CSE 435/535 Information Retrieval
Fall-2021
University at Buffalo (SUNY)

Instructor: - Dr. Rohini K. Srihari

Project 1 : Collecting and Indexing documents in Solr

Part-1 Due Date: **8th September 2021, 23:59 EST/EDT**
Final Due Date : **19th September 2021, 23:59 EST/EDT**

1. Introduction

This project aims at introducing students to analysis of online conversation. An online conversation is defined as any exchange in the form of written text between multiple users (example tweets and replies). Prior to mining and analyzing such content, collection, cleaning and storage of such data is a crucial step. This project aims at familiarizing students with collection of online conversations, and efficiently storing them, which makes data retrieval and analytics easier for downstream applications. This project will also introduce students to the different technical aspects involved in this course and subsequent projects. By the end of this project, a student would have achieved the following:

- Become familiar with AWS, and EC2 instances
- Learn about the Twitter API, and querying twitter using keywords, language filters and retrieving tweets from the user timeline. **Make sure to store your data as you will be using this data in the final project.**
- Setup a Solr (a fully functional, text search engine in Java) instance, and understand basic Solr terminology and concepts. Refer to Solr Reference Guide 8.1 which contains detailed explanation of operations you will be using in all the projects in this course.
- Understanding fundamental concepts of IR such as tokenization, indexing, search etc.
- Indexing thousands of multilingual tweets.

The specific challenges in completing this project are as below:

- Determining prominent Twitter political influencers, and collecting their tweets and replies to their tweets.
- Figuring out keywords and hashtags for effective data collection, which can be used for downstream tasks of classification, stance detection, sentiment analysis etc.
- Working with multilingual data, and learning to index and tokenize them for efficient search.
- Correctly setting up the Solr instance to accommodate language and Twitter specific requirements.

The rest of this document will guide you through the necessary setup, introduce key technical elements and then finally describe the requirements in completing the project. This is an **individual project** and all deliverables **MUST** be submitted by **19th September 2021, 23:59 EST/EDT.**

Note: The data collected and work done in this project will act as foundation for Project 4.

2. Major Tasks and Challenges

Major Task	Challenges
Twitter Developer Account	Takes maximum of 2-3 days to set up.
AWS registration and EC2 setup	
Solr Setup	
Solr	• Getting familiar with Schema files, filters and analyzers. • Understanding Solr Admin UI and how indexing works. • Understanding how queries work on Solr Admin UI.
Script for crawling and processing raw tweets	• Retrieving tweets and replies from the user timeline. • Searching using hashtags and keywords. • Search while being within Twitter rate limit. Read the rate limit guidelines carefully. • Getting familiar with JSON content and extracting fields needed for this project. • Finding ways to handle emoticons, dates and multilingual data.
Crawling 50,000 tweets with various requirements	• Starting early to meet all minimum requirements. • Handling duplicate tweets • Handling RTs • Handling replies

3. Prerequisites

The first thing you need is to set up an AWS EC2 instance where you will be hosting your project; Twitter developer account to retrieve tweets; and Solr instance for indexing operations. Please refer to "SetupGuidebook.pdf" to guide you through all the setups.

Note: Windows users will also need to install open source client Putty and file transferring tool FileZilla to access EC2 instances.

4. Collecting Twitter Data

For tweets collection, there are three main elements that you need to know with regards to using the Twitter API : Authentication, Streaming vs REST APIs and Twitter Clients. **Note that for project 1, sharing tweets will not be allowed.** We will include various checks in grading script to make sure all the students have collected their own crawled tweets. Another reminder that Academic Integrity is an important issue and the department will not be lenient, if the policy is breached. You can store the collected tweets in UBBox, for future use. Please access your UB box account here https://www.buffalo.edu/ubit/ubbox.html

4.1 Authentication

Twitter uses OAuth to authenticate users and their requests to the available HTTP services. The full OAuth documentation is long and exhaustive, so we only present the relevant details. Please refer "SetupGuidebook.pdf" to setup Twitter developer account and create access tokens.

4.2 Streaming and REST APIs

We are only concerned about querying for tweets i.e. we do not intend to post tweets or perform any other actions. To this end, Twitter provides two types of APIs : REST (which mimics search) and Streaming (that serves "Live" data).

You are encouraged to experiment with both to see which one suits your needs better. You may also need a case by case strategy search would give you access to older data and may be more useful in case sufficient volumes

don't exist at a given time instant. On the other hand, the Streaming API would quickly give you thousands of tweets within a few minutes if such volumes exist. Both APIs return a JSON response and thus, it's important that you get yourself familiarized with the different fields in the response.

Please read up on the query syntax and other details here: https://dev.twitter.com/. You may be interested in reading up on how tweets can be filtered based on language and/or geolocation. These may help you in satisfying your language requirements fairly quickly.

Similarly, you can find documentation for the Streaming API at the same location. Since we are not worried about exact dates (but only recent dates), either of the APIs or a combination may be used. We leave it to your discretion as to how you utilize the APIs.

Note: Twitter specifies rate limits on standard API for GET (read) endpoints. Please refer https://developer.twitter.com/en/docs/basics/rate-limits.

4.3 User Timeline

In this project, you are required to retrieve most recent tweets from the timeline of influential personalities and you can do that by using Twitter's REST APIs. However, there is strict rate limiting on the number of tweets you can retrieve from the timeline. If these limits are not followed, you may end up getting your account suspended, therefore, we encourage you to please refer their API reference here : https://developer.twitter.com/en/docs/tweets/timelines/api-reference/get-statuses-user_timeline.html to get information on the rate limits before you start your implementation.

4.4 Twitter Clients

Finally, there is a plethora of Twitter libraries available that you can use. A substantial (though potentially incomplete) list is present here: https://developer.twitter.com/en/docs/developer-utilities/twitter-libraries.

You are welcome to use any library based on your comfort level with the library and/or the language used.

5. Indexing

Before we describe the indexing process, we introduce some terminology.

5.1 Solr terminology

- Solr indexes every **document** subject to an underlying **schema**.
- A schema, much akin to a database schema, defines how a document must be interpreted.
- Every document is just a collection of **fields**.
- Each field has an assigned primitive (data) **type** int, long, String, etc.
- Every field undergoes one of three possible operations : *analysis*, *index* or *query*
- The analysis defines how the field is broken down into tokens, which tokens are retained and which ones are dropped, how tokens are transformed, etc.
- Both indexing and querying at a low level are determined by how the field is analyzed.

Thus, the crucial element is configuring the schema to correctly index the collected tweets as per the project requirements. Every field is mapped to a type and each type is bound to a specific tokenizer, analyzer and filters. The schema.xml is responsible for defining the full schema including all fields, their types and analyzing, indexing directives.

Although a full description of each analyzer, tokenizer and filter is out of the scope of this document, a great starting point is at the following page : https://cwiki.apache.org/confluence/display/SOLR/AnalyzersTokenizersTokenFilters where you can find tips and tricks for all important elements you might want to use for this project. You are encouraged to start either in a schemaless mode or start with the default schema, experiment with different filters and work your way from there.

5.2 Indexing strategies

This is the part where students need to figure out the appropriate way to index their collected tweets. Overall, there are two overarching strategies that you must consider:

- Using out-of-the-box components and configuring them correctly. For example, the *StopFilter* can be used to filter out stopwords as specified by a file listed in the schema. Thus, at the very minimum, you would be required to find language specific stopword lists and configure the filters for corresponding type fields to omit these stopwords.
- Preprocessing tweets before indexing to extract the needed fields. For example, you could preprocess the tweets to introduce new fields in the json response as per project requirement. Here again, it is left to your choice of programming language and/or libraries to perform this task.

Solr supports a variety of data formats for importing data (xml, json, csv, etc). You would thus need to transform your queried tweets into one of the supported formats and POST this data to Solr to index.

6. Project requirements

We now describe the actual project. As mentioned before, the main purpose of this project is to index a reasonable volume of tweets and perform rudimentary data analysis on the collected data. This year's project will be focused on the reactions and opinions of people to the COVID-19 pandemic, and the available vaccines. The goal of this project is to perform social media mining to retrieve and index tweets from government officials and the general population across different geographies. Data collected and indexed in this project will be used for downstream tasks and analysis in Project 4.

In this project, we are specifically interested in tweets from the following types of persons of interest (POI):

- Official government health agency of the country
- Politicians of the country

And the following countries:

- USA
- India
- Mexico

Your dataset should be multilingual and contain following languages

- English
- Hindi
- Spanish

Collect tweets from at least **5 POIs per country**, where

1. 1 of the POIs is the official government health agency of the country
2. Rest of the POIs should be from the current ruling or opposition party, where 1 of them should be the president/prime minister of the country (who has affiliations to the ruling party).

In total, you will have tweets from at least 15 POIs. Make sure you choose POIs who have significant user engagement on their tweets.

Task 0 (Phase 1) : Figure out a set of at least 30 keywords and hashtags for collecting Covid-19 vaccine and general Covid-19 related tweets.

Task 1 (Phase 2) : Figure out the required set of language filters, person of interest and combinations thereof to crawl and index tweets, subject to the following requirements:

1. At least 50,000 tweets **in total** with not more than 15% being retweets.
    a. At least 500 tweets per POI (7,500 total for all 15 POIs). Out of the 500 tweets per POI, at least 50 should be related to Covid-19 and the Covid-19 vaccines. Note that Twitter allows max 3200 recent tweets to be extracted from a person's account.
    b. At least 32,500 tweets from the general population, and related to the Covid-19 vaccines.
    c. The rest 10,000 tweets must be replies to either point b or point a, constrained by:
        i. At least 1,500 tweets from b should have 1 reply

ii. At least 10 replies for a minimum of 300 Covid-19 related tweets by the POIs (Minimum 3,000 replies in total for all POI).

2. At a high level, there should be
    a. At least 5,000 tweets per language i.e, English, Hindi and Spanish.
    b. At least 5,000 tweets per country.

Note that the above are the minimum requirements. You are free to crawl tweets in other languages outside this list (or crawl tweets without a language filter for example) as long as the above requirements are met. Further, this data would be validated against your Solr index. Thus, based on how you setup your indexing, you may lose some tweets and/or have duplicates that may reduce your indexed volumes. Hence, it is encouraged that you accommodate some buffer during the crawling stage.

Once you have collected your tweets, you would be required to index them in a Solr instance. You would need to tweak your indexing to adhere to two distinct sets of requirements language specific and Twitter specific as described below.

Task 2 (Phase 2) : Index the collected tweets subject to the following requirements:

1. Person of Interest: Name and Ids of at least 15 persons of interest
2. Country: one amongst USA, India and Mexico
3. One copy of the tweet text that retains all content (see below) irrespective of the language. **This field should be set as the default field while searching.**
4. Language of the tweet (as identified by Twitter) and a language specific copy of the tweet text that removes all stopwords (language specific), punctuation, emoticons, emojis, kaomojis, hashtags, mentions, URLs and other Twitter discourse tokens. Thus, you would have at least four separate fields that index the tweet text, the general field above plus three for each language. **For any given tweet, only two of the four fields would have a value.**
5. Separate fields that index: hashtags, mentions, URLs, tweet creation date, emoticons (emoticons + emojis+ kaomojis)

Note: After Phase 1, we will collate the list of submitted keywords, and release a set of final top keywords, which needs to be used for data collection in Phase 2. Use of any other keywords in Phase 2 will not be permitted.

7. Submitting your project and Naming Conventions

The submission of this project will happen in 2 phases.

a) **Phase 1: Submission of Keywords:** You are required to submit a .pickle file containing a dictionary of Covid-19 related keywords/hashtags, and a list of Covid-19 vaccine related keywords/hashtags, which you are going to use for collecting the tweets. There should be at least 30 keywords in total (covid & covid vaccine keywords combined). The file should be named *project1_keywords.pickle,* and must be submitted using cse_submit script.

Example file contents:

```
{
"covid": ["covid", "coronavirus", .....],
"vaccine": ["covid vaccine", "pfizer", "vaccine mandate", ...]
}
```

The due date for this phase is September 8, 2021, 23:59 EST.

b) **Phase 2: Final Submission:** You are required to submit a .json file containing the IP address of your EC2 instance, and the final set of keywords used by you. The file should be named *project1_index_details.json.*

Example file contents:

```
{
        "ip": "172.222.111.000",
        "port": "8983",
        "core": "IRF21P1",
        "covid_keywords": ["coronavirus", ...],
        "vaccine_keywords": ["covid vaccine", ...]
}
```

The due date for this phase is September 19, 2021, 23:59 EST.

In order to make a submission, you need to use the cse_submit script, the details of which will be made available soon.

Convention of field names in the index: The required field names are as given below:

1. poi_name : Screen name of one of the 15 persons of interest, should not be set to any value for non POIs. Field type string.
2. poi_id : User Id of one of the 15 persons of interest, should not be set to any value for non POIs. Field type plong.
3. verified: Boolean value
4. country : One of the 3 countries. Field type string.
5. id: The tweet id of the tweet. Field type plong.
6. replied_to_tweet_id : In case of a reply tweet, the tweet id to which the reply is made. Should not be used otherwise. Field type plong.
7. replied_to_user_id : In case of a reply tweet, the user id to which the reply is made. Should not be used otherwise. Field type plong.
8. reply_text : Text of the reply to a particular tweet, if replied_to_tweet_id is not null. Should not be used otherwise. Field type text_general.
9. tweet_text : Default field. Field type text_general.
10. tweet_lang : Language of the tweet from Twitter as a two letter code. Field type string.
11. text_xx : For language specific fields where xx is at least one amongst en (English), hi (Hindi) and es (Spanish). Select appropriate language specific field type in Solr, in order to leverage pre-defined functionalities for that specific language.
12. hashtags : if there are any hashtags within the tweet text. Should not be used otherwise. Multivalued strings field.
13. mentions : if there are any mentions within the tweet text. Should not be used otherwise. Multivalued strings field.
14. tweet_urls : if there are any urls within the tweet text. Should not be used otherwise. Multivalued strings field.
15. tweet_emoticons : if there are any emoticons within the tweet text. Should not be used otherwise. Multivalued, strings field, or a custom defined field in case you are pre-processing the tweet using Solr.
16. tweet_date : Tweet creation date rounded to nearest hour and in GMT. Field type pdate.
17. geolocation: This is an optional field, and will contain the exact latitude and longitude of the user if available.

Note that, “string” field type means it’s a single-valued field. On the other hand, “strings” means array of string values.

8. Grading

This project is worth a total of 15 points, these are distributed as follows:

Task	Criterion	Description	Points
Task 0 (Phase 1)	Identifying Keywords	Submit at least 30 Covid-19 and vaccine related keywords	2
Task 1 (Phase 2)	Tweet Volumes	Validate at least 50,000 tweets	1
	Language Volumes	Validate at least 5,000 tweets for en, hi, es	1
	Country Volumes	Validate at least 5,000 tweets per country	1
	Person of Interest criteria	Validate: ● At least 15 persons of interest chosen ● At least 500 tweets retrieved from person of interest’s timeline ● At least 50 tweets of the 500 tweets are related to Covid-19.	2 (0.5+0.5 +1)
	Retweet Counts	Validate at most 15% retweets	1
	Non POI tweets	At least 32,500 tweets (and replies) from non POI, related to Covid-19 vaccines.	1
	Reply Counts	Validate ● At least 10,000 replies in total ● At least 10 reply from a minimum of 300 Covid-19 related tweets by the POIs (Minimum 3,000 replies in total for all POI)	4 (1 + 2 + 1)

		• At least 1,500 tweets from non POIs should have a minimum of 1 reply.	
Task 2 (Phase 2)	Sanity	Validate Solr instance runs + can run some queries	1
	Schema Validation	All fields are named as required, contain values as required, etc.	1

Not earlier than a week before the final submissions, an automatic grader will be made available as an API, which can be used to sanity check your submission. The details of the API will be shared soon.

9. FAQs

Q: If a tweet contains RT @ and the field value 'retweeted' is false, then is it considered a fresh tweet or retweet?
A: We would rely on the fields we have asked you to index to test for this. Since, retweeted isn't one of them, we would only look at the raw text.

Q: Is the max 15% retweets restriction also applicable to language specific tweets?
A: Applies at a global level.