

검색엔진 작동 원리부터 응용까지

만들며 배우는 검색엔진 원리와 응용

김남준

오늘 할 내용

- 검색엔진 (Search Engine) 작동 원리
- 응용 (한글 초성 검색)
- 과제

사전 요구 사항

- Python을 사용 해 보았다.

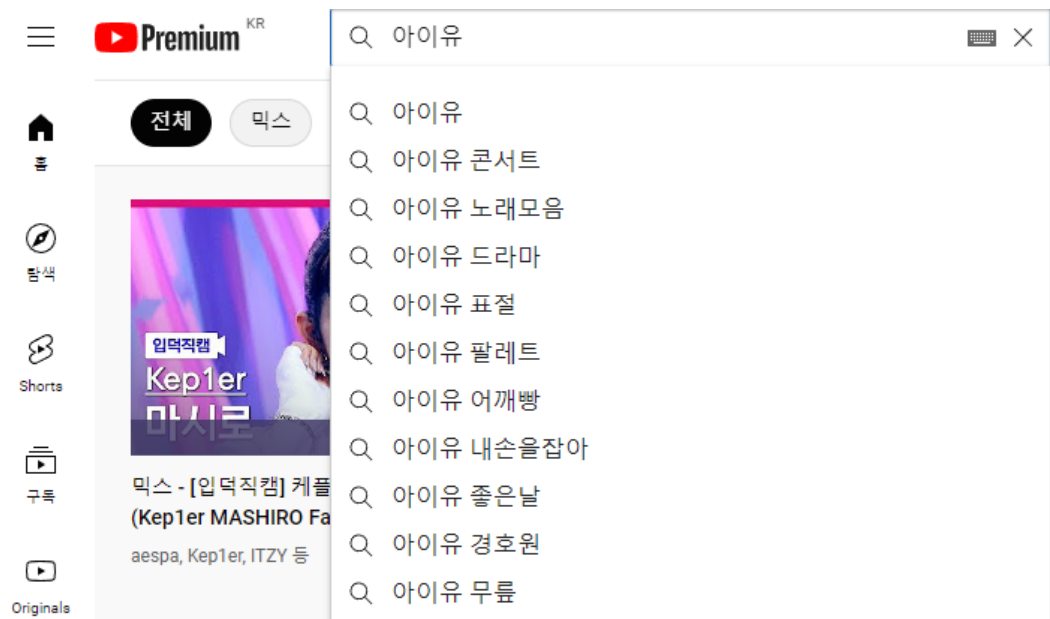
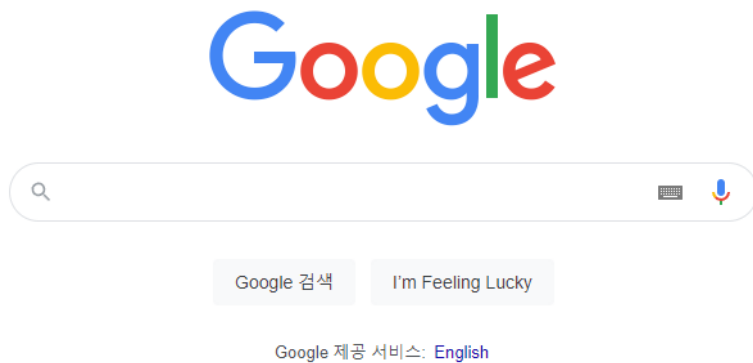
시작 전

- 검색엔진은 사드세요..



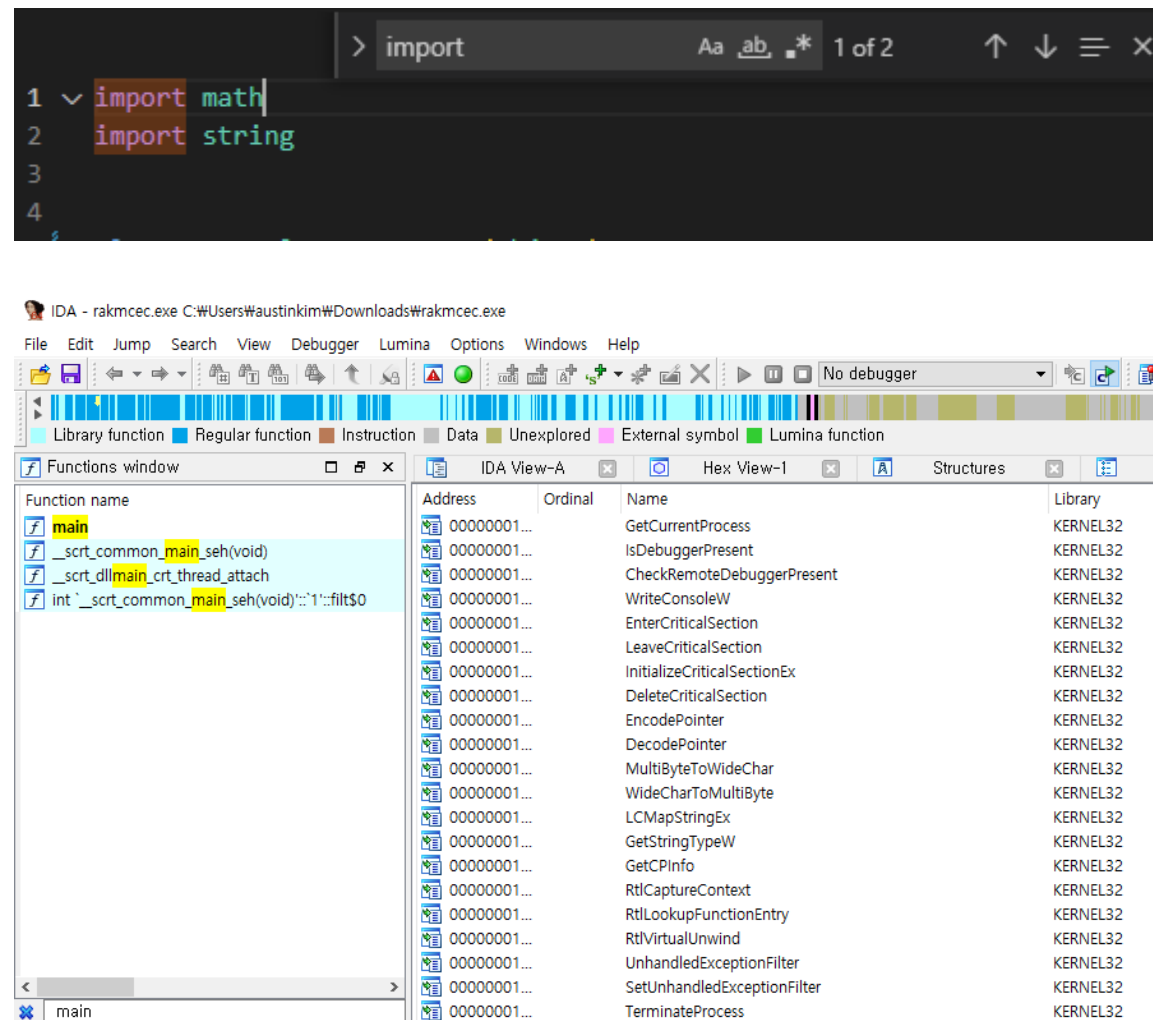
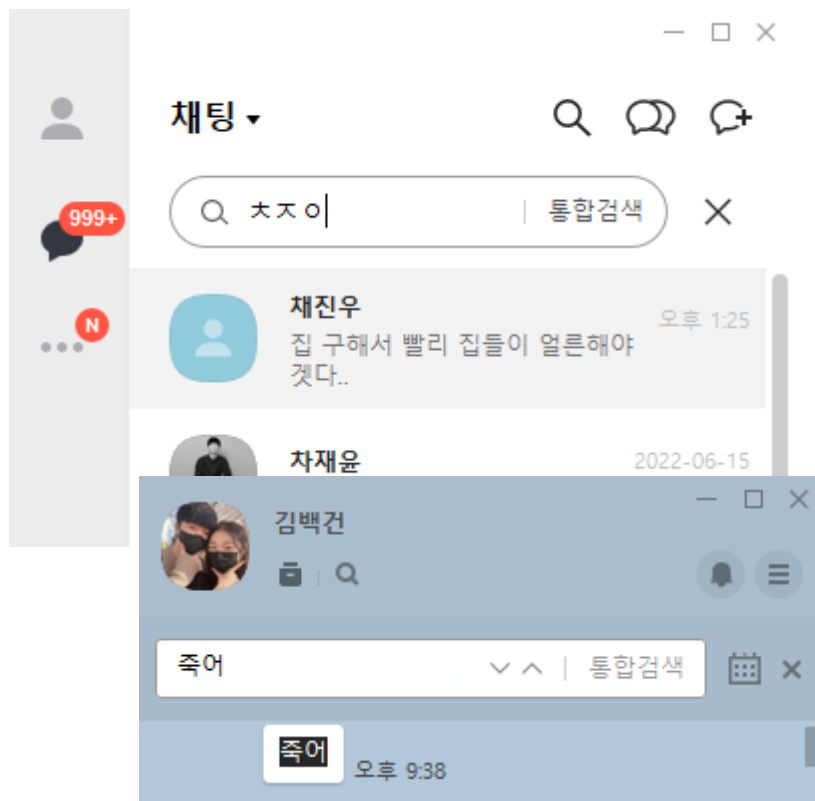
검색 엔진?

- 우리는 매일 검색 엔진을 사용합니다



검색 엔진?

- 진짜 매일 사용합니다



왜 검색엔진을 사용해야 하는가?

- 검색 없이 내가 원하는 정보를 찾기 **매우** 어려움



처음에는

- 정보를 수집한 뒤 정리해서 저장 해 두기 시작 (데이터의 모음)
- 이걸 우린 **데이터베이스(Database)**라고 부르기로 했어요.

SQL Query

```
SELECT userid, name, address  
FROM service_users  
WHERE userid = 'austinkim';
```



SQL Query Result

userid	name	address
austinkim	Namjun Kim	605 Skiff Street, North Haven, CT 06473

근데

- 요구사항이 점점 많아지면서 검색에 한계가 발생합니다.

예를 들자면?

- "경기도 안양시" 와 "경기도 광주시"에 사는 사람을 검색하고 싶어요. (LIKE 문 해결)
만약 중국에서 이 서비스를 한다면? (중국 인구 14.02억, 2020년 기준)
- 사람 이름을 초성으로만 검색할 수 있으면 좋겠습니다.
- 주소를 입력하면 그 주소와 가장 비슷한 곳에 사는 사람을 찾고 싶어요 (랭킹)
- 기타등등...

그래서 오늘 배워볼 것은

- 이러한 문제를 해결하기 위해 사람들이 만들어 둔 좋은 기술들을 배워보는 것.
- 그리고 그 기술을 내 손으로 직접 만들어 보는 과정
- 다시 한번 말하지만, 실제로 검색 엔진을 써야 한다면 제발 사드세요.

검색 엔진의 3요소

- 수집 (Crawling)
- 색인 (Indexing)
- 질의 (Searching)

수집

- Crawling (크롤링)
- 색인하고 질의할 정보를 가져오는 과정.
- 우리가 크롤러(ex. 웹 크롤러)라고 부르는 것들이 여기에 속함.
- 외부 서비스 API를 사용하거나, 자체 정보를 통해 수집할 수 있음.

색인

- Indexing (인덱싱)
- 검색을 더욱 빠르게 하기 위해서 데이터의 특성이나 정보를 저장하는 행위

예시

- 우리는 이미 인덱싱을 실생활에서 많이 보고 있습니다.
- 책이나 논문, 단어집 등등



질의

- Searching (or Querying)
- 사용자가 질의(Text, SQL)을 통해 원하는 정보를 얻는 행위
- 질의 결과를 사용자에게 잘 보여주기 위해 스코어링(랭킹 시스템 등)을 사용하기도 함

질의

스코어링 / 랭킹

- 내 질의 내용이랑 제일 유사한 순으로 정렬
- 내가 제일 원하는 정보를 가장 위로

VIEW

• 전체 • 블로그 • 카페

Travel Anywhere in the world/장승승의 코메디... | 인플루언서 | 2021.05.19.
[캐나다일상] 코로나백신 화이자 접종 후기
코로나 백신 접종 후기 PFIZER 우선 밴쿠버는 코로나 핫스팟 중심 지역(응 우리 동네 그중 하나)들을 중심으로 전 연령을 맞추고 있다. 오타와를 비롯한 다른 주들...

#코로나백신접종 #코로나백신화이자 #코로나백신접종후기



파란여행 | 캐나다 현지 한인 여행사 | 2021.09.28.
캐나다 여행방법&입국서류 총정리! (ArriveCAN/PCR음성확인서/...
" #캐나다입국서류 #arrivecan #한국에서캐나다 #백신접종완료 #pcr검사 #pcr음성확인서 #영문증명서 #캐나다eta #입국과정 #캐나다여행 #토론토 #공항코로나테...

#캐나다입국서류 #arrivecan #한국에서캐나다 #백신접종완료 #pcr검사

캐나다/한국/미국 입국시 확인사항&준비를 한눈에보기! (pcr음성확인서/백신...
자가격리 없이 캐나다에서 한국 입국하기(백신접종/격리면제서/방문비자/pcr...



김치군의 내 여행은 여전히 ~ing | 2022.07.26.
캐나다 여행 입국 서류 - 영문 백신접종증명서, eTA, ArriveCAN, ...
캐나다 여행 입국 서류 - 영문 백신접종증명서, eTA, ArriveCAN, 유아&아동 코로나 규정과 검사 캐나다 여행도 이제 자유롭게 할 수 있게 되었고, 더이상 출국 전에 P...

#캐나다 #캐나다여행 #입국 #캐나다입국 #캐나다입국서류



공부하는 워킹맘의 고군분투기 | 2022.04.08.
캐나다 입국 코로나 영문진단서, 백신접종 완료자는?
캐나다, 백신 접종 완료자 입국 전 코로나 검사 폐지. 뉴스가 눈에 들어왔다. 날짜를 확인하니 4월 2일. https://biz.sbs.co.kr/article/20000056865 자가격리는 물론 신속...

#캐나다이국규문나영무지다서 #캐나다하당삭기 #해이축국규문나지다서



만들기 전에 다시 한번 봅시다

검색 엔진의 3요소는

- 수집 (Crawling)
- 색인 (Indexing)
- 질의 (Searching)

오늘 구현해 볼 내용은

- ~~• 수집 (Crawling)~~
- 색인 (Indexing) + 전처리 (Pre-processing)
- 질의 (Searching)

전처리

색인 전에 전처리(Pre-processing) 과정이 필요합니다

- Cleaning + Normalization (정제 + 정규화)
- Tokenization (토큰화)
- Removing Stopwords (불용어 제거)
- Stemming (어간 추출)
- Lemmatization (표제어 추출)

예제

이해하기 쉽도록 예시 문장을 가져와서 변환 해 보도록 하겠습니다. (영문)

Bitcoin transactions are verified by network nodes through cryptography and recorded in a public distributed ledger called a blockchain.

Cleaning and Normalization

- 토큰화(Tokenization) 하기 전에 데이터를 정제하고 정규화 하는 작업.

정제(Cleaning)

- 대소문자 통합, 특수문자 등 노이즈 데이터 제거, 짧은 길이의 단어 제거($n=3$)

원문) Bitcoin transactions are verified by network nodes through cryptography and recorded in a public distributed ledger called a blockchain.

정제 후) bitcoin transactions verified network nodes through cryptography recorded public distributed ledger called blockchain

Cleaning and Normalization

정규화(Normalization)

- 표현 방법이 다른 단어를 하나로 통합시켜 동일하게 만드는 과정.

원문) bitcoin transactions verified network nodes through cryptography recorded public distributed ledger called blockchain

정제 후) bitcoin **transaction** **verify** network **node** through cryptography **record** public **distribute** ledger **call** blockchain

Tokenizing

- 코퍼스(말뭉치, Corpus)에서 Token으로 단위를 바꾸는 과정.
- 의미가 있는 단위로 토큰화 작업을 진행
- 예시) 띄어쓰기를 기준으로 토큰화를 진행

원본) bitcoin transaction verify network node through cryptography record public distribute ledger call
blockchain

토큰화 후) List["bitcoin", "transaction", "verify", "network", "node", "through", "cryptography", "record", "public",
"distribute", "ledger", "call", "blockchain"]

Removing Stopwords

- 분석 시 의미가 없는 단어들을 제거하는 과정 (불용어, 사용하지 않는 단어)
- 여기서는 주로 불용어 사전을 이용하여 제거하는 방식

```
>>> from nltk.corpus import stopwords
>>> import nltk
>>>
>>> nltk.download('stopwords')
[nltk_data] Downloading package stopwords to
[nltk_data] C:\Users\ austinkim\AppData\Roaming\nltk_data...
[nltk_data] Unzipping corpora\stopwords.zip.
True
>>> db = stopwords.words('english')
>>> len(db)
179
>>> tokens = ["bitcoin", "transaction", "verify", "network", "node", "through", "cryptography", "record", "public", "dis
tribute", "ledger", "call", "blockchain"]
>>> tokens = [v for v in tokens if v not in db]
>>> tokens
['bitcoin', 'transaction', 'verify', 'network', 'node', 'cryptography', 'record', 'public', 'distribute', 'ledger', 'cal
l', 'blockchain']
```


Stemming

- 단어에서 어간(단어의 핵심 부분)을 추출하는 작업 (affix를 제거)
- ex) handles, handle, handler, handling -> handl
- Porter Stemming Algorithm: 5개의 단계로 접미사를 제거하는 알고리즘

Rule	Example
SSES → SS	caresses → caress
IES → I	ponies → poni
SS → SS	caress → caress
S →	cats → cat

Lemmatization

- 단어에서 표제어(사전에서 검색 가능한 단어)를 추출하는 과정
- ex) handles, handle, handler, handling -> handle

Stemming 와 Lemmatization의 차이?

- 둘 다 Corpus 내 단어의 수를 줄이는데 사용

Stemming vs Lemmatization

change
changing
changes
changed
changer

chang

change
changing
changes
changed
changer

change

Indexing

- 위에서 고생하여 뽑은 Token들을 색인에 저장하는 과정
- 일반적으로 저장이라고 하지 않고 인덱싱(색인)한다고 한다.
- **Inverted Index**

Inverted Index

- 역인덱싱(Inverted Index)
- 우리가 일반적으로 RDBMS에서 사용했던 방식은 Forward Index라고 부릅니다.

docID		geo-scopeID
1		Europe
2		Europe
3		France
4		England
5		Portugal
6		Quebec
7		Europe
8		Spain

Forward Index

geo-scopeID		docID
Europe		1 2 7
France		3
Portugal		5
England		4
Quebec		6
Spain		8

Inverted Index

차이를 구분하는 법

- Forward Index는 책 앞에 있는 차례 페이지
- Inverted Index는 책 맨 뒤에 있는 색인 페이지

그래서 그게 뭔데요?

- Term(키워드)을 가지고 있는 Document ID를 저장하는 방식.
- 더 많은 저장공간을 차지하지만, 데이터가 늘어나도 빠른 속도로 검색 가능

이 방식을 왜 사용하나요?

- 예를 들어 Forward Index Scan일 경우 (query = "코로나")

Doc ID	Text	Result
1	비트코인, 금리인상 공포에 털석... 2만 1천달러대 턱걸이	일치 텍스트 없음
2	4.4조원 비트코인 곧 매물로.. 시장은 초긴장	일치 텍스트 없음
3	방학 끝나면 학교 현장 코로나 확산 우려	일치 텍스트 있음
4	경기도 코로나19 신규확진 3만1339명, 사망 19명	일치 텍스트 있음
5	브렉시트의 저주인가? 영국 소비자 물가 상승률 10% 돌파	일치 텍스트 없음

이 방식을 왜 사용하나요?

- 그러나 Inverted Index 방식일 경우 (query = "코로나")

Term	Document ID
비트코인	1, 2
금리인상	1
방학	3
학교	4
코로나	3, 4
신규확진	4
브렉시트	5
영국	5
소비자	5

질의

- 인덱싱을 했으니, 질의를 해야 한다.
- 근데 뭐가 가장 정확한 검색 결과인가요?
- Document 중 내가 가장 찾고 싶은 결과를 보고 싶어요.

Term	Document ID
비트코인	1, 2
금리인상	1
방학	3
학교	4
코로나	3, 4
신규확진	4
브렉시트	5
영국	5
소비자	5

TF-IDF

- 문서의 집합에서 특정 단어가 얼마나 중요한지 알려주는 수치적으로 나타내는 방법

$$TF(t, d) = \frac{\text{number of times } t \text{ appears in } d}{\text{total number of terms in } d}$$

$$IDF(t) = \log \frac{N}{1 + df}$$

$$TF - IDF(t, d) = TF(t, d) * IDF(t)$$

이렇게 생겼습니다.

공식 뜯어보기

- TF (Term Frequency) \rightarrow $tf(d, t)$

특정 문서 d 에서의 특정 단어 t 의 등장 횟수를 가져옴

- DF (Document Frequency) \rightarrow $df(t)$

특정 단어 t 가 등장한 문서의 수

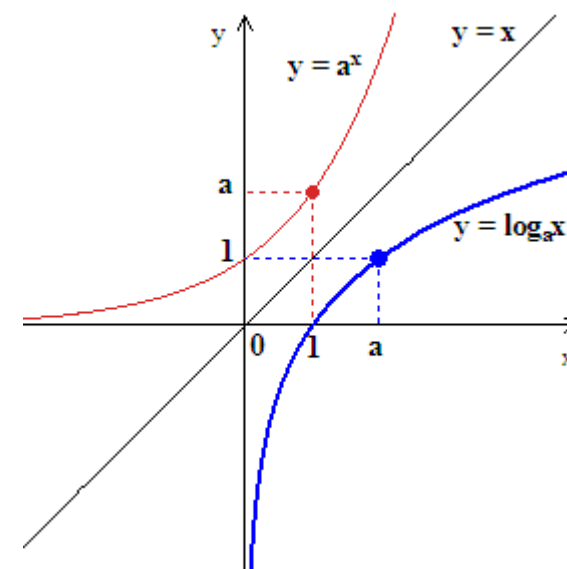
- IDF (Inverse Document Frequency)

$df(t)$ 의 반비례

$$idf(d, t) = \log\left(\frac{n}{1 + df(t)}\right)$$

IDF

- 왜 IDF는 DF의 역수가 아닐까? $idf(d, t) = \log\left(\frac{n}{1 + df(t)}\right)$
- 단순히 IDF가 DF의 역수를 취해준다면, n 이 커질수록(문서의 수가 많아질수록) IDF가 기하급수적으로 커지기에 로그를 취해 이를 방지한다.
- 1을 더해주는 이유는 $df(t)$ 가 0일 경우를 방지하기 위해서



TF-IDF

- 다시 돌아와서, TF-IDF는 IDF(문서 전체에서 Term의 중요도) * TF(문서 내의 Term 빈도)다.
- 이걸 이용해서 내가 질의한 Term과 제일 관련도가 높은 Document 대로 가져올 수 있다.
- 근데 이런 완벽해 보이는 TF-IDF에도 문제가 있으니..

TF-IDF의 한계점

- TF-IDF는 단어의 반복성(문서에서 자주 언급)과 대중성(여러 문서에서 사용)이 점수에 영향을 끼침.
- 신조어의 경우 반복성은 충족 하더라도, 대중성은 충족하기 어려움. → 검색이 잘 안됨
- 불용어가 검색 점수에 영향을 줌 (필터링이 되면 좋겠지만 완벽하지 않으므로)
- 문서 길이가 길 경우 점수에 오염을 줄 수 있음 (여러 곳에서 언급되진 않았으나 특정 문서에서만 언급이 많이 된 경우 TF-IDF에서는 점수가 높게 나올 수 있음)

Okapi BM25

- 이를 해결하기 위해 Okapi BM25를 현재 사용 중 (Elasticsearch 등)

$$\text{score}(D, Q) = \sum_{i=1}^n \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot \left(1 - b + b \cdot \frac{|D|}{\text{avgdl}}\right)}$$

해설

$$\text{score}(D, Q) = \sum_{i=1}^n \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot \left(1 - b + b \cdot \frac{|D|}{\text{avgdl}}\right)}$$

- 이 공식에는 사용자가 설정할 수 있는 변수가 있습니다. ($k = 1.2$, $b = 0.75$)

- $f(q_i, D) \rightarrow$ 문서 D 에 있는 q_i 의 빈도

- $|D| \rightarrow D$ 의 길이

- $\text{avgdl} \rightarrow$ 문서의 평균 길이

$$\text{IDF}(q_i) = \ln\left(\frac{N - n(q_i) + 0.5}{n(q_i) + 0.5} + 1\right)$$

- $\text{IDF}(q_i) \rightarrow q_i$ 에 대한 IDF weight, $n(q_i) \rightarrow q_i$ 를 포함하는 문서의 수, $N \rightarrow$ 전체 문서의 수

응용

- 이렇게 만든 검색 엔진을 응용해서 다양한 검색에 사용할 수 있습니다.
- 검색 품질을 늘리기 위한 다양한 기술 등이 있지만, 여기서는 “**한글 초성 검색**” 을 예시로 사용합니다.

초성 검색

- 검색 키워드: "ㅇㅇㅇ"
- 검색 결과: 우영우, 아이유

The image shows a YouTube search results page for the keyword "ㅇㅇㅇ". The top navigation bar includes the YouTube Premium KR logo, a search bar with "ㅇㅇㅇ" entered, and various utility icons. The search results are displayed in a grid format. The first result is a video titled "(자폐인 변호사) 우영우" (The Lawyer for the Autistic) by ENA, featuring Woo Young-woo. The second result is a video titled "아이유(IU)의 킬링보이스를 라이브로!" (IU's Killing Voice Live!) by dingo, featuring IU. The third result is a playlist titled "아이유 노래모음 30곡 (가사포함) | IU Playlist 30 Songs (Korean Lyrics)" by BANGGONVON. The search results are filtered by "모든 필터" (All filters).

YouTube Premium KR

ㅇㅇㅇ

모든 필터

(자폐인 변호사) 우영우

ENA

자폐스펙트럼을 가졌지만 압도적 승소율의 천재 변호사 박은빈(우영우)이 세상이 가진 편견을 박살내주는 대.꿀.잼. 필수시청 드라...

조회수 1695만회 · 1개월 전

고용

이 영상엔 유료광고가 포함되어 있습니다. 채널 ENA는 olleh tv 29번 / skylife 1번 / Btv 40번 / U+tv 72번 / LG헬로비전 45번 / 딜라이브 58 ...

자막

아이유(IU)의 킬링보이스를 라이브로! - 하루 끝, 너의 의미, 스물셋, 밤편지, 팔레트, 가을 아침, 뽀빠, Blueming, 에잇, Coin, 라일락 | ...

조회수 4875만회 · 1년 전

딩고 뮤직 / dingo music

0:00 하루 끝 02:05 금요일에 만나요 03:09 너의 의미 04:55 스물셋 06:18 밤편지 07:57 팔레트 09:10 가을 ...

4K 자막

하루 끝 | 금요일에 만나요 | 너의 의미 | 스물셋 | 밤편지 | 팔레트 | 가을 아침 | ... 첩터 13

아이유 노래모음 30곡 (가사포함) | IU Playlist 30 Songs (Korean Lyrics)

조회수 482만회 · 9개월 전

방공원 PLAYLIST

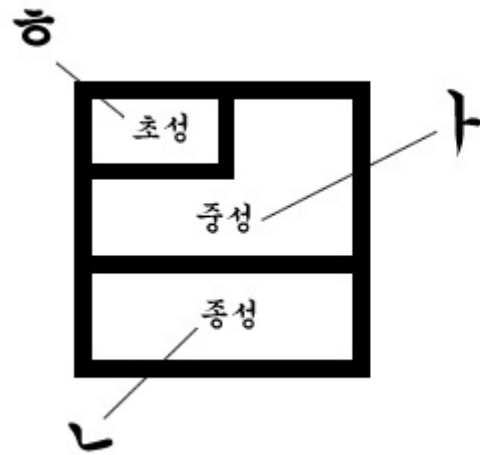
구독, 좋아요, 댓글은 영상 제작에 큰 힘이 됩니다. ✓ 해당 채널은 수익을 창출하지 않으나, 조회수 1만 회 이상부터 광고가 표시될 수 ...

그럼 어느 과정에서 처리해야 할까요?

- Tokenization 후 추출된 초성 값을 Indexing 하면 끝!

한글 처리 방법

- 한글은 초성, 중성, 종성으로 구성되어 있습니다.



Exam.

ㅎ	ㅏ	ㅑ
ㅓ	ㅕ	ㅗ
ㅗ	ㅛ	ㅜ

Module Form.

ㅏ	ㅑ	ㅓ	ㅕ	ㅗ	ㅛ	ㅜ	ㅠ	ㅡ	ㅣ
ㅓ	ㅕ	ㅗ	ㅛ	ㅜ	ㅠ	ㅡ	ㅣ	ㅏ	ㅑ

초성 추출

- 유니코드 한글 시작 44032
- 유니코드 한글 끝 55199
- 초성 시작 단어 4352
- 한글의 경우 자모 결합 가지수가 각 초성당 588개

공식

[(내가 원하는 글자) - 초성 시작 단어(44032) / 결합 가짓수(588)]
+ 초성 시작 단어(4352)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F																	
0	115F 1100 1101 11A8 11A9 11FA 1116 1158 11C7 07C8 07C9	1100 1101 11A8 11A9 11FA 1116 1158 11C7 07C8 07C9	1101 11A8 11A9 11FA 1116 1158 11C7 07C8 07C9	1102 1113 1114 1115 1116 1117 1118 1119 111A 111B	1103 1114 1115 1116 1117 1118 1119 111A 111B 111C	1104 1115 1116 1117 1118 1119 111A 111B 111C 111D	1105 1116 1117 1118 1119 111A 111B 111C 111D	1106 1117 1118 1119 111A 111B 111C 111D	1107 1118 1119 111A 111B 111C 111D	1108 1119 111A 111B 111C 111D	1109 111A 111B 111C 111D	110A 111B 111C 111D	110B 111C 111D	110C 111D	110D 111E	110E 111F																	
1	07D1 07D2 07D3 07D4 07D5 07D6 07D7 07D8 07D9	07D1 07D2 07D3 07D4 07D5 07D6 07D7 07D8 07D9	07D2 07D3 07D4 07D5 07D6 07D7 07D8 07D9	11A7 11A8 11A9 11AA 11AB 11AC 11AD 11AE 11AF	1104 1105 1106 1107 1108 1109 110A 110B 110C	1105 1106 1107 1108 1109 110A 110B 110C 110D	1106 1107 1108 1109 110A 110B 110C 110D	1107 1108 1109 110A 110B 110C 110D	1108 1109 110A 110B 110C 110D	1109 110A 110B 110C 110D	110A 110B 110C 110D	110B 110C 110D	110C 110D	110D 110E	110E 110F																		
2	07E1 07E2 07E3 07E4 07E5 07E6 07E7 07E8 07E9	07E1 07E2 07E3 07E4 07E5 07E6 07E7 07E8 07E9	07E2 07E3 07E4 07E5 07E6 07E7 07E8 07E9	11B7 11B8 11B9 11BA 11BB 11BC 11BD 11BE 11BF	1107 1108 1109 110A 110B 110C 110D 110E 110F	1108 1109 110A 110B 110C 110D 110E 110F	1109 110A 110B 110C 110D 110E 110F	110A 110B 110C 110D 110E 110F	110B 110C 110D 110E 110F	110C 110D 110E 110F	110D 110E 110F	110E 110F	110F 1110	1110 1111	1111 1112																		
3	11D1 11D2 11D3 11D4 11D5 11D6 11D7 11D8 11D9	11D1 11D2 11D3 11D4 11D5 11D6 11D7 11D8 11D9	11D2 11D3 11D4 11D5 11D6 11D7 11D8 11D9	1117 1118 1119 111A 111B 111C 111D 111E 111F	1107 1108 1109 110A 110B 110C 110D 110E 110F	1108 1109 110A 110B 110C 110D 110E 110F	1109 110A 110B 110C 110D 110E 110F	110A 110B 110C 110D 110E 110F	110B 110C 110D 110E 110F	110C 110D 110E 110F	110D 110E 110F	110E 110F	110F 1110	1110 1111	1111 1112																		
4	11E1 11E2 11E3 11E4 11E5 11E6 11E7 11E8 11E9	11E1 11E2 11E3 11E4 11E5 11E6 11E7 11E8 11E9	11E2 11E3 11E4 11E5 11E6 11E7 11E8 11E9	1117 1118 1119 111A 111B 111C 111D 111E 111F	1107 1108 1109 110A 110B 110C 110D 110E 110F	1108 1109 110A 110B 110C 110D 110E 110F	1109 110A 110B 110C 110D 110E 110F	110A 110B 110C 110D 110E 110F	110B 110C 110D 110E 110F	110C 110D 110E 110F	110D 110E 110F	110E 110F	110F 1110	1110 1111	1111 1112																		
5	11F1 11F2 11F3 11F4 11F5 11F6 11F7 11F8 11F9	11F1 11F2 11F3 11F4 11F5 11F6 11F7 11F8 11F9	11F2 11F3 11F4 11F5 11F6 11F7 11F8 11F9	1117 1118 1119 111A 111B 111C 111D 111E 111F	1107 1108 1109 110A 110B 110C 110D 110E 110F	1108 1109 110A 110B 110C 110D 110E 110F	1109 110A 110B 110C 110D 110E 110F	110A 110B 110C 110D 110E 110F	110B 110C 110D 110E 110F	110C 110D 110E 110F	110D 110E 110F	110E 110F	110F 1110	1110 1111	1111 1112																		
6	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1117 1118 1119 111A 111B 111C 111D 111E 111F 111G 111H 111I 111J 111K 111L 111M 111N 111O 111P 111Q 111R 111S 111T 111U 111V 111W 111X 111Y 111Z	1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110S 110T 110U 110V 110W 110X 110Y 110Z	110T 110U 110V 110W 110X 110Y 110Z	110U 110V 110W 110X 110Y 110Z	110V 110W 110X 110Y 110Z	110W 110X 110Y 110Z	110X 110Y 110Z	110Y 110Z	110Z
7	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1117 1118 1119 111A 111B 111C 111D 111E 111F 111G 111H 111I 111J 111K 111L 111M 111N 111O 111P 111Q 111R 111S 111T 111U 111V 111W 111X 111Y 111Z	1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110S 110T 110U 110V 110W 110X 110Y 110Z	110T 110U 110V 110W 110X 110Y 110Z	110U 110V 110W 110X 110Y 110Z	110V 110W 110X 110Y 110Z	110W 110X 110Y 110Z	110X 110Y 110Z	110Y 110Z	110Z
8	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1100 1101 1102 1103 1104 1105 1106 1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1117 1118 1119 111A 111B 111C 111D 111E 111F 111G 111H 111I 111J 111K 111L 111M 111N 111O 111P 111Q 111R 111S 111T 111U 111V 111W 111X 111Y 111Z	1107 1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1108 1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	1109 110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110A 110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 110W 110X 110Y 110Z	110B 110C 110D 110E 110F 110G 110H 110I 110J 110K 110L 110M 110N 110O 110P 110Q 110R 110S 110T 110U 110V 11																								

구현된 것을 보도록 할까요

- <https://github.com/bunseokbot/simple-python-search-engine>

```
D:\dev\simple_search_engine>python engine.py
Term: militia
A well regulated Militia, being necessary to the security of a free State, the right of the people to keep and bear Arms, shall not be infringed.
TF-IDF Score: 1.2039728043259361
BM25 Score: 1.477104722757996
No person shall be held to answer for a capital, or otherwise infamous crime, unless on a presentment or indictment of a Grand Jury, except in cases arising in the land or naval forces, or in the Militia, when in actual service in time of War or public danger; nor shall any person be subject for the same offence to be twice put in jeopardy of life or limb; nor shall be compelled in any criminal case to be a witness against himself, nor be deprived of life, liberty, or property, without due process of law; nor shall private property be taken for public use, without just compensation.
TF-IDF Score: 1.2039728043259361
BM25 Score: 0.8018220859122895

Term: 대한민국
대한민국은 민주공화국이다.
TF-IDF Score: 0.8266785731844679
BM25 Score: 1.1425736584735233
대한민국의 주권은 국민에게 있고, 모든 권력은 국민으로부터 나온다.
TF-IDF Score: 0.8266785731844679
BM25 Score: 0.878147415942302
대한민국의 국민이 되는 요건은 법률로 정한다.
TF-IDF Score: 0.8266785731844679
BM25 Score: 0.9987417682230217
대한민국의 영토는 한반도와 그 부속도서로 한다.
TF-IDF Score: 0.8266785731844679
BM25 Score: 0.987441476353444
대한민국은 통일을 지향하며, 자유민주적 기본질서에 입각한 평화적 통일정책을 수립하고 이를 추진한다.
TF-IDF Score: 0.8266785731844679
BM25 Score: 0.7434874087174042
대한민국은 국제평화의 유지에 노력하고 침략적 전쟁을 부인한다.
TF-IDF Score: 0.8266785731844679
BM25 Score: 0.905480774519078
```

한장으로 정리해 보요

- 검색 엔진의 3요소는 수집(크롤링), 색인(인덱싱), 그리고 질의(쿼리)이다.
- 색인 과정에서 전처리와 Tokenizing 을 포함한 여러 단계를 거쳐 색인된다.
- 색인 된 이후에는 질의를 통해 데이터를 검색할 수 있다.
- 내 질의 내용에 대해 우선순위별로 스코어링 할 수 있는 TF-IDF, Okapi BM25가 있다.
- 실제로 써야 한다면 만들지 말고 사드세요..

End of Document

@austinkim