daten.berlin.de Searchterms

Figure 1: logo for “daten.berlin.de searchterms” dataset

This dataset contains the searchterms that users looked for on the Berlin Open Data Portal (https://daten.berlin.de). Terms are collected per month (starting in February 2019, when we started using our new analytics software), and ranked by how often they were searched (i.e., the number of page impressions).

Requirements

The code to extract the searchterm statistics is written in Ruby. It has been tested with Ruby 2.7.1.

The required gems are defined in the Gemfile. In particuler, these are:

- webtrekk_connector
- ruby-keychain
- activesupport

If you have bundler, you can install the required gems as follows:

```
bundle install
```

daten_berlin_de.searchterms.json

Download here: daten_berlin_de.searchterms.json

For each searchterm that was entered in a given month, the page impressions, visits, average page duration (in seconds) and exit rate (%) are listed.

The following example illustrates the structure of the data file:


```
{
  "timestamp": "2020-05-29T15:21:32+02:00",
  "source": "Webtrekk",
  "stats": {
    "site_uri": "daten.berlin.de",
    "earliest": "2019-02",
    "latest": "2020-04",
    "months": {
      "2020-04": {
        "terms": {
          "corona": {
            "impressions": 27,
            "visits": 20,
```

```
            "page_duration_avg": 36.81,
            "exit_rate": 20.0
          },
          "verkehr": {
            "impressions": 24,
            "visits": 8,
            "page_duration_avg": 38.08,
            "exit_rate": 0.0
          },
          ...
          "new york": {
            "impressions": 1,
            "visits": 1,
            "page_duration_avg": 0.0,
            "exit_rate": 0.0
          }
        },
        "removed_items": {
          "comment": "Removed 13 searchterms as potentially personal information.",
          "count": 13
        }
      },
      "2020-03": {
          ...
      },
      ...
    }
  }
}
```


Filtering Personal Information

All searchterms that potentially contain personal information are removed from the data before publishing it here.

In particular, the following categories of searchterms are removed:

- personal names
- (postal) addresses
- geographic coordinates
- personal e-mail adresses
- phone numbers
- land lots (German „Flurstück“)

Blocklist

Instances of these categories are currently not detected automatically, but rather manually via the use of a blocklist (not included in this repository), which is being extended each time the dataset is updated (i.e., every month).

Allowlist

There are exceptions where searchterms are included in the data, even though they belong to one of the exclusion categories. In particular, we allow the following kinds of searchterms:

- **Personal names of public figures**
  The criterion for being a public figure is: there is a (stable) Wikipedia page for that person. The criteria for people to have Wikipedia page are defined here.
  Another possible criterion is that a name has an entry in a bibliographic authority file (something like a database of all known authors), such as the Gemeinsame Normdatei. In other words, a name is the name of a published author.
- **Functional e-mail addresses**
  Functional e-mail addresses (addresses not tied to a particular person, but to a role or a post such as `info@example.com`, `opendata@berlin.de` etc.) do not contain personal information and can therefore be included.

Searchterm Normalization

Searchterms are currently not normalized in any way. This means that different spellings of the same term (most importantly: differences in case) are treated as different searchterms. It is possible to sum page impressions for each term. This is not possible for visits, because the same visit might include two or more searchterms, and so the actual number of visits for a set of searchterms might be less than the sum of visits for all of them.

For example:

```
{
...
  "terms": {
    "corona": {
      "impressions": 27,
      "visits": 20,
      "page_duration_avg": 36.81,
      "exit_rate": 20.0
    },
    ...
    "Corona": {
      "impressions": 8,
```

```
      "visits": 6,
      "page_duration_avg": 21.25,
      "exit_rate": 0.0
    },
    ...
    "covid": {
      "impressions": 2,
      "visits": 2,
      "page_duration_avg": 13.0,
      "exit_rate": 0.0
    },
...
}
```


People searched for `corona` (lower case), `Corona` (upper case), `covid` and possibly other related searchterms. It would be valid to say that the total number of page impressions for all spellings of `corona` is `27+8=35`, and `27+8+2=37` for all Corona-related searches. However, the total number of visits for all spellings of `corona` is `20+6=26` *or less* because some of these searches may have occured within the same visit.

How to Update the Search Data with a New Month

Because adding a new month to the data involves manually editing the blocklist and allowlist, it is more complicated than just running a make target. Here is what needs to be done:

Extract the Unfiltered Data

```
make unfiltered
```

This will extract the search data from Webtrekk Analytics up until the last day of the previous month. The ouput is written to `data/temp/daten_berlin_de.searchterms.unfiltered.json`.

Extract a List of Searchterms for the Previous Month

We need to manually go through all the new search terms, pick those that are problematic and then either add them to the blocklist or allowlist. To generate a simple list of terms (without hits, visits etc.) for a given month, do `make data/temp/terms_YYYY-MM.json`, e.g.:

```
make data/temp/terms_2020-06.json
```

Select Problematic Terms

Manually go through the list and extract all potentially problematic search terms. What I do is simply delete all *unproblematic* ones, leaving me with the list of

problematic ones.

Update blocklist and allowlist

For each problematic term, decide if it really needs to be filtered out or if maybe it should be allowed after all. See Filtering Personal Information.

Each new addition to the blocklist simply needs to be added to the appropriate category (though the categories are just a way to structure the list for humans, they are not used otherwise).

Each new addition to the allowlist looks like this:

```
{
  ...
    "friedrich wilhelm förster": {
        "variants": [
            "friedrich wilhelm foerster",
            "friedrich wilhelm förster"
        ] ,
        "reference": [
            "https://de.wikipedia.org/wiki/Friedrich_Wilhelm_Foerster" ,
            "https://d-nb.info/gnd/118692038"
        ]
    }
  ...
```

So, either add a new variant to an existing entry or create a new one.

Remarks:

- It's possible that an entry has only one `variant`.
- The entry's key and the grouping of variants are irrelevant, just a way to structure the list for a human reader.
- Also, the `reference` is technically not necessary, but helpful as a reminder why the decision was made to include a searchterm in the allowlist.

Create Final, Filtered Data

Now that we have the updated block- and allowlist, we can filter the data and create the final data file, which goes into `data/target`.

```
make final
```

Logo

- search logo by FontAwesome under CC BY 4.0.

License

Dataset URL: https://daten.berlin.de/datensaetze/suchbegriffe-datenberlinde

This page was generated from the github repository at https://github.com/berlinonline/berlin_dataportal_searchterms.

Last changed: 2024-05-31