
카테고리를 이용한 검색어 알고리즘

201600226 김세민

목차

배경 및
필요성

알고리즘

발전 가능성 및
보완점

기존의 티어링, HashSort, 등의 복잡한 알고리즘을 가장 간단하게
표현해 **탐색 횟수와 DB사용량을 줄여 검색 효율을 높일 수 있는 방법**



각 사용자의 **검색어와 시간, 횟수만으로**
일시간 검색어, 개인별 트렌드, 통합트렌드, 연관검색어를
도출하는 알고리즘

목차

배경 및
필요성

알고리즘

발전 가능성 및
보완점

점액어 우선순위 알고리즘

개인별

최근 해당 분야
검색 빈도

분야	C ₁	C ₂	C ₃	C ₄	...	C _n
가중치 빈도 가중치(FW)	FW ₁	FW ₂	FW ₃	FW ₃	...	FW _n

시간의 흐름

분야	C	C	C	C	...	C
가중치 횟수 가중치(CW)	CW ₁	CW ₂	CW ₃	CW ₃	...	CW _n

동 검색 횟수 내림차순 정렬

분야별 총 검색 횟수

C_n의 노출 비율

=

C_n의 가중치 값
전체 가중치 값

=

$$\frac{C_n(FW+CW+TW)}{\sum (FW_k+CW_k+TW_k)}$$

목차

배경 및
필요성

알고리즘

발전 가능성 및
보완점

검색어 우선순위 알고리즘

개인별

User1

카테고리1 -Σ

카테고리2 -Σ

카테고리3 -Σ

각 유저별 개인의 리스트 생성
(가중치 값에 따른)

0/search/list?user_uid=1

카테고리	검색횟수	시간	가중치
1	19	2022-11-14 07:59:22	12
3	12	2022-11-10 07:31:49	9
2	5	2022-11-10 04:04:03	5
4	4	2022-11-10 04:04:28	4
5	3	2022-11-09 09:23:23	0

유저 아이디 값의 따라 각각의 리스트 출력.

목차

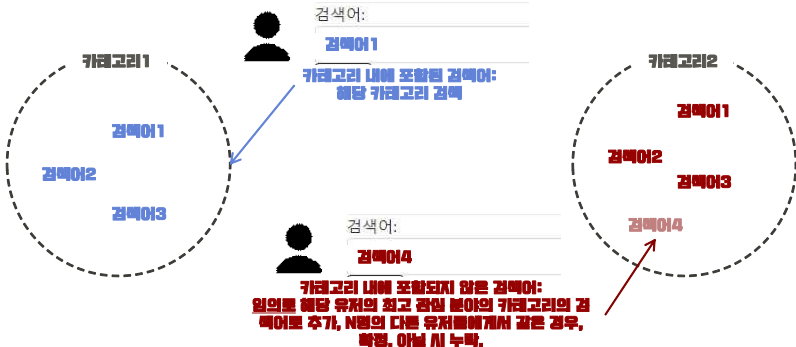
배경 및
필요성

알고리즘

발전 가능성 및
보완점

검색어 우선순위 알고리즘

추천 검색어 -sample data



목차

배경 및
필요성

알고리즘

발전 가능성 및
보완점

점액어 우선순위 알고리즘

통합 (트렌딩)



Category1

유저1 - Σ

유저2 - Σ

유저3 - Σ

⋮



각 카테고리의 개인 별 **가중치 합산**으로 **통합 트렌딩** 리스트 생성.

카테고리	가중치(카테고리별 합산)
1	60
5	31
4	29
3	19
2	11

개인별 가중치들을 **카테고리별로 합산**해 **트렌딩** 카테고리를 우선 출력.

목차

배경 및
필요성

알고리즘

발전 가능성 및
보완점

실시간 검색어



Keyword

각 키워드 별 **최근 검색량**에 따른 순위.

현재시간 > **집계 시간** > (현재시간 - **시간변수**)

순위	검색어	카테고리num
1	웹스	1
2	오라클	1
3	데이터	1
4	SQL	1
5	스프링	2
6	부트	2
7	JSP	2

시간변수 조정을 통해
일자별, 시간별 통으로 분류 가능.

목차

배경 및
필요성

알고리즘

발전 가능성 및
보완점

N 검색

- ① 검은사막 인벤
- ② 검은태양
- ③ 검색어를 입력하세요

AJAX를 통한 실시간 자동완성 기능

```
public static final int UpdateWeightValue = 1;  
public static final int RealtimeListValue = 0;
```

적당한 가중치 변수의 값을 특정짓지 못하여 고정형 변수로 사용중,
유저수 및 검색량의 추이에 따른 가변형 변수로 보다 정확한 우선순
위의 출력.

부실한 UI, 로그인 시큐리티의 보완