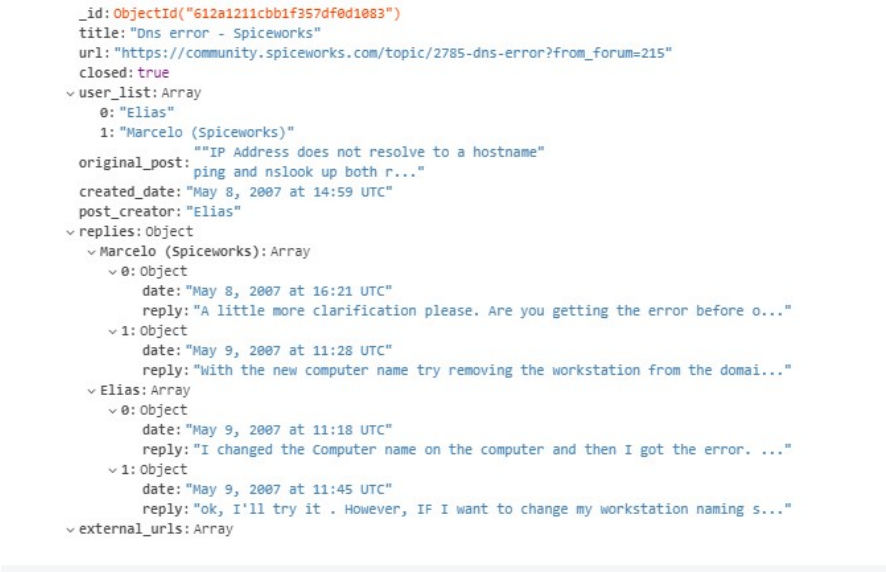
Crawlers:

Each forum webpage contains multiple post URLs. The multiprocessing crawlers extract post URLs from the forum webpages and crawl them. The crawled contents are stored by using Mongodb. Failed URLs will be retried up to three times.

Data format:



user_list: stores users who have replied to the post.

original_post: the original question / feedback wrote by the post creator.

replies: replies made by each user are stored under his or her name with date.

external_urls: stores URLs mentioned in replies or in the original post, temporarily disabled in the current crawler version.

Key functions:

start_crawling(): initializes Mongodb connections, extracts post URLs from the given forum webpage URL.

store_crawled_data(): by calling customized helper classes(refiner&parser,) three versions of the crawled data are stored on Mongodb server. Raw: stores the original HTML documents. Refined: stores parsed texts. Pure: stores tokenized texts without punctuation.

Helper classes:

refiner: handles the original HTML document and extracts post related information including date, username, original post contents, reply contents, etc. Each crawler has its own customized refiner class.

parser: uses NLTK and HTMLparser to tokenize post contents.

Key variables:

self.crawl_interval: this variable controls how long one crawler process will sleep after extracting web contents from one URL.

pageNum(under the main function): this variable controls the page range the crawler is going to crawl.

client(under the start_crawling function): this variable decides which database will be used to store the crawled datasets.

Library Versions:
nltk 3.6.1
pymongo 3.11.4
urllib3 1.25.8
beautifulsoup4 4.9.3
requests 2.25.1

Configure MongoDB server's remote connection: (MongoDB server version: 4.2.15)
sudo vim /etc/mongod.conf

```
frank@qifanz-MS-7C94:~$ sudo vim /etc/mongod.conf
```

change bindIp to 0.0.0.0; enable security authorization:

```
# network interfaces
net:
  port: 27017
  bindIp: 0.0.0.0


# how the process runs
processManagement:
  timeZoneInfo: /usr/share/zoneinfo

security:
  authorization: 'enabled'
```

To close remote connection:
change bindIp to 127.0.0.1; disable security authorization

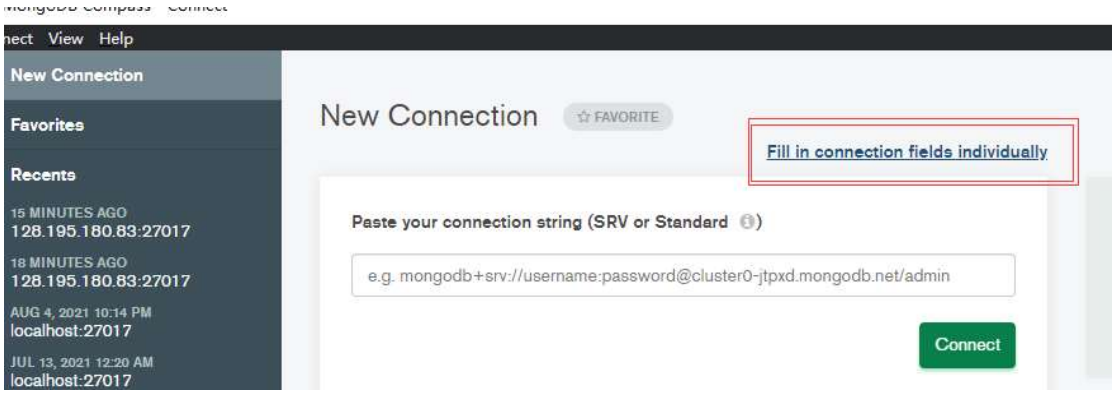
```
# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1


# how the process runs
processManagement:
  timeZoneInfo: /usr/share/zoneinfo

#security:
#  authorization: 'enabled'
```

Then restart Mongodb server:
sudo service mongod restart

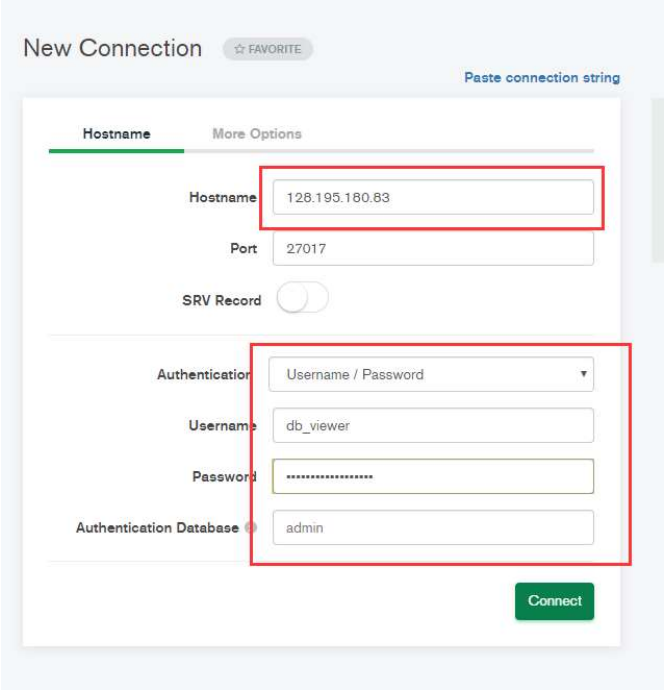
```
frank@qifanz-MS-7C94:~$ sudo service mongod restart
```

Admin: usrname: "admin", pwd: "ucidsplab_dbadmin"
Read-only: usrname: "db_viewer", pwd: "ucidsplab_dbviewer"

Remote connection by using MongoDB Compass (first time only):
Create a new connection, and select "Fill in connection fields individually":

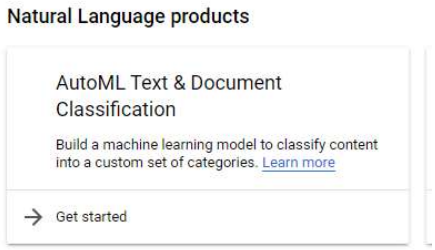


Connect by using the Read-only username & password provided above:

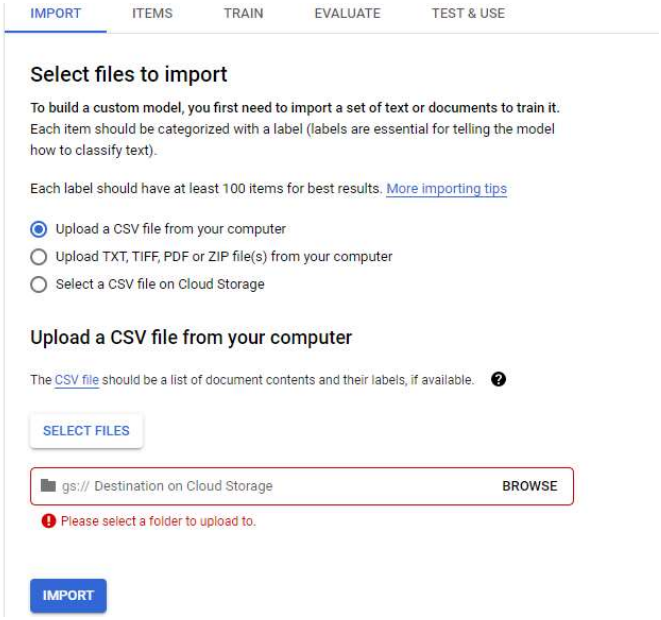


Google AutoML Natural Language:
Email Address: uci.dsp.dns.forum@gmail.com
Password: UCI_DSP_DNS_Forum

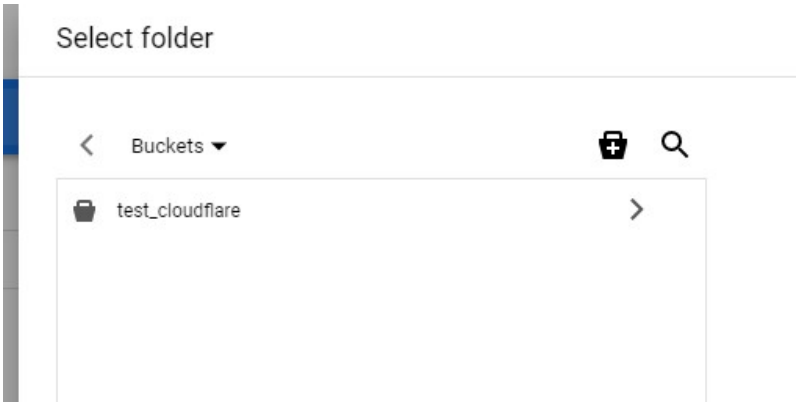
Select Text & Document Classification



Import CSV files and upload CSV files to Google Cloud Storage:



Choose an existing Cloud folder or create a new one:



After imports, assign labels to each data row and start training. The training process can take hours to complete.