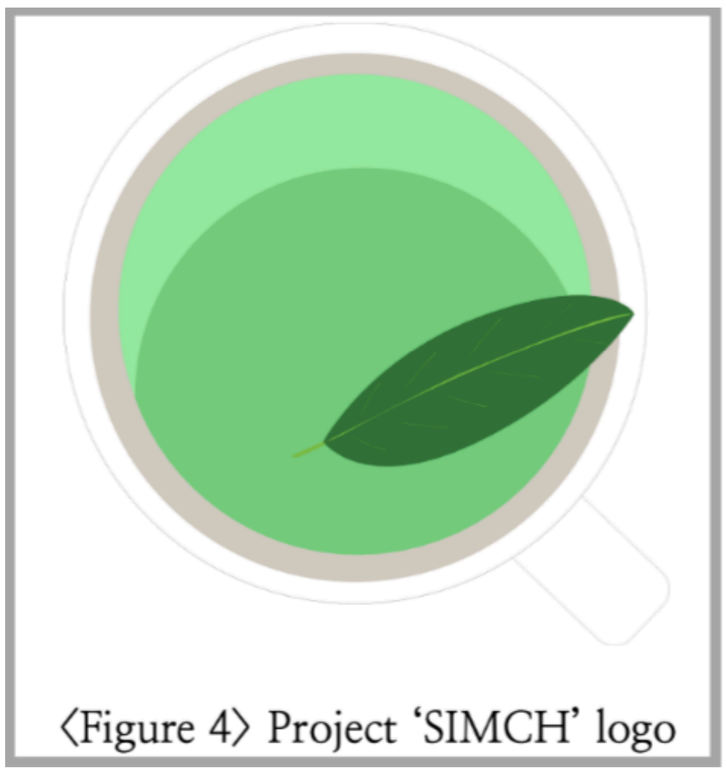
개인 사용자용 경량화 검색 엔진 - SIMCH

팀 명 이웅희 팀 원 이웅희 지도교수 이슬

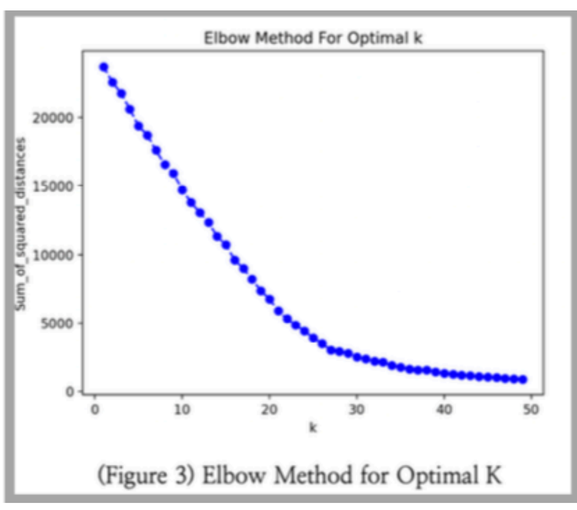
개발 동기 및 목적



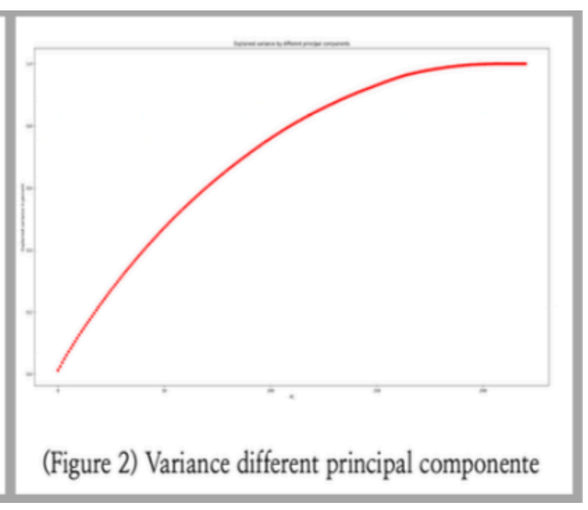
〈Figure 4〉 Project 'SIMCH' logo

본 프로젝트는 경량화된 개인 사용자용 검색 엔진 서버 구현을 목적으로 한다. 기존 유료로 제공되는 여러 검색 엔진을 대체하며 오픈 소스로 공개되었다.
검색 엔진 구현을 위하여 4가지 방법을 통해 접근하였고, 각 접근법을 비교 분석하여 더 나은 접근법을 찾는 것을 목적으로 하였다.
궁극적으로 최종 선택된 접근법을 바탕으로 보다 발전시켜, 경량화된 오픈 소스 검색 엔진이 많은 개인 사용자들에게 큰 효과를 보게 해줄 것을 기대하며 의의로 삼았다.

주요기술



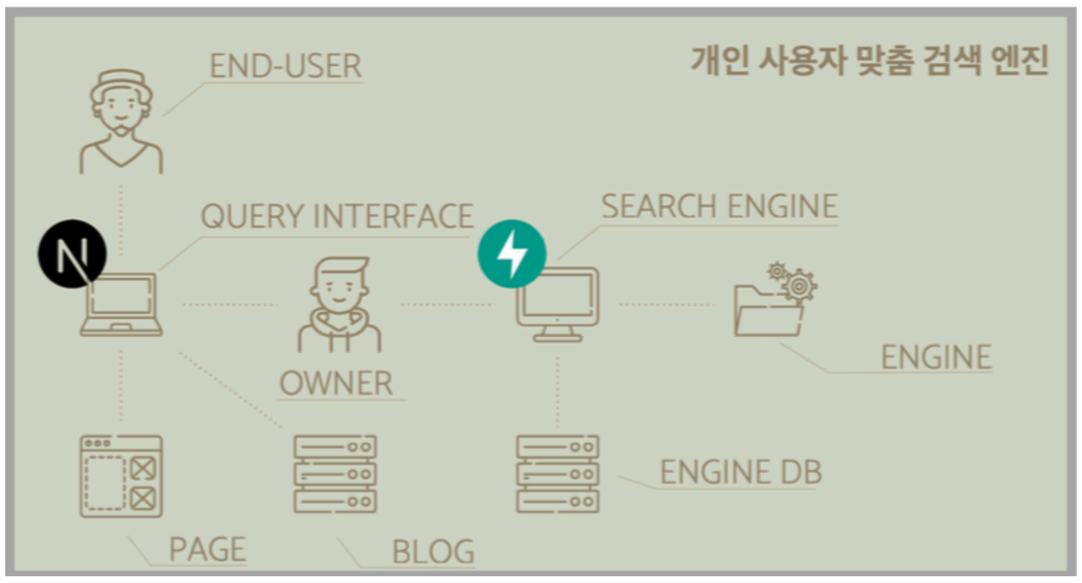
(Figure 3) Elbow Method for Optimal K



(Figure 2) Variance different principal componente

본 프로젝트는 'Naive', 'TF-IDF', 'Latent-Factor' 그리고 'K-Means'로 명명한 4가지 방법을 사용하였고, 각 방법의 핵심 알고리즘은 그 이름과 같다.
'Naive' 방법은 단순히 문서 내에서 입력받은 키워드 토큰들의 출현 횟수를 셈하여 점수를 정의했다.
'TF-IDF' 방법은 '사전 설정' 과정에서 저장된 TF-IDF 행렬을 이용한다. 각 쿼리 토큰에 해당하는 모든 값들을 더하여 해당 문서의 점수로 정의했다.
'Latent-Factor' 방법은 '사전 설정' 과정에서 저장된 LF 행렬을 이용한다. LF 행렬은 특이값 분해 후 전체 에너지 값 중 85%를 준수하는 선에서 근사값을 가지며 각 쿼리 토큰에 해당하는 모든 값들을 더하여 해당 문서의 점수로 정의한다.
'K-Means' 방법은 모든 문서와 쿼리 중 관련도가 높은 문서끼리 군집화되며 그 중 쿼리가 속한 군집 중 중점과 가장 가까운 문서를 반환 한다.

개발 내용



본 프로젝트는 Python 프레임워크인 'FastAPI'를 기반으로 구현하였다. 검색 엔진은 크게 'Crawl', 'Index', 'Search', 그리고 'Rank' 4가지 과정으로 나뉘며 각각 개별 클래스로 구현된다.
검색 엔진은 '사전 설정'과 '서버 구동'을 지원하며 사전 설정을 통해 검색에 필요한 데이터를 생성한 후 서버를 구동하여 사용자의 프로젝트 서버와 통신하며 데이터를 주고받는다. 서버는 오직 하나의 라우터만 제공하며 쿼리와 알고리즘 유형을 입력받는다.
프로젝트 과정에서 성능 측정 및 테스트를 위하여 'Next.JS'기반 테스트 블로그를 제작하였고 테스트 블로그에 들어간 데이터는 여러 공식 문서를 크롤링하여 Markdown 파일로 생성 후 저장하였다. 각 문서에 출처와 관련된 태그를 같이 저장하여 검색 엔진 성능 척도로 사용하였다.

결과 및 분석

	Naive	TF-IDF	Latent-Factor	K-Means
처리 속도 (ms)	13.2	11.4	9.5	5021.4
정확도	83 %	92 %	48 %	4 %

〈Table 1〉 30 iter, Mean of Results

본 프로젝트에서 사용된 4가지 접근법의 성능을 측정하기 위해 크롤링된 원 출처를 바탕으로 정한 태그 형식의 레이블을 사용하였다. 테스트 데이터는 총 9개의 사이트에서 220개의 포스트를 수집하였다.
처리 속도면에서 'Naive'와 'TF-IDF' 그리고 'Latent-Factor' 간에는 큰 차이가 없었고 'K-Means'의 경우 캐싱 데이터 활용이 불가했기 때문에 압도적으로 느린 결과를 보였다.
정확도 면에선 앞선 두 방법을 제외하곤 실사용에 무리가 있었으며, 특히 'K-Means'의 경우 복원률 약 30%정도에서 클러스터링을 진행하였기 때문에 아주 저조한 정확도를 보임을 알 수 있었다. 'Latent-Factor' 역시 낮은 정확도를 보이는데 많은 키워드에 비해 원본 행렬이 상당히 희소했기 때문이라 추측된다. 비록 85%의 에너지를 유지했음에도 불구하고 당초 데이터의 희소성을 극복할 수는 없었다.
가장 결과가 좋았던 'TF-IDF' 역시 단순 문자열 일치의 연장에 불과하단 한계가 있지만 전체 문서를 고려하여 키워드의 중요도에 차별을 둔 점에서 의의가 있다.