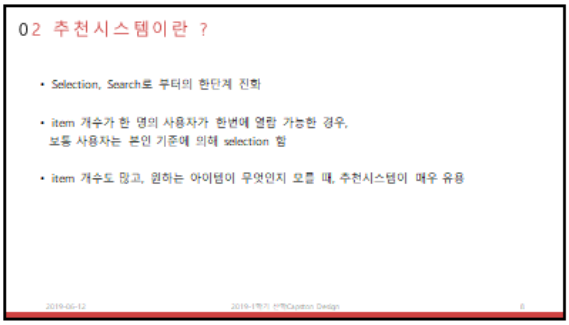


5

6



7



8

02 추천시스템 알고리즘 단계

Popularity (가장 단순)
- 가장 쉽게 인기도 즉, 높은 평점을 갖는 item을 추천 가능
- 모두에게 동일한 item이 추천 됨

Collaborative Filtering (중간 단계)
- user & item간의 평가를 이용하여, 사용자끼리의 유사도를 찾음
- 특정 user와 유사한 사용자들이 남긴 평점을 기반으로 해당 user의 평점을 예측
- 해당 user가 해당 item을 얼마나 좋아할 것인지 수치적으로 예측하는 것을 의미

Personalization (개인화)
- 모두에게 동일한 item이 아닌, 각 개인의 성향에 맞는 item을 추천
- user item 평가로 Collaborative Filtering Algorithm, SVD등이 사용 됨

2019-06-12 2019-1학기 산학Capston Design 9

9

02 협업필터링

✓ 협업필터링(collaborative filtering)?
많은 사용자들로부터 얻은 기호정보에 따라 사용자들의 관심사들을 자동적으로 예측하게 해주는 방법

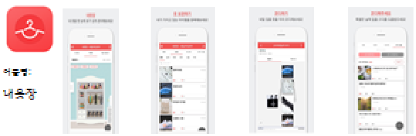
- Item based 아이템 기반
->사용자A가 선호하는 제품과 비슷한 제품을 찾아 추천해주는 방법
- User based 사용자 기반
->사용자 A와 사용자 B가 비슷한 성향을 가진 것으로 파악된다면 서로의 평가를 활용하는 방법

➡ FRUB프로젝트는 아이템 기반+사용자 기반

2019-06-12 2019-1학기 산학Capston Design 10

10

03 기존사례 비교 - 어플

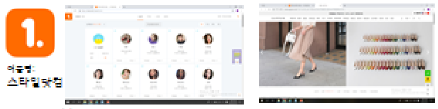


어플명:
내옷장

2019-06-12 2019-1학기 산학Capston Design 11

11

03 기존사례 비교 - 웹사이트



어플명:
스타일쉐어

2019-06-12 2019-1학기 산학Capston Design 12

12

03 기존사례 비교 – 문제점, 해결방안

- Cold Start Problem
  - 첫 사용자는 유사한 사용자가 없기 때문에 추천이 되질 않음
  ->프로그램 실행 시, 일정 개수 이상의 코디에 평점을 남기는 것을 유도
- User-Rating sparsity Problem
  - 대부분의 사용자는 평점을 잘 남기지 않는 경향이 있음
  ->이들에게는 개인화가 어려움, Popularity based 추천

2019-06-12 2019-1학기 산학Capston Design 13

13

04 프로젝트 소개 - FRUB

- FRUB
  ✓ Fashion Recommendation system Using Big Data
  -> 빅데이터를 활용한 패션 추천 시스템

  ✓ 화면에 표시되는 패션 사진을 보고 사용자들이 점수를 매기면 취향을 분석하여 연관되는 다른 사진을 추천해 줌

2019-06-12 2019-1학기 산학Capston Design 14

14

04 FRUB 프로젝트 소개

✓개발 동기
추천 시스템을 사용하는 웹사이트를 개발하며
빅데이터 분야를 접목시켜 보고 그 필요성을 느끼기 위함

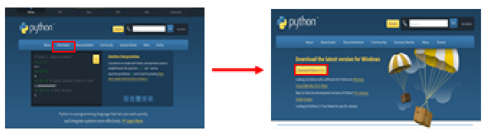
✓필요성
현대 경제시장에서 추천 시스템은 필수불가결한 요소이며
빅데이터 수집, 분석, 활용이 복합적으로 사용되는 시스템

2019-06-12 2019-1학기 산학Capston Design 15

15

04 프로젝트 소개 – FRUB 개발환경 (파이썬)

파이썬 -> 플랫폼 독립적 / 인터프리터식, 객체지향적, 동적 타이핑 대화형 언어
https://www.python.org/
데이터 분석에 특화된 언어이기때문에 사용,유용한 라이브러리들이 많음



2019-06-12 2019-1학기 산학Capston Design 16

16

04 프로젝트 소개 – FRUB 개발환경 (아나콘다)

17

04 프로젝트 소개 – FRUB 개발환경 (파이참)

PyCharm

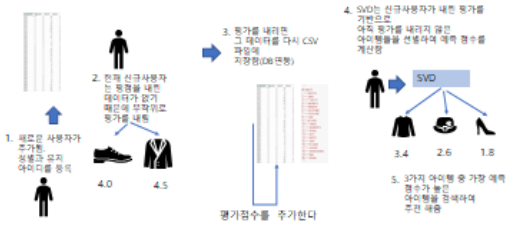
- https://www.jetbrains.com/pycharm/download/#section=windows
- 다운로드시 별도의 제약 없음
- 파이참 선택 이유 : Python 개발에 필요한 모든 도구를 제공

18

FRUB 프로젝트에 어떻게 적용할까 ?

19

04 FRUB 프로젝트 소개



20

04 FRUB 프로젝트 소개

아이템에 대한 사용자의 평가 점수 csv파일

아이템 목록

1. 새로운 신규 사용자에게 아이템을 무작위로 추천하여 평가 -> csv에 추가
2. 다른 아이템의 예측 점수 값을 SVD로 계산 -> 가장 높은 아이템을 추천

=> 저장->분석->처리->시각화의 단계로 새로 저장된 사용자의 데이터가 다시 평가에 사용되고 순환됨

21

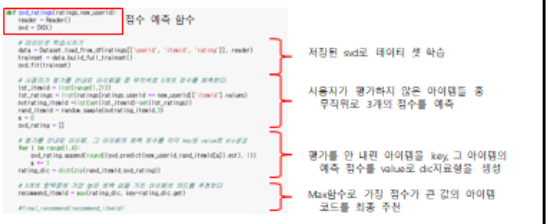
04 FRUB 프로젝트 소개 – 코드설명(로그인)

22

04 FRUB 프로젝트 소개 – 코드설명(추천)

23

04 FRUB 프로젝트 소개 – 코드설명(점수 예측)



24

04 FRUB 프로젝트 소개 – 코드설명(최종)

25

04 FRUB 프로젝트 실행 영상

26

04 FRUB 프로젝트 결론

- 추천시스템에서 빅데이터가 사용되는 원리를 이해했다
- 기존 추천시스템의 개선 방안을 모색했다
- 개선된 추천시스템 구현 했다

27

04 FRUB 프로젝트 소감

28

Q U E S T I O N
Q & A
A N S W E R

29

T H A N K Y O U
감 사 합 니 다

30

참조

참조도서

점프 투 파이썬

https://wikidocs.net/book/1

파이썬으로 데이터 주무르기

https://github.com/PinkWink/DataScience

참조 링크

https://github.com/hardikvasa/google-images-download

https://github.com/OlafenwaMoses/ImageAI

https://pypi.org/project/Google-Images-Search/

https://pypi.org/project/google_images_download/1.0.1/

파이썬 surprise

https://datascienceschool.net/view-notebook/fcd3550f11ac4537acec8d18136f2066/

https://www.fun-coding.org/recommend_basic7.html

https://towardsdatascience.com/how-to-build-a-simple-recommender-system-in-python-375093c3fb7d?gi=b35f7d6b5cdf

https://stackabuse.com/creating-a-simple-recommender-system-in-python-using-pandas/

https://www.datacamp.com/community/tutorials/recommender-systems-python

https://proinlab.com/archives/2103

https://sherry-data.tistory.com/44

https://mingkim.github.io/programming/2015/08/12/%EB%A6%AC%EC%97%91%ED%8A%B8%EB%A1%9C-%EA%B0%84%EB%8B%A8%ED%95%9C-%EC%9D%B4%EB%AF%B8%EC%A7%80-%EC%B6%94%EC%B2%9C-%EC%95%8C%EA%B3%A0%EB%A6%AC%EC%A6%98-%EA%B5%AC%ED%98%84%ED%95%98%EA%B8%B0.html

https://medium.com/daangn/%EC%9D%B4%EB%AF%B8%EC%A7%80-%ED%83%90%EC%A7%80%EA%B8%B0-%EC%89%BD%EA%B2%8C-%EA%B5%AC%ED%98%84%ED%95%98%EA%B8%B0-abd967638c8e

https://codevkr.tistory.com/36?category=705611

http://www.dator.co.kr/bmonthly/369960

https://terms.naver.com/entry.nhn?cid=58370&categoryId=58370&docId=3386304&expCategoryId=58370