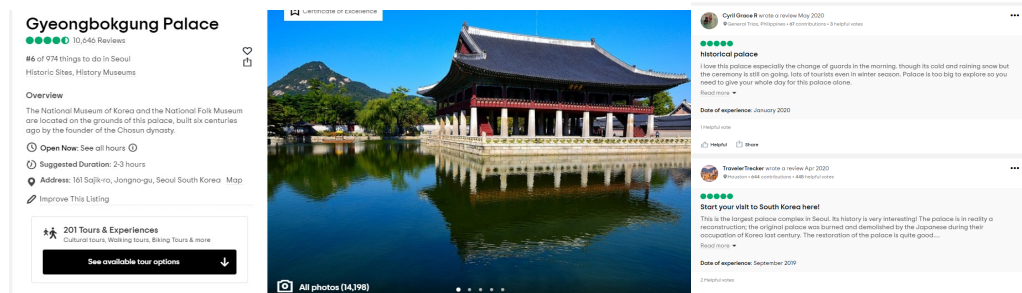


15주차 최종결과보고서

분류 1	창의융합형		
팀 명	글로벌비전	제출자	김응주
학습 일시	2020.06.28 00:00 ~ 2020.06.28 23:59	마감 일시	2020.06.28 23:59
제출 일시	2020.06.28 20:31	수정 일시	2020.06.28 20:32

참빛설계학기 최종결과보고서(학생주도형)

지도교수 확인란			
확인 기입 必 (인)			
팀명	글로벌비전		
대표연락처	010-4456-3191	대표 E-mail	dmdwn3979@naver.com
프로젝트명	영어권 여행자 후기 분석 시스템 TRAS		
총활동기간	3월 14일 ~ 6월 28일	신청학점	6학점
프로젝트 목표	영어권 여행자 후기를 이용한 여행지 추천 시스템		
	<p>1. 주제 선정</p> <p>코로나가 덮친 올해를 제외한, 지금까지의 한국 관광 통계는 지속적인 증가 추세를 보이고 있습니다. 전망이 밝은 관광업, 여행자들을 위한 시스템을 만들기 전에 저희는 소비자(여행자)의 입장에서 생각을 시작하였습니다. 여행을 계획 할 때 무엇을 참고할까? 여러 통계자료와 직접적인 설문(외국인 지인) 그 리고 개인적인 여행 경험을 통해 자국어와 영어로 이루어진 여행 사이트들을 주로 참고한다 라는 결론을 얻을 수 있었습니다. 이렇게 저희는 우리만의 "여행자들을 위한 시스템(사이트)"을 만들어 보자, 라는 점 에서 시작하게 되었습니다.</p> <p>2. TRAS 영어권 여행자 후기 분석 시스템</p> <p>여행객들을 위한 수많은 사이트(영어, 한국어, 중국어 등)는 이미 존재합니다. 저희는 다른 사이트와는 차별화 된 추천 활용 방안을 생각하던 중, 그 많은 사이트들이 갖고 있지만 활용이 덜 되었다고 생각 된 '후기'를 활용해 보자는 생각을 하게 되었습니다. 실제로 많은 후기들이 있지만, 그들의 별점 평균과 최근 후기 정도 보여지는 것이 현재 사이트들의 모습이었습니다. 이용자들의 needs를 반영하기에 댓글의 활용은 매우 효과적일 것이라 판단하였습니다. 지금까지의 관광 관련 연구는 주로 설문조사 방식으로 진행 되었고, 여행지 리뷰라는 비정형 데이터를 다룬 연구는 많지 않았습니다. 그리하여 텍스트 마이닝 기법 을 활용하여 직접적인 후기를 다루어 보고 의미를 도출하는 것이 저희의 목표가 되었습니다. 기존 한국 여행에 대한 연구들은 중국 위주로 이루어졌습니다. Dbpia, KISS 검색결과를 통해 중국어권은 관광의 주 목적이 73%가 쇼핑인 반면, 한국 관광공사에 따르면 (외래 관광객 실태조사) 관광의 주 목적으로서 역사와 문화 부분은 영어권이 43%를 차지하였습니다. 이처럼 한국 역사와 문화는 다양한 영어권 나라 들이 관심을 갖고 있으며 또한 세계 공용어이기에 리뷰 데이터가 많고, 기술도 잘 갖춰져 있어 영어 텍스 트를 분석하기로 결정했습니다.</p>		



(그림 1. TripAdvisor의 서울 경복궁 관련 정보와 댓글)

3. 프로젝트 관리, 파이썬 오픈 소스

팀원 모두가 코딩에 능숙하진 않았고 완전히 새로운 기술들을 접하기도 하였습니다. 따라서 저희는 프로젝트 관리를 위하여 github를 이용, 관리 하였고 필요한 기술(오픈소스)을 지속적으로 학습하였습니다. 관리된 github 페이지는 아래의 링크와 같습니다.

언어는 파이썬을 선택하였습니다. 파이썬은 통계 전용 언어인 R과 달리 크롤링부터 머신러닝, 사이트 제작까지 가능한 다목적 언어이며, 풍부한 오픈소스 생태계를 가지고 있습니다. 따라서 파이썬을 이용해 데이터 분석 전 과정을 쉽게 모듈화하고 시스템으로 통합 할 수 있었습니다.

<https://github.com/twinstae/tripReviewAnalysisSystem>

4. 기술 개발

크롤러

프로젝트의 시작은 크롤링으로부터 시작하였습니다. 우선적으로 여행지의 리뷰들을 수집하는 크롤러의 프로토타입으로 시작하였습니다.

개발 과정과 버그

기존 코드를 활용하여 응용하려다 보니 개발과정 속에서 수많은 버그가 발생하였습니다. 크롤러의 오류로 결측치가 생긴 attraction데이터가 있었고, 페이지 형 리뷰 (TripAdvisor에서 show more 버튼이 없는)를 크롤링하는 기능도 필요하였습니다. 이러한 과정들을 하나씩 해결해 나아가며 결국 직접적인 데이터가 없는(후기가 없는) attraction을 제외한 모든 정보는 크롤링 할 수 있게 되었고, 페이지 형 리뷰들도 pagination하며 크롤링 하였습니다. 이후 발생한 수많은 버그들도 모두 고쳤습니다. 코드가 복잡해지면서 디버깅을 하는데 원인을 찾기가 점점 어려워 졌습니다. 그래서 단순한 모듈로 쪼개어 정돈하고 유닛 테스트를 실시했습니다. 덕분에 이미지 크롤링 등 새로운 기능을 추가하고 유지보수가 편리해졌습니다.

크롤러의 응용

크롤러는 최종적으로 모듈형으로 개발되었습니다. 셀레니움과 뷰티풀 수프 라이브러리를 이용하여 180개의 어트랙션, 총 18000개의 리뷰를 수집하였고, 단순히 별점과 리뷰 내용만 읽어오는 것이 아닌, TripAdvisor의 자체적 분류 여행지 카테고리, 각 여행자의 여행 시기(계절), 여행자 구성 등 여러 정보를 수집하여 분석에 힘을 주었습니다. 결과적으로 이미지 크롤링이라는 새로운 과제에도 쉽게 변형, 재사용할 수 있었습니다. 현재 크롤러는 여행지별, 페이지 넘기기, 여행자 분류별, 목표 item 탐색, 예외처리, 리뷰 텍스트, 날짜, 주소, 이미지 크롤링 또한 가능합니다.

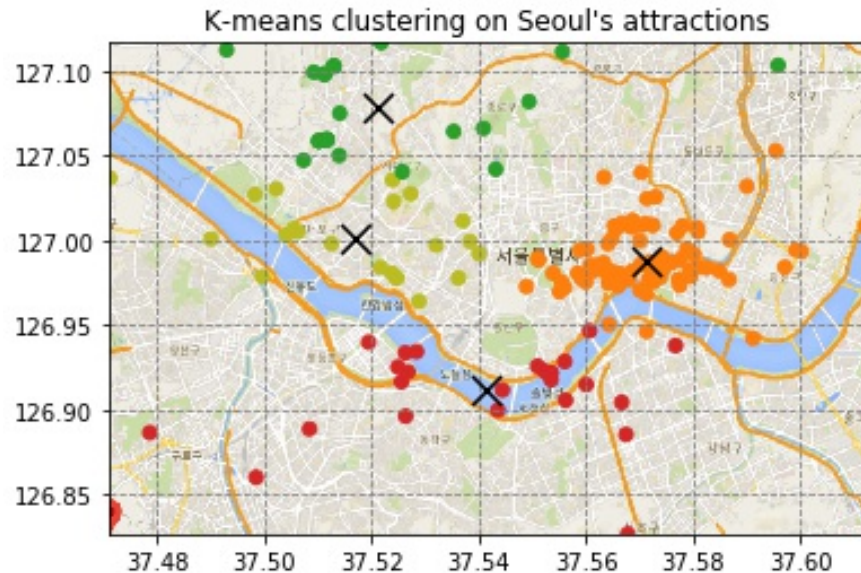
텍스트 마이닝과 머신러닝

전반적인 추천 시스템을 위한 모델 개발에 앞서 저희는 기본적인 TF-IDF와 로지스틱 회귀를 이용한 리뷰 벡터화 긍정/부정 감정 분석 프로토타입 모델을 만드는 것으로 시작하였습니다. 다소 적은 데이터로부터 시작하여(크롤링이 모두 되지 않았던 초기 상태) 매개변수들을 조작하며 유의미한 값을 추출하진 못했지만 이 후 모델들을 사용하는데 있어 초석이 되었습니다. 이후 A priori 분석을 통해 연관 규칙 분석도 해보았고 문장 단위 사전 기반 감정 분석, 여행지 별 긍정 부정 문장 추출, 여행 후기 감정 사전 구축 등을 진행하였습니다. 그 중 케라스 라이브러리 LSTM 인공 신경망을 활용하여 자체 감정 분석 모델을 구축하였습니다. 신경망은 2만개의 데이터를 통해 학습하였고 100개의 검증데이터를 이용하였습니다. 이렇게 나이브 베이즈 모델의 20% 정확도를 기점으로 케라스 LSTM 모델의 60% 정확도, 데이터의 전처리와 파라미터값의 직접적인 설정을 통해 85%의 정확도를 이끌어냈습니다.

여행지에 대한 정보 중 위치를 활용하여 거리 기반 추천을 만들었습니다. 하지만 단순히 16,110개의 위치의 쌍들을 길찾기 API로 크롤링 하는 것은 불가능했습니다. 따라서 kmeans를 활용하여 여행지에서

4개의 군집을 생성하였고 인근 여행지와 같은 군집 내 여행지 간의 쌍을 만들 고려하여 1천 쌍 가까이 경우의 수를 줄였습니다.

활동내용



(그림1. K-means를 활용한 attraction들의 군집화)

개발 모델: 앞선 모델들을 바탕으로 여러가지 추천 시스템을 개발하였습니다.

1. 거리 기반 추천: 해당 여행지와 가까운 여행지를 추천합니다. 군집화 된 데이터를 기반으로 여행지와 가까운(0.5km내외 혹은 1.5km 내외) 여행지를 추천합니다.
2. 별점 기반 추천: 후기 데이터의 별점 데이터를 활용하여 추천합니다. 표준 양식은 0~1값(1일수록 추천)으로 feature를 가공하여 추천 후보 rating에 곱해주었습니다.
3. 태그 기반 추천: ngram을 이용하여 주제어를 활용한 태그 추출 시스템을 만들었습니다. 'free entrance', 'night view' 등 각종 유용한 정보를 리뷰들을 통하여 추출, 이를 사용자가 보기 쉽도록 제시합니다. 또한 텍스트 유사도 모델을 구축하여 텍스트 요약 알고리즘을 사용, genism의 textrank로 리뷰 문항의 추출로 텍스트 유사도 기반 추천 시스템을 만들었습니다. 이는 비교적 추상적으로 나왔던 tag에 대하여 sort 기반으로 추천을 해보려했던 생각에이러 현재 텍스트 유사도 기반 추천과 함께 sort와 tag에 기반한 추천을 추가하게 되었습니다.
4. 텍스트 유사도 기반 추천: 새로운 여행지를 선택할 때 마다 유사도 딕셔너리를 업데이트 시켜 지금까지 선택한 여행지들과 비슷한 여행지에 가중치를 주는 텍스트 유사도 기반 추천을 만들었습니다.
5. 먼저 텍스트 요약 알고리즘인 genism textrank를 활용하여 모든 리뷰를 대표하는 표본을 추출하였습니다. 감정 분석 신경망을 이용하여 가장 긍정적인 리뷰와 부정적인 리뷰를 선정하여 보여주었습니다

5. 사이트 구축

AWS 아마존 웹 서비스 RDS 시스템을 이용하였고, MySQL에 데이터 베이스를 구축하였습니다. RDS 시스템은 자체 서버를 구축할 필요가 없어 간편하면서도 나중에 배포할 사이트와도 연동하기 쉽습니다. 동시에 보안 면에서도 개발자 외에 다른 사용자의 접근을 차단해주기 때문에, 안전합니다.

수집된 자료들은 모두 데이터베이스로 이관되었습니다. 이때 수집된 자료의 공유를 위하여 파이썬 웹 프레임워크 디장고에서 지원하는 ORM기능을 활용하였습니다. ORM은 각 데이터를 객체의 형태로 관리하고, 데이터베이스에는 구조화된 SQL을 이용해 맵핑하여 저장합니다. 이는 객체를 이용하여 일관적으로 데이터를 관리할 수 있으면서, 후에 데이터를 시각화하고 사이트에서 불러들이는 데에도 편리합니다.

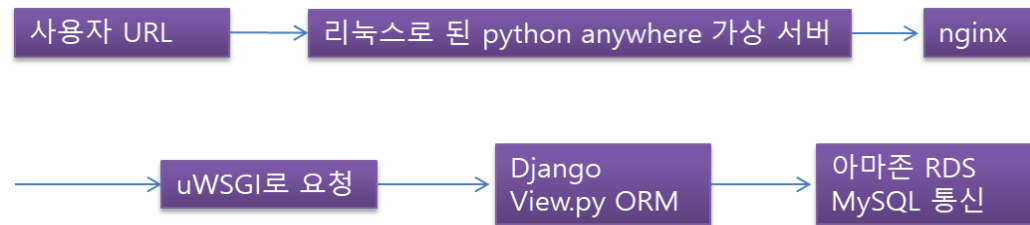
사이트 제작에 있어, 전체 페이지를 새로고침이 아닌 일부의 데이터를 서버에 요청하여 response하는 기법인 ajax를 사용하여 여행지의 지점(경로)을 마커에 담을 수 있도록 하였습니다. 이는 일부 여행사이트가 보여준 장점들을 벤치마킹하여 이용자들이 직접 편하게 여행지를 비교 한 후, 원하는 루트를 생성 및 비교 할 수 있도록 하는 기능의 기반이 됩니다.

데이터 시각화를 원활하게 진행하기 위하여 react를 사용하였습니다. 데이터의 동적인 표현을 연출하기 위해서 react는 적절한 효과를 보여주었습니다. 기존 jinja를 활용했을 때는 여행지에 대한 정보를 담고 있는 카드 180장이 모두 만들어진 상태로 대기해있던 반면, react를 활용하여 ajax로부터 데이터를 받아와 카드 생성이 될 수 있도록 하였습니다. 마찬가지로 데이터의 시각화에 있어 구글 지도 api를 활용하

였습니다. 자바스크립트 map을 활용하여 지도와 상호작용하는 동적인 애플리케이션을 생성하였습니다. 마커를 통해 카드를 보여주었고, 카드를 선택시 새로운 마커가 생성되었습니다. 이렇게 지도와 연동된 웹 어플리케이션을 만들었고, 사용자에게 필요한 시각효과를 줄 수 있습니다.

최종적으로 python anywhere라는 플랫폼을 이용하였습니다. 파이썬으로 이루어진 시스템에 대한 최적화가 잘 되어있어 유용하였습니다. 서버 프로그램은 NGINX를 이용하였고, uWSGI를 통해 Django와 연결하였습니다. 이때 보안을 유지하기 위하여 리눅스 서버에 환경 변수로 비밀번호를 모두 넣어, 코드에서는 환경 변수를 읽어 올 수 있게 하였습니다.

이러한 서버 적재는 사용자에게 웹 어플리케이션을 제공 할 수 있도록 합니다.



(그림1. 웹 어플리케이션을 제공하기 위한 구조)



(그림2. 데이터베이스와 서버, 프론트 엔드 시각화 사용 시스템)

6. 결과물 및 후기

크게는 3가지 과정(크롤러, 머신러닝, 웹사이트)을 통해 저희는 하나의 프론트엔드(웹사이트)로 표현을 하였습니다. 다소 비중이 앞쪽(백엔드)으로 쏠려 있어 아직 사용자에게 다가올 수 있는 부분은 미흡 할 수 있습니다. 하지만 저희의 목표와 맞게 TRAS 영어권 여행자 후기 분석 시스템은 사용자에게 제공할 단어 하나하나가 유의미한 지표를 가질 수 있도록 노력하였습니다. 실제로 화면상에서 보이는 모든 것들은 하나 이상의 분석이 들어갔고 단순히 사람의 손으로 선정된 것이 아닌, 여러 기술들을 활용하여 이끌어 낸 것입니다.

TRAS는 여행자에게만 정보를 제공하지 않습니다. 여행지 관리자, 사업자에게도 충분히 유의미한 지표를 제공해줍니다. 이는 당연하게도 "후기"데이터를 이용했기 때문입니다. 여행지 관리자들은 각 여행지의 어떠한 부분들이 장단점을 갖고 있는지 알 수 있고, 사업자들은 그 지역의 특색과 인지도를 파악할 수 있습니다.

이렇게 만들어진 분석 "시스템"은 여러 잠재력을 갖고 있습니다. 저희는 시스템 자체를 모듈화 시켰고, 이는 다른 지역 뿐만이 아닌 다른 나라, 다른 사이트로도 확장이 쉽게 가능합니다. 이는 시스템의 확장성(Scalability)를 고려한 것이며 일차원적인 의미가 아닌, 시스템이 더욱 커지는 것을 고려한, 추 후 다른 연구에서도 충분히 활용 될 만한 가치가 있다는 뜻 입니다. 추가적으로 크롤러의 경우, 유지보수성(Main tainability)을 고려하여, 기능 추가 부분을 최대한 유연하게 만들었고, 각종 버그들도 수정하여 모듈화 시켰습니다. 이렇게 저희는 프로젝트의 시작부터 끝까지 시스템의 확장성과 유지보수성에 대한 중요성을 느낄 수 있었습니다.

이 속에서도 많은 배운 점과 개선 점이 보였습니다. 우선적으로 협업의 관점에서 효율적인 역할 분배와 관리가 일어나지 않았습니다. 이로 인해 프로젝트의 진행에 차질이 있었으며 난관에 부딪히기도 하였습니다. 코드의 공유에 있어도 문제가 있었습니다. 정돈되지 않은 코드는 다음 단계에서 활용하는 팀원에게 어려움을 주었고 이는 곧 정돈되고 모듈화 시키며, 유닛 테스트를 활용하자는 방안으로 이루어졌습니다.

이렇게 단순히 프로젝트를 통한 학습과 결과물 뿐만이 아닌, 역할 분배와 협업관리의 중요성, 설계, 구현

, 테스트 등 체계적인 개발 프로세스의 중요성을 크게 느낄 수 있는 프로젝트이었습니다.

주차별 활동보고
(요약)

주차	주요 활동	활동방법	활동시간
1	탐색적 문헌 읽기 -관광 관련 기존 연구 확인	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
2	탐색적 문헌 읽기 - 관광 분야 공공데이터 수집	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
3	웹크롤링을 통한 초기 데이터 수집	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
4	탐색적 자료 분석	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
5	감성분석 프로토 타입 모델 시험 개발.	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
6	감성분석 프로토타입 검증	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
7	중간 보고서 작성	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
8	중간보고서 및 성찰일기 I (자기평가) 제출	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
9	크롤러 디버깅, LSTM 인공 신경망 감정 분석 모델 구축	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
10	여행지 추천 알고리즘 개발	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
11	토픽 모델링	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
12	프론트 엔드 개발	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
13	추천 시스템 개발	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
14	추천 시스템 향상 및 사이트 배포	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간
15	발표 준비 및 최종결과보고서 작성	화상 회의, 공동 작업, 대면 회의(주 1회)	총 12~20시간



그림1. 3주차 대면 회의

활동사진

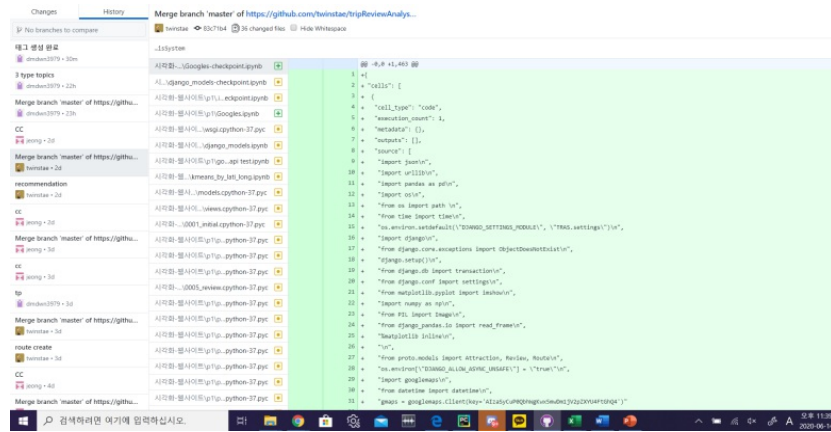
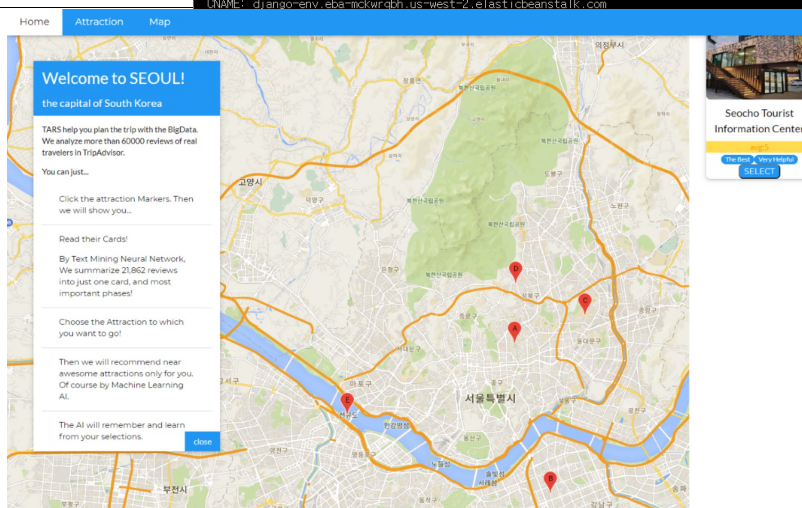
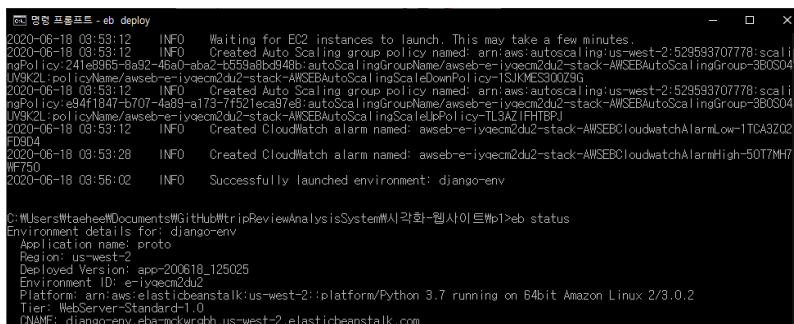
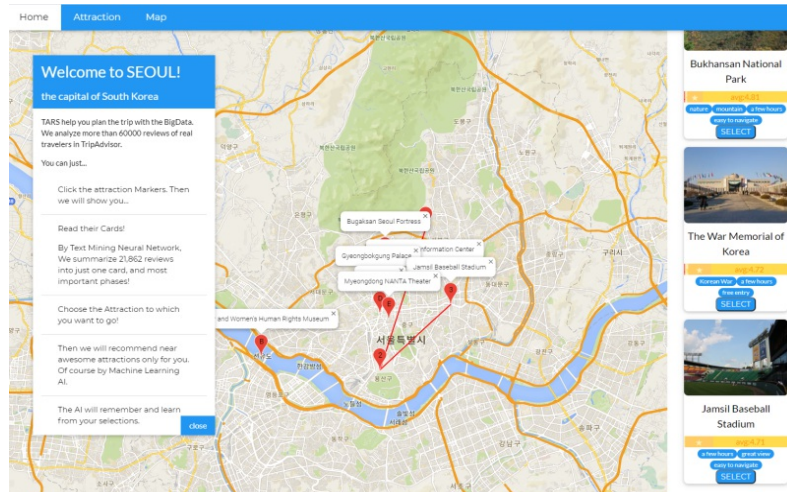
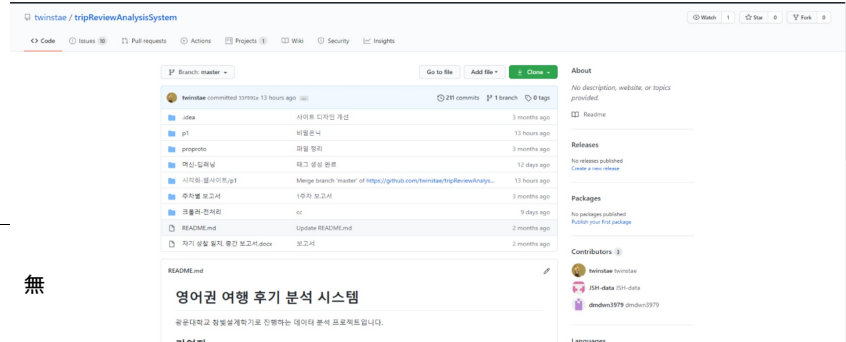


그림 2. 13주차 공동, 개인 작업(github)



최종결과물 및 성과물

사이트: <http://stelo.pythonanywhere.com/map/>

		
예산집행내역	無	
활동소감 및 건의사항	<p>프로젝트 활동을 통해 여러가지 배움을 느낄 수 있었습니다. 전공, 비 전공 적인 지식뿐만이 아닌, 팀 프로젝트 (협업)에서 필요한 자식을 얻을 수 있었습니다.</p> <p>현재 환경(도구나)으로 인해 생각제한점이 있었습니다. 주로 온라인 작업으로 이루어진 반면, 대면 공동 작업한 모든 코드는 깃허브 작업이 활발하게 이루어졌으면 하는 아쉬움이 남습니다.</p>	
지도교수 총평		

2020. 06. 26.

광운대학교 대학혁신사업단 귀하