mysqlでの全文検索での検索の仕方

@vividmuimui

2018/06/06 社内LT資料

はじめに

話すこと

- クエリ周りの話

話さないこと

- 他の全文検索システムとの違いとか
- mysqlでの全文検索の組み込み方
- text perserの話
  - ngram, mecab

例で使うテーブル

```
SELECT body, title FROM posts;
```

body	title
美味しいお肉とお魚が食べたい	肉と魚
美味しいお肉が食べたい。例えば松阪牛	肉
美味しいお魚が食べたい。例えばサバ味噌	お魚
あああああああああああ	全く関係ない記事

Field	Type	Null	Key	Default	Extra
title	varchar(255)	NO	MUL	NULL	
body	text	NO		NULL	

body, titleで全文検索用のindexが貼ってある

```
FULLTEXT KEY `index_posts_on_title_and_body` (`title`,`body`)
  COMMENT '日本語全文検索インデックス(ngram)' /*!50100 WITH PARSER `ngram` */ ,
```

クエリの基本

```
SELECT *
FROM `posts`
WHERE MATCH (`posts`.title, `posts`.body)
  AGAINST ('+お肉 +お魚' IN BOOLEAN MODE);
```

```
MATCH (col1,col2,...) AGAINST (expr [search_modifier])
```

- (col1,col2,...) 検索対象のカラム。複数設定可
- expr 検索する文字列
- search_modifier 検索のモードの設定

search_modifier の種類

3種類ある

- 自然言語検索 IN NATURAL LANGUAGE MODE
  - デフォルトの修飾子
- Bool検索 IN BOOLEAN MODE
- クエリ拡張での検索 WITH QUERY EXPANSION or IN NATURAL LANGUAGE MODE WITH QUERY EXPANSION

1️⃣ 自然言語検索

IN NATURAL LANGUAGE MODE

検索文字列がそのまま検索される(この表現はだいぶ怪しい・・・)
検索対象のカラムのテキストと類似度が評価される。
類似度が高い順にソートされ返される。

```
SELECT body FROM `posts`
WHERE MATCH (`posts`.title,`posts`.body)
  AGAINST ('松阪牛を焼いた美味しいお肉が食べたい' IN NATURAL LANGUAGE MODE);
```

body
美味しいお肉が食べたい。例えば松阪牛
美味しいお肉とお魚が食べたい
美味しいお魚が食べたい。例えばサバ味噌

- **松阪牛を焼いた美味しいお肉が食べたい**という文字列自体はどれもいないが、類似度で検索結果が決まるので、3件返ってくる
- **あああああああああああ**は結果に入っていない

類似度の確認

```
SELECT
  body,
  MATCH (`posts`.title,`posts`.body)
    AGAINST ('松阪牛を焼いた美味しいお肉が食べたい' IN NATURAL LANGUAGE MODE)
    AS score
FROM `posts`
ORDER BY score DESC;
```

body	score
美味しいお肉が食べたい。例えば松阪牛	1.30292546749115
美味しいお肉とお魚が食べたい	0.2154965400695800 8
美味しいお魚が食べたい。例えばサバ味噌	0.12487750500440598
あああああああああああ	0

自然言語検索の感想

こういう検索方法だとわかっているとと
とりあえぞそれっぽい単語をいっぱい並べれば目的の検索結果をより引き当てやすくなる

ヘルプやFAQ系の検索とかも有用かも？

2️⃣ Bool検索

IN BOOLEAN MODE

googleとかtwitterとかのいわゆる普通の検索のモード

- +でAND, -でNOTなどの演算子をサポートしている
- '+お肉 +お魚'というように、検索したい文字列の前に演算子をつけて検索する
- 'お肉 お魚'のように演算子がない場合は、お肉またはお魚が入っている文章、というようにORでの検索になる
- 自然言語検索 IN NATURAL LANGUAGE MODEとは違ってソートは勝手にはされない

```
SELECT body FROM `posts`
WHERE MATCH (`posts`.title, `posts`.body)
  AGAINST ('+お肉 +お魚' IN BOOLEAN MODE);
```

body

美味しいお肉とお魚が食べたい

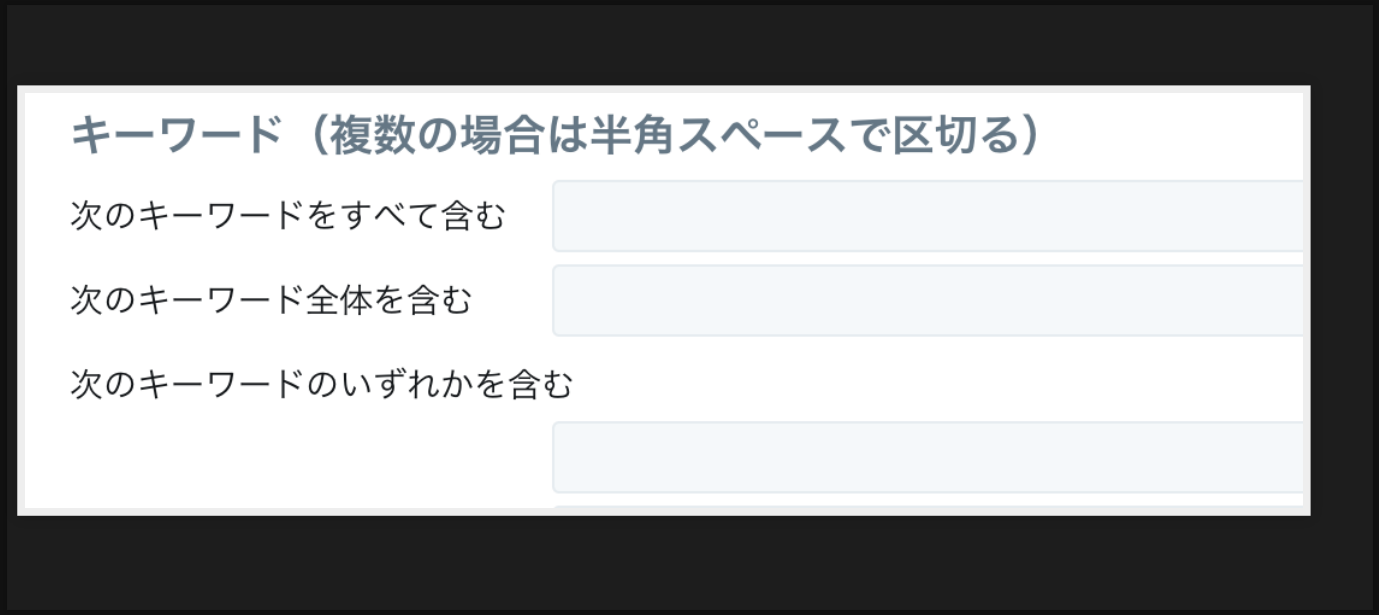
演算子の種類

- + 存在しなければならない
- - 含まれない
- **演算子なし** OR
- @distance
  - '"word1 word2 word3" @8'
  - @distance演算子前の"で囲まれた文字列が指定された距離内にいることを指定できる
- > <
  - 貢献度を変更する
    - 類似度と同様に一定以下だと検索結果として返さなくなる(???)
- ( ) グループ化
- ~
  - 貢献度をマイナスにする
  - -よりはソフト
- *
  - 難しい...

- "
  - 入力されたそのままのフレーズを含む行にのみ一致
  - '"some words"'

Bool検索の感想

演算子を使いこなせば以下のような、いわゆる高度な検索は作ることができそう



3️⃣ クエリ拡張での検索

WITH QUERY EXPANSION or IN NATURAL LANGUAGE MODE WITH QUERY

databasesと検索したときにMysql Oracle RDBMSなどが結果として返ってきてほしいケースに使う

```
SELECT body FROM `posts`
WHERE MATCH (`posts`.title, `posts`.body)
  AGAINST ('お肉' WITH QUERY EXPANSION);
```

body
美味しいお肉が食べたい。例えば松阪牛
美味しいお魚が食べたい。例えばサバ味噌
美味しいお肉とお魚が食べたい

お肉で検索したが、**お肉**という文字列が含まれてないものも検索結果に出てくる

スーパーざっくりとした仕組み

内部では2回検索が行われる
検索キーワードが databasesのとき

- 1回目はdatabasesで検索
- 1回目の検索で関連度の高い検索結果の文字列をもとに新しい検索文字列を作成する
- 新しい検索文字列で検索をし検索結果を返す

クエリ拡張での検索の感想

> *関連性のないドキュメントが返されるとノイズが大幅に増加する傾向があるため、検索フレーズが短すぎる場合にのみ使用してください*

ドキュメントに上記のように書いてあるように使い所は難しそう。
google検索とかの **もしかして 〇〇?**みたいな機能ができるかも・・？

今回省略した話

- 大文字小文字は区別して検索される？
- ストップワード
- 類似度・貢献度の計算方法、閾値