Report

1.Steps

1.1 Spell Correction

The spell correction of this search engine is based on Norvig's spelling program. After downloading php version from the given website, I mainly did the following tasks:

- Download Apache Tika jar "tika-app-1.19.1.jar" as parser.
- Parse the html files of NYPost and generate "big.txt" by using the command: "find /Users/xuechengzhe/nypost -name "*html" -print0 | xargs -0 java -jar tika-app-1.19.1.jar -T >> big.txt".
- Implement the function of spell correction. The main idea is: split the query into words, for each word, do the correct operation and add the corrected words together as the new corrected query. If the new query is the same as the old one, do nothing; if not, prompt "Did you mean" the new query.
- "Train" the search engine by using some queries to generate "serialized_dictionary.txt", hence expedite the future query.

1.2 Autocomplete

The autocomplete function is realized by using Solr's internal autocomplete function as the following steps:

- Add a search component to solrconfig.xml to tell Solr to use the SuggestComponent.
- Add a request handler to solrconfig.xml to configure default parameters for serving suggestion requests.
- Implement the corresponding function in php. The main idea is by visiting the URL of suggestion of each query term " http://localhost:8983/solr/myexample/suggest?q=" + term, get the suggested word for this term and show the five suggestions when typing each query term in the fall-down box.

1.3 Snippets

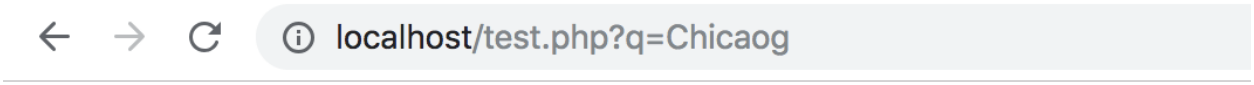
For generating snippets, we need to visit each result's webpage and find the first snippets including full or part of the query. The main steps are as followings:

- Download "simple_html_dom.php" from website and use it to parse the according html files when generating snippets.
- Search the parsed web line by line to find the query terms. When find the line containing the terms, locate the first index and set the term to the middle of the snippets, add "…." to the start or end of the snippets if necessary.
- Change the solrconfig.xml for making 'AND' as default instead of 'OR' for multi-word queries in solr.

2. Result Analysis

2.1 Spell Correction

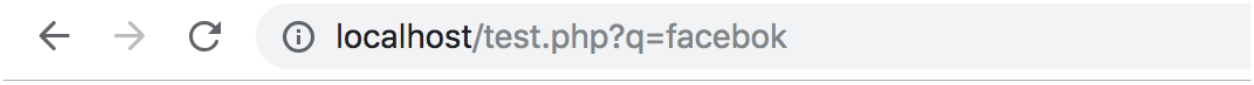
- query: "Chicaog" corrected: "chicago"



Search: Chicaog Algorithm: Solr Lucene PageRank Submit

Did you mean: chicago ?
Results 0 - 0 of 0 using Solr Lucene:

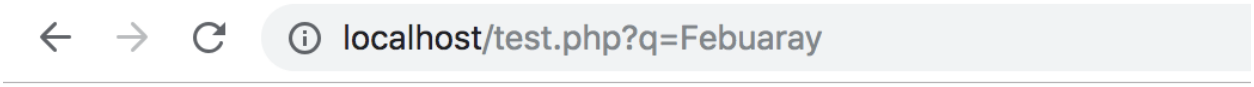
- query: "facebok" corrected: "facebook"



Search: facebok Algorithm: Solr Lucene PageRank Submit

Did you mean: facebook ?
Results 0 - 0 of 0 using Solr Lucene:

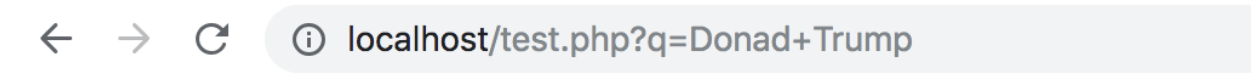
- query: "Febuaray" corrected: "february"



Search: Febuaray Algorithm: Solr Lucene PageRank Submit

Did you mean: february ?
Results 0 - 0 of 0 using Solr Lucene:

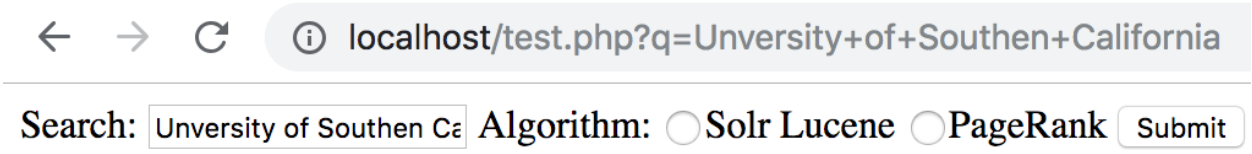
- query: "Donad Trump" corrected: " donald trump"



Search: Donad Trump Algorithm: Solr Lucene PageRank Submit

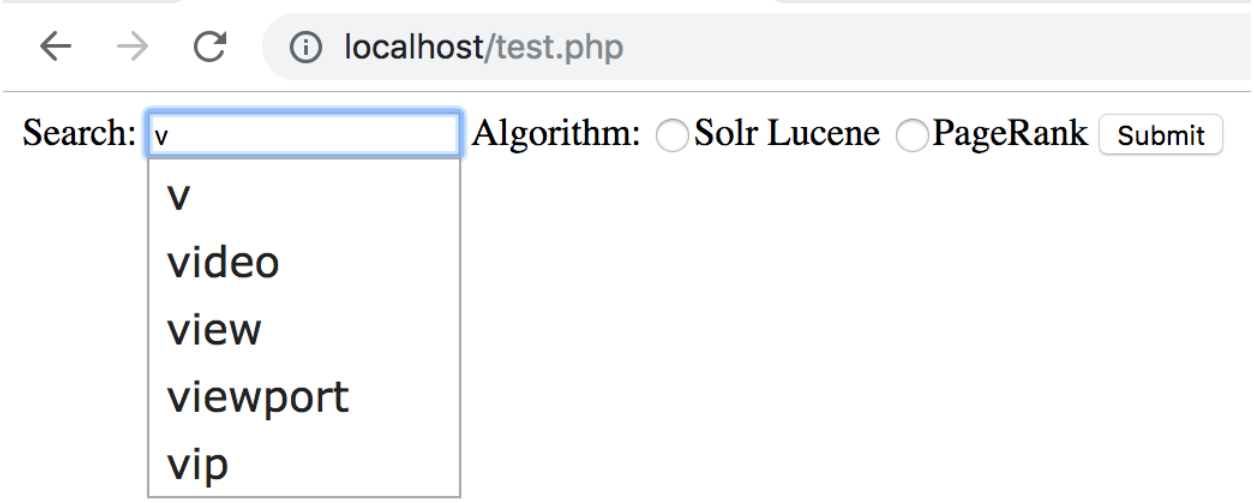
Did you mean: donald trump ?
Results 1 - 10 of 7571 using Solr Lucene:

- query: "Unversity of Southen California" corrected: " university of southern california"

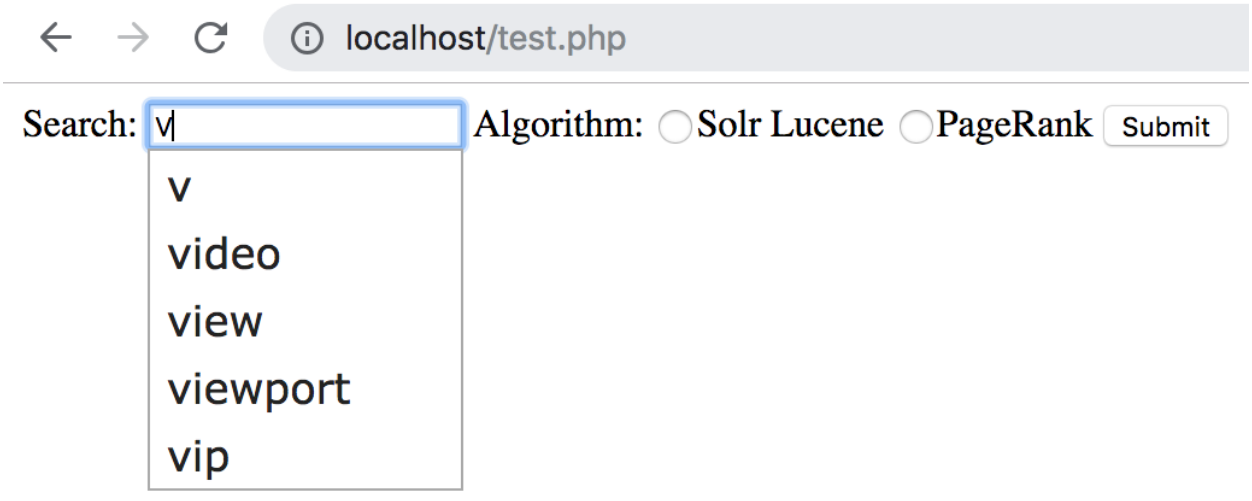


2.2 Autocomplete

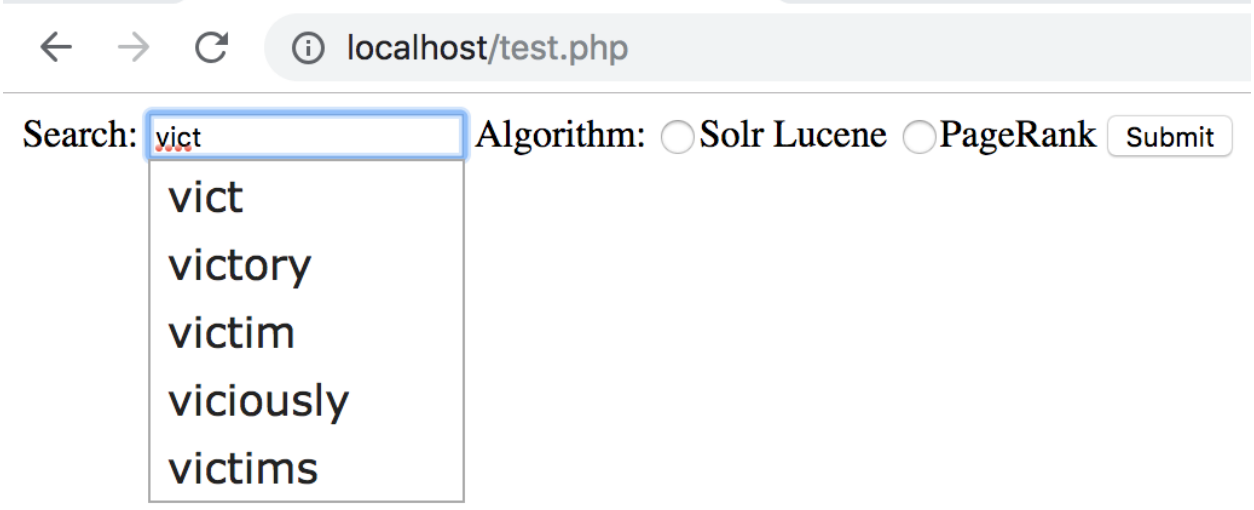
- type: v (lower case)



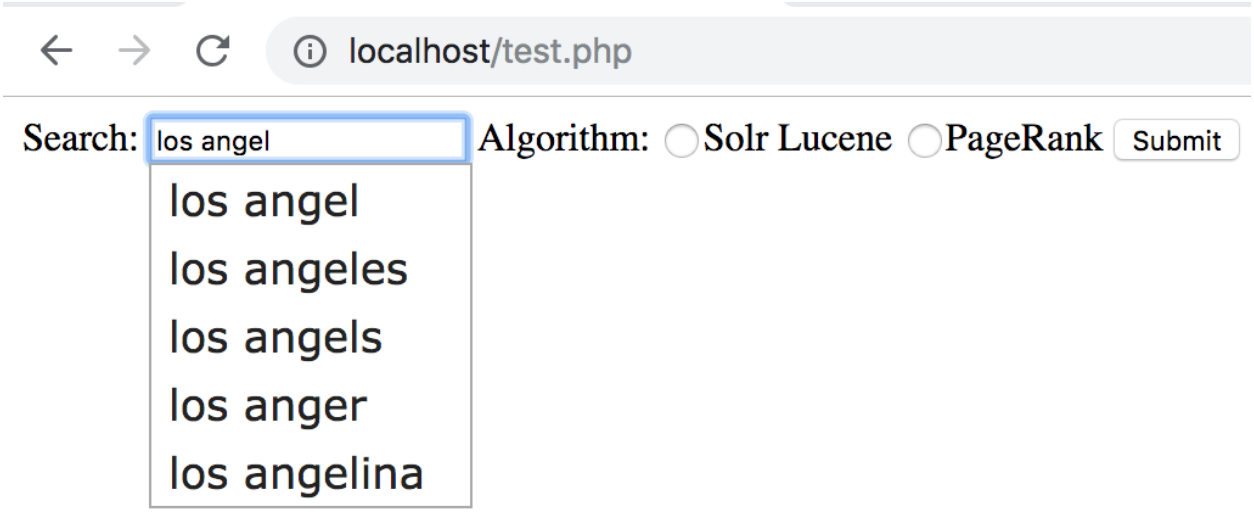
- type: V (upper case)



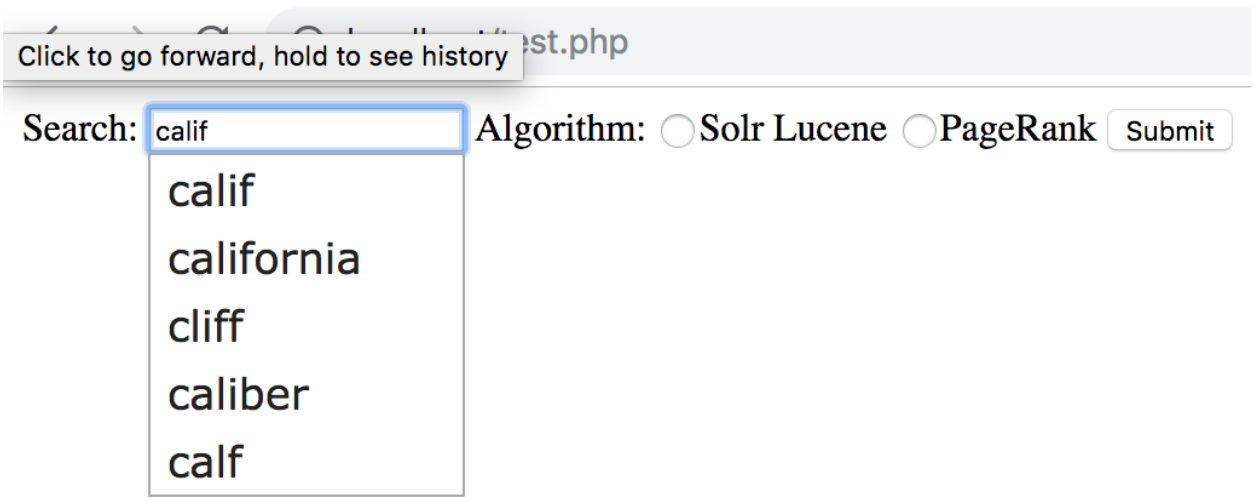
- type: vic



- type: los angel



- type: calif


Click to go forward, hold to see history
st.php
Search: calif
Algorithm: Solr Lucene PageRank Submit
calif
california
cliff
caliber
calf