Assignment - 1

Lakshya Gupta and Palemkota Maithresh

May 5, 2021

We believe that a **complete address** should contain

- Name of the person (if present) and his/her designation.
- Name of the company (if present)
- House number, street, city, state, country, pin codes
- Landmarks (if present)
- Email ID, Phone numbers and fax numbers.

1 Best Saving

- For 0.html, total space gained is 97.68 %
- For 1.html, total space gained is 99.84 %
- For 2.html, total space gained is 93.26 %
- For 3.html, total space gained is 99.11 %
- For 4.html, total space gained is 98.67 %

Total gain for 5 files is 98.44%.

Using the criteria used to define an address, we estimate that the theoretical maximum savings for the given inputs is around 98.60%.

2 Data Processing Steps

We wrote several functions, each of which does a specific job.

- Firstly, the function $'strip'$ removes any unnecessary white spaces and new lines in the data.

- We make different files containing city names, state names, head keywords(potential words that may exists in the beginning of the address), additional keywords (potential words that may exist within the body of the address), last keywords(potential words that may be present at the end of the address). The keyword files are created using the python script *'keywords_new.ipynb.*

- We, then identify the positions of the pincodes, phone numbers and fax numbers using the function *'pincode'*. This function is called inside an another function *'parse'*

- After identifying the positions of pincodes, phone numbers and fax numbers, we check if there exists any keywords nearby these positions. If we don't find enough keywords, that position is not considered a part of an address.

- If a pincode, phone number and fax number is identified to be a part of the address, then we check for the positions of the head keywords, add keywords and last keywords in nearby locations. Depending on the positions of these keywords, an appropriate range of the address is chosen. This is done by the function *'range_find'*.

- To catch addresses that do not contain pincode/mobile number/fax number, we try to find at least three keywords in vicinity from distinct categories from five categories(head, add, last, state, city). If more than 3 keywords are present in a range of 200 characters, then it is considered as an address. This is done by the *'no_pincode'* function. An example of an address that this function can catch is : *Kokilaben Dhirubhai Ambani Hospital, Rao Saheb Achutrao Patwardhan Marg, Four Bunglows-Andheri West, Mumbai, Near Kamdhenu Departmental Store.* This address is identified using the keywords *'Marg', 'West', 'Mumbai' and 'Near'.*

- After identifying the positions where an address can be, we feed those positions into the function called as *trim*. This functions extracts the addresses from the data using the identified positions.

3 Prospects

- We would like to check for for more html (in hundreds or more) files and see if the parameters we choose are the best ones.

- We would like to reduce the redundancy of the code, as it has been made from continuous progress without a fixed structure from the beginning.

- Although our code can identify the address without pincode/mobile numbers /fax numbers, it is not very efficient. By identifying more keywords, we think it is possible to make it better.

4 Difficulty

This task was neither too difficult nor easy, as basic idea to what we had to do was clear from the very beginning and we had to look for the ways to achieve that. But to achieve the ideal result is very difficult so we had to think what ways can lead us near it. Once we have a basic direction to begin with (pincode in our case), things that can be done next follows through.