

Term-Based Suggester

You can implement a typeahead search experience through a term-based suggester. This allows you to pre-define search terms to return as suggestion results as users are actively typing, and provide immediate feedback to help users find what they are looking for quicker.

Terms are defined in a text-based file that acts as a dictionary. This dictionary is based off of the Solr suggester component, and defines the search terms along with specifying weighting and additional payloads to return.

Creating a Term File

Search terms in the dictionary consist of up to three fields: suggestion, rank, and payload.

- **suggestion:** A string of the search term to return as a suggestion. This term is case-insensitive. Matches can occur in any position of the suggestion. For example, if the user types "hammer," then a term "claw hammer" can return as a match. This term is required.
- **rank:** A number that applies weighting to the suggestions. Suggestions with a greater rank value appear higher in the list of results. This term is optional.
- **payload:** A string that provides metadata. This metadata is returned as part of the suggestion results and can be used by your site in a custom manner. This term is optional.

Fields are separated by a tab, and terms are separated by a new line.

An example file:

```
zip ties 100 example payload
pumpkin pie 5 example payload
```

Uploading a Term File

You can upload term files through either the API or the Admin UI.

Uploading via API

You can upload a term file by specifying it with the following API call:

```
PUT catalog/admin/searchSchema/suggest/{language}
```



English is currently the only supported language.

Using Postman to make this call can help ensure that the file formatting is properly preserved during the upload. Refer to [Getting Started with Postman](#) for more information on using Postman.

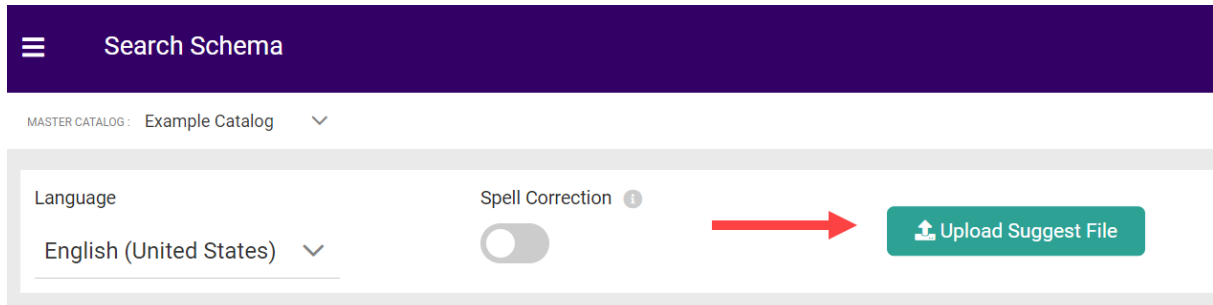
You can also download an existing term file with the following API call:

GET catalog/admin/searchSchema/suggest/{language}

Uploading via Admin

To upload a term file via the Admin UI:

1. Go to **Main > Search > Schema**.
2. Click **Upload Suggest File**.



3. Select the file from your computer and click **Open**.
4. The file will then download.

Suggest API Calls

The `suggest` and `suggest2` API calls return term-based search results. These results are returned as `suggestionType` "Term" under suggestions.

Here is an example of what is returned for the search term "wrench":

```

{
  "query": "wrench",
  "suggestionGroups": [
    {
      "name": "Terms",
      "suggestions": [
        {
          "suggestionType": "Term",
          "suggestion": {
            "term": "torque <b>wrench</b>",
            "weight": 21515
          }
        },
        {
          "suggestionType": "Term",
          "suggestion": {
            "term": "<b>wrench</b>",
            "weight": 7351
          }
        },
        {
          "suggestionType": "Term",
          "suggestion": {
            "term": "allen <b>wrench</b>",
            "weight": 6692
          }
        },
        {
          "suggestionType": "Term",
          "suggestion": {
            "term": "pipe <b>wrench</b>",
            "weight": 5431
          }
        }
      ]
    }
  ]
}

```

Settings

Settings are stored alongside the term file. The following settings are available:

- **highlight:** A boolean that specifies whether returned suggestions will have the matching term highlighted in bold. The default is true.
- **allTermsRequired:** A boolean that specifies whether all terms entered by the user must be in the suggestion for it to be returned. For example, if a user types "claw hammer," then a suggestion of "hammer" will not be returned. The default is true.

Settings can be accessed with the following API call:

```
GET catalog/admin/searchSchema/suggest/settings/{language}
```

You can change these settings by specifying new settings in the following API call:

PUT catalog/admin/searchSchema/suggest/settings/{language}

An example of the settings:

```
{  
  "name": "suggest_21378",  
  "highlight": "true",  
  "allTermsRequired": "true"  
}
```