2021 Database System Project #3

Text mining with MongoDB

1 문제 정의

본 프로젝트에서는 텍스트 마이닝 기법 중 하나인 TF-IDF 가중치를 이용하여 제공된 트위터 데이터들을 분석하고 개별 트윗마다 핵심어를 추출함으로써 트윗들 사이의 유사도를 구할 수 있는 프로그램을 작성한다. 또한 트위터 분석을 위해 비정형 데이터를 다루기 쉬운 NoSQL 기반 데이터베이스인 MongoDB를 사용함으로써 NoSQL 데이터베이스의 사용법을 익힐 뿐 아니라 관계형 데이터베이스와 NoSQL 데이터베이스간의 차이점을 인식하는 것을 목적으로 한다.

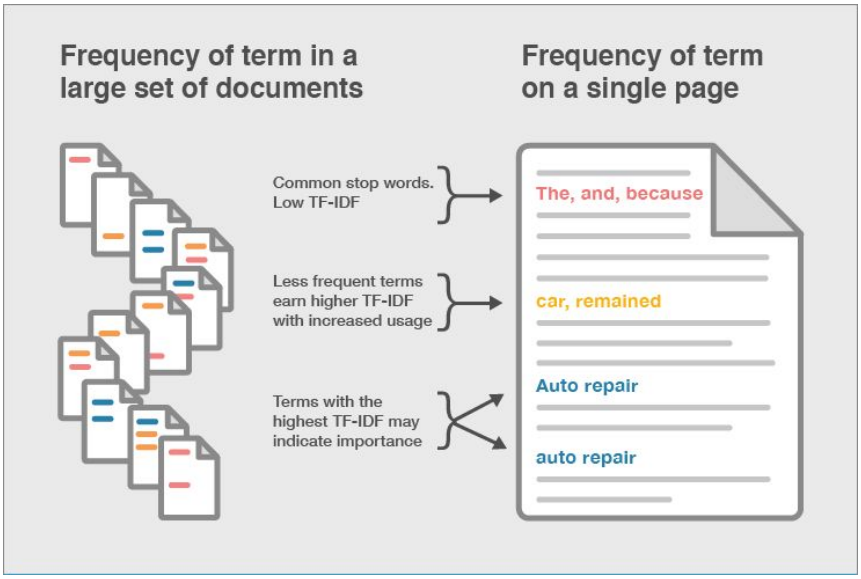
텍스트 마이닝이란 비정형 데이터 마이닝의 유형 중 하나로 자연어 처리 기술과 문서 처리 기술을 적용하여 유용한 정보를 추출, 가공하는 것을 목적으로 하는 기술이다. 실생활에서 만들어지는 대부분의 문서는 형태로 보관되며 제목, 저자, 출판날짜 등과 같은 구조적인 특징들과 문서의 요약, 내용과 같은 크기가 일정하지 않은 비구조적 요소들을 포함하기에 반구조적 데이터로 분류된다. 응용분야로는 Risk management, Knowledge management, Cybercrime prevention, Customer care service, Business intelligence, Spam filtering 등이 있다.

TF-IDF는 언어 자료 내의 특정 문서에서 어떤 단어의 중요도를 평가하기 위해 사용되는 통계적인 수치로 문서의 핵심어 추출, 검색 엔진의 검색 결과 순위 결정, 문서들 사이의 유사도 등에 적용될 수 있다. 일반적으로 한 문서에서 중요한 단어일수록 해당 문서 내에서 반복적으로 나타날 가능성이 높고, 그렇지 않은 단어의 등장 빈도는 낮을 것이다. 그러나 우리말의 조사나 부사와 같이 개별 문서 내에서 많이 등장하면서 또한 모든 문서 집합 내에서 반복적

으로 나타나는 단어의 경우 이런 단어들을 문서를 대표한다고 할 순 없다. TF-IDF 가중치는 이런 원리에 입각해 설정된 값으로 TF(단어 빈도, term frequency)는 특정한 단어가 문서 내에 얼마나 자주 등장하는지를 나타내고, 이 값이 높을수록 문서에서 중요하게 여겨진다. 하지만 단어 자체가 문서군 내에서 자주 사용되는 경우, 이것은 그 단어가 흔하게 등장한다는 것을 의미하고 이것을 DF(문서 빈도, document frequency)라고 한다. DF의 역수를 IDF(역문서 빈도, inverse document frequency)라고 하며 TF-IDF는 TF와 IDF를 곱한 값이다. 전체 문서집합을 D, 개별 문서를 $\mathrm{d} \in D$, 단어 t에 대해 문서 d에서 등장 횟수를 $n_t(d)$라 하면 TF-IDF 식은 아래와 같다.

$$\mathrm{tf(t,d)} = \frac{n_t(d)}{\sum_k n_k(d)}$$

$$idf(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

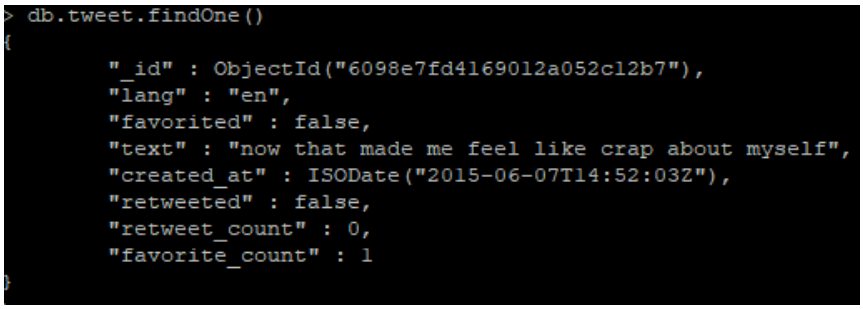


MongoDB는 C++로 작성된 오픈소스의 문서지향(Document-Oriented)적 데이터베이스이며 기존의 RDBMS의 한계를 극복하기 위해 만들어진 새로운 형태의 데이터저장소이다. 관계형 데이터베이스가 아니므로 RDBMS처럼 고정된 스키마 및 JOIN 연산이 존재하지 않으며, Document라고 불리는 기본 데이터 구조 단위로 이루어진다. 모든 데이터 구조는 한 개 이상의 key-value 쌍으로 이루어진다.

```
{
    "_id": ObjectId("5099803df3f4948bd2f98391"),
    "username": "ironman",
    "name": { first: "Tony", last: "Stark" }
}
```

2 요구사항

본 프로젝트에서는 제공된 트위터 데이터에 대해 형태소 분석을 통해 문서를 키워드의 집합으로 분할한 후 TF-IDF 작업을 수행한 후 기사들 간의 유사도를 구하는 것을 목적으로 한다. 다음은 MongoDB에 저장된 트위터 데이터에 대한 예제 화면이다. 아래 예제 화면을 참고하여 문서의 구조를 분석하고, 2.1 ~ 2.5까지의 작업을 진행한다.



2.1. MongoDB 질의(20점)

다음의 질의를 작성하고 **질의문과 결과를** 보고서에 기입하시오.

1) 전체 트윗의 개수를 구하시오.

2) retweet된 트윗의 개수를 구하시오.

3) 좋아요(favorite_count)가 2 이상인 트윗의 개수를 구하시오.

4) 전체 트윗을 날짜 별로 정렬해서 가장 빠른 날짜의 트윗을 출력하시오. (ObjectID와 날짜만 출력할 것.)

5) 2015년 6월 1일부터 2015년 6월 30일까지 6월 한 달 동안 실린 기사의 개수를 구하시오. (6월 30일에 올라온 트윗도 포함.)

2.2. 형태소 분석 및 불용어 처리(10점)

형태소 분석은 텍스트 마이닝 시 자연어 처리의 첫번째 단계로 입력문자열을 형태소열로 바꾸는 작업을 말한다. 형태소란 의미의 최소단위로써 더 이상 나눌 수 없는 가장 작은 단위의 의미 요소를 말한다. 모든 트윗에 대해 형태소열을 데이터베이스에 저장하라. 또한 사용자로부터 트윗의 Object id를 입력 받아 해당하는 트윗의 형태소들을 출력해주는 프로그램을 작성하고, 출력화면을 보고서에 기입할 것.

2.3. Word Count 구하기(15점)

Word Count는 TF-IDF 작업을 수행하기 전 사전 단계로 각 문서들마다 포함되어 있는 고유한 개별 단어들의 빈도수를 측정한다. 형태소 분석 단계에서와 마찬가지로 모든 트윗에 대해 고유한 단어들의 등장횟수를 측정하고, 그 값을 데이터베이스에 저장하라. 사용자로부터 트윗의 Object id를 입력 받아 해당하는 트윗의 WordCount 리스트를 출력해주는 프로그램을 작성하고, 출력화면을 보고서에 기입할 것.

2.4. TF-IDF 수행(15점)

본격적으로 텍스트 마이닝을 수행하기 위해 2.3에서 저장한 WordCount 값을 이용해 단어들마다 문제 정의 부분에서 설명한 TF-IDF 값을 계산한다. 계산한 TF-IDF 가중치를 WordCount를 저장한 방식과 마찬가지로 데이터베이스에 저장한다. 트윗의 Object ID를 입력받고 데이터베이스로부터 관련된 문서들을 찾은 후 TF-IDF 가중치를 출력하는 프로그램을 작성하고, 출력화면을 보고서에 기입할 것. (출력은 TF-IDF 값이 가장 높은 단어 순서대로 상위 10개의 단어와 TF-IDF 값을 출력하라.)

2.5. 문서 유사도 구하기(20점)

두 트윗의 Object ID를 입력 받고 데이터베이스로부터 관련된 문서들을 찾은 후 TF-IDF 가중치를 이용해 두 문서의 유사도를 구하는 프로그램을 작성하라. 두 문서간의 유사도는 주로 Cosine 유사도를 이용해 구해지는데, 문서 A,B가 주어졌을 때 문서를 단어 벡터형태로 표현하고 벡터 성분을 tf-idf값으로 할당한 뒤 유사도를 다음과 같은 식에 의해 구한다.

$$\text{Cosine Similarity(A, B)} = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2}\sqrt{\sum_{i=1}^{n} (B_i)^2}}$$

주의: 위 식의 $A_i B_i$는 두 문서간의 공통된 단어의 TF-IDF 값을 곱한 것이다.

예를 들어, {jazz(0.7), classic(0.1), music(0.5)}라는 단어(괄호 안은 tf-idf 가중치)로 이루어진 문서 A와 {music(0.7), pop(0.3)}라는 단어로 이루어진 문서 B가 있다면, 전체 문서의 총 단어 집합은 {jazz, classic, music, pop}이 된다. 이에 따라 문서 A를 단어 벡터로 표현하면 {0.7,0.1,0.5,0.0}이 되고, 문서 B를 단어 벡터로 표현할 경우 {0.0,0.0,0.7,0.3}이 된다. 이 때 두 문서의 유사도는 위의 코사인 유사도 식에 의해 아래와 같이 계산된다.

$$\text{유사도} = \frac{(\mathbf{0.7 * 0.0 + 0.1 * 0.0 + 0.5 * 0.7 + 0.0 * 0.3})}{(\mathbf{0.7^2 + 0.1^2 + 0.5^2 + 0.0^2})(\mathbf{0.0^2 + 0.0^2 + 0.7^2 + 0.3^2})} = \frac{\mathbf{0.35}}{\mathbf{0.435}} \cong \mathbf{0.8}$$

2.6 프로그램 메뉴 구성도(필수)

처음 프로그램을 실행하면 아래와 같이 전체 메뉴 구성도를 구성한다. 그 후 사용자로부터 세부 프로그램 번호를 입력 받아 해당하는 프로그램을 수행하도록 한다.

1. WordCount
2. TF-IDF
3. 문서 유사도

3 사용환경

서버: Host - dblab.sogang.ac.kr / Port - 22

운영체제: Ubuntu 14.04.5 LTS

데이터베이스: Mongodb 3.2.20

사용언어: python 2.7.10

라이브러리: pymongo

서버계정: db학번 (e.g. db20211234)

서버비번: 1234 (e.g. 1234)

데이터베이스 계정: db학번 (e.g. db20211234)

데이터베이스 비번: 1234 (e.g. 1234)

4 제출물

4.1 기술 문서(보고서)(40점)

4.1.1 MongoDB 질의문 및 결과(20점)

데이터베이스 서버에 접속해 MongoDB 질의를 수행하고 결과 화면을 screenshot으로 첨부할 것.

4.1.2 RDB vs. NoSQL DB에 대한 비교(20점)

텍스트 마이닝을 수행할 때에는 RDB보다 NoSQL DB를 사용하는 것이 좋은가? 이에 대한 답을 하고 이유를 기술하시오(진행된 프로젝트와 관련 지어 기술).

4.2 Python 프로그램 파일(60점)

작성한 프로그램 소스 파일

5 제출방법

5.1 Hard Copy

기술 문서를 1부 출력하여 제출

5.2 Soft Copy

기술 문서와 Python 코드 파일을 압축하여 사이버캠퍼스로 제출

파일 명 등의 양식은 다음을 따를 것.

기술 문서 파일 : DBprj#3_학번.docx (e.g. DBprj#3_20211234.docx)

Python 파일 : DBprj#3_학번.py (e.g. DBprj#3_20211234.py)

압출 파일 : DBprj#3_학번.zip (주의 : zip 이외의 다른 압축형식은 받지 않음)

6 제출 기한

06월 07일(월) 오후 11:59시 전까지 사이버 캠퍼스 제출

7 평가 기준

- 요구 사항들이 적절히 반영 되었는가
- 제출물이 정해진 기한 내 제출 되었는가

8 기타

- **데이터베이스 접속 계정 및 방법은 추후 공지하고 관련 내용을 실습할 예정**
  실습일: 05/26(수) 질의응답: 06/02(수)
- 입력 형식과 출력 형식은 자율에 맡기되 사용자 편의성을 고려해 구성할 것.
- Copy는 1회 적발 시 0점 처리, 2회 적발 시 과목 성적 F 처리
- 다음과 같은 경우 감점
  - 기한을 지키지 않은 경우.(하루당 10%씩 감점)
  - 첨부 파일의 압축이 손상되거나 바이러스가 있는 경우 0점 처리
  - 제출 양식을 지키지 않은 경우 제출물을 찾지 못하면 미 제출 처리 될 수 있음. 그 이외의 경우 10%씩 감점