Project: Capstone Project 1: Data Wrangling

Experience Coffee Searcher

Extraction meaning from text faces the challenge to deal with text data, the most abundant source of unstructured data and to retrieve the essence of a paragraph in a few couples of words. How to turn flowing text going through Yelp and coffee blogs in features vector for machine learning models?

1. Data Extraction

The following diagram (Fig. 1) shows the dynamic between data in the extraction process.

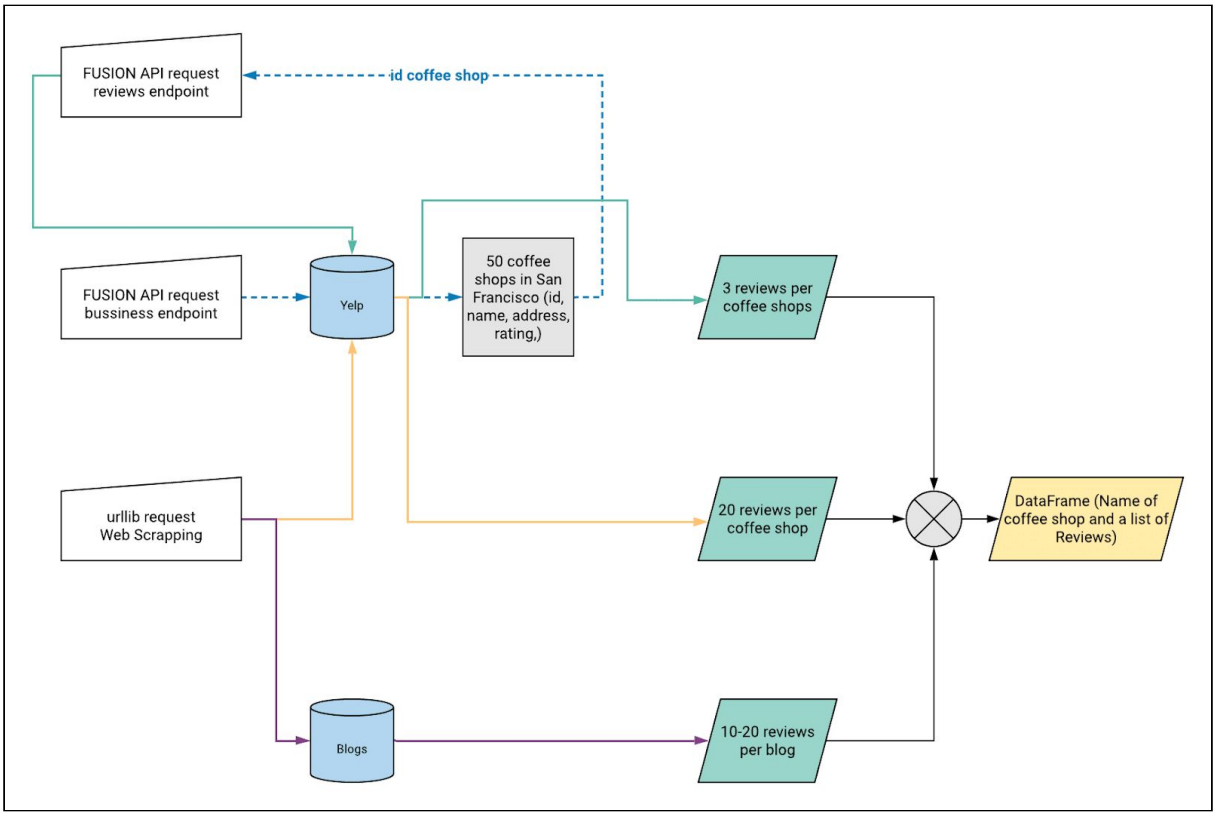

Fig. 1: Data flow from extraction to building of DataFrame

Yelp Reviews Extraction

The first step is to send a request to Fusion (Yelp API) using the endpoint /businesses/search to get names/id of coffee businesses in San Francisco. For this purpose, request parameters are San Francisco (as location), coffee (as the search term) and the maximum limit of data. The output is information about 50 businesses including id, name, address, opening hours, rating, number of reviews. Id's are the inputs to the second request to the endpoint /businesses/{id}/reviews to obtain 3

reviews per each coffee shop. This narrow number of review, encourage us to extract more reviews from Yelp using scrapping.

The second step is web scrapping with Beautiful soup library, searching Coffee Shops in San Francisco through the Yelp platform. Each page of results has 30 coffee shops. We will start working with 2 pages (60 businesses, including some names gotten with the API request to add more reviews and other news results). Yelp contains almost 1000 results of coffee shops in San Francisco. If it is necessary to improve the variety of data, it will include more search pages. The focus in this part is about inspecting the Yelp page and understand the basic syntax of HTML to find the HTML document, titles headings, paragraphs and hyperlinks to open the section that each coffee shops has in Yelp with it owns reviews. At this way, we get 20 reviews per shop. In this section, all HTML tags are removed to extract names and reviews.

Blogs Reviews Extraction

Process of data consists of web scrapping in blogs about coffee shops in San Francisco written since 2017. Additionally, for selecting blogs, locations were checked (some blogs include coffee shops from Oakland or Berkeley). In general, these kind of reviews are brief, but you can find a lot of useful words in a couple of phases. In selected blogs, 2 are related to better coffee shops (authors write about drinks and space features) and 2 are about better places for working or studying. Blogs chosen are listed below (Table 1).

Title of coffee blog	Authors	URL
"28 of San Francisco's Essential Coffee Shops: Hot spots for your morning cup"	Ellen Fort and Caleb Pershan	https://sf.eater.com/maps/best-coffee-shops-san-francisco-oakland-berkeley
"The Best Coffee Shop in Every SF Neighborhood"	Margaux Poupard and Urmila Ramakrishnan	https://www.thrillist.com/drink/san-francisco/best-coffee-shops-san-francisco
"The Best SF Coffee Shops For Getting Work Done"	Taylor Abrams, Frida Garza, and Will Kamensky	https://www.theinfatuation.com/san-francisco/guides/the-best-sf-coffee-shops-for-getting-work-done
"16 San Francisco Coffee Shops with Free Wifi and Legit Food"	Daisy Barringer, Ellen Fort, and Caleb Pershan	https://sf.eater.com/maps/best-coffee-shop-wifi-cafe-food-san-francisco

Table 1: Coffee Blogs chosen to the text-processing

At this point, we can build a DataFrame with names and descriptions of coffee shops using pandas. Each description is a *corpus* (collection of text documents) and

it contains different reviews or *documents* (written from experts and customers or only one of that). DataFrame looks like Fig.2

	Name	Description
0	Four Barrel Coffee	[In love with the look and vibes here at Four ...
1	Réveille Coffee Co.	[Love the vibe in this cozy little space! It's...
2	The Mill	[Pizza is served after 6!!!!!!!!\n\nFor those...
3	Rise & Grind Coffee and Tea	[A great place to sit and work.\nPlenty of out...
4	Home	[Visiting friends in SF, my husband was cravin...
5	Native Twins Coffee	[This place to caters to all types of eaters. ...
6	Home	[So cute and such a fun place to study or chat...
7	Blue Danube Coffee House	[To who ever made my Greek yogurt today:\n\nYo...
8	Snowbird Coffee	[Incredible staff with a refined menu. \n\nThe...
9	Andytown Coffee Roasters	[Personally, I am a huge fan of coffee. Andyto...
10	Wrecking Ball Coffee Roasters	[This cafe is instagram worthy! The wallpaper ...

Fig. 2: Sample of Coffe shops Dataframe

2. Text Pre-processing

Following diagram shows the Natural Language Processing pipeline used as pre-processing. Remotion of HTML tags it was applied previously with BeautifulSoup library. In this section, we describe briefly the rest of wrangling steps.

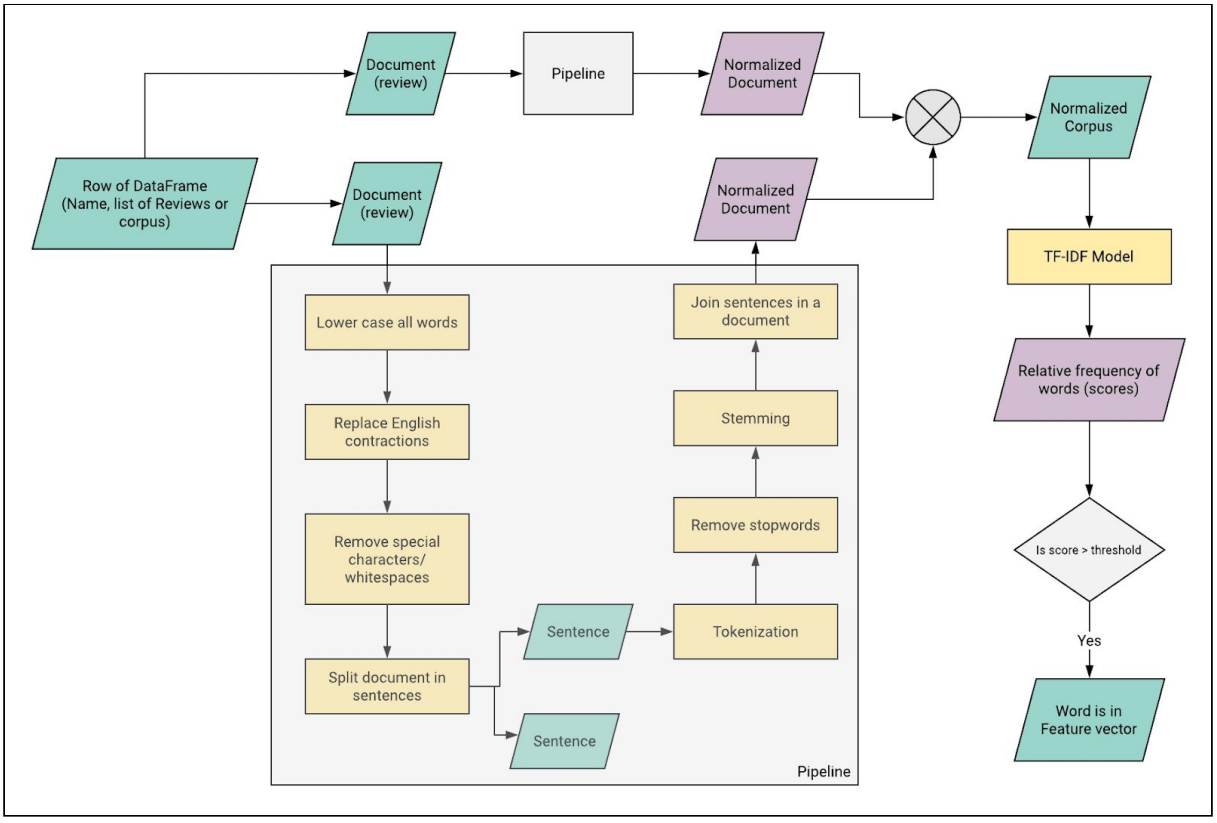

Fig. 3: NLP pipeline of text data

A dictionary to expand English contractions is utilized to help with text standardization process. After that, all special characters/whitespace/ are removed and each document (paragraph) is split into sentences. Next step is stemming, looking for the base of words used to delete unnecessaries prefixes and suffixes. Then, tokenization of sentences is applied to remove stopwords (like prepositions, articles and all words that appear frequently in the text but they don't have significance). Finally, the sentences are joined as a paragraph again.

Doing the above pipeline with all reviews per coffee shops, it is possible building a normalized corpus that we use as the input to TF-IDF Model, that assigns a normalized frequency (score) to each word. This score is directly proportional to the frequency of the term in a document but inversely proportional to the frequency of the term in the whole corpus. It prevents assigning more importance to repetitive words that overshadow others.