NLP assignment 5: Report

Debraj C, Sanath K K, Siddarth R

1 Introduction

In this assignment, we built a city rating system. We collected data for 10 cities from many different online sources. The 10 cities are: Ahmedabad, Bangalore, Bhopal, Chandigarh, Chennai, Delhi, Hyderabad, Kolkata, Mumbai, and Pune.

2 Data collection

For every city we collected information on: Safety, Accidents, Health, CCTV, Hotels and Food, Public toilets, Natural hazard and emergency services, Public transport, Power supply, and Water availability. Where safety includes information on crime, women safety, and law enforcement.

Sometimes, data is too unorganized (making it hard to use). Sometimes, data is too organized with complicated data collection rules (making it hard to cheaply collect more of such data). Our data collection rules strikes the right balance between simplicity and usability of the data. Our data collection rules for this project:

- The data is a text document and is written in sections.
- Each section is about one feature, with the feature name at the top of the section.
- You leave a line after the feature title and leave 3 lines between sections.

Note: Our code is robust to having a few more extra line between sections. Errors of this form are expected and will not affect the code. The code is also robust to missing features (or even extra features, which the code would simply ignore).

3 Code

Consider a city and the relevant text document. We extract the relevant sections from the text document, and use one of NLTK's sentiment analyzer on each of these sentences. The sentiment analyzer gives a score between -1 and 1. Hence, we get a sentiment score for each feature.

We now build a weight vector $w \in [0,1]^{10}$, a weight for each of the 10 features, such that $\sum_i w_i = 1$. Let $v \in [-1,1]^10$ denote the vector of sentiment scores. Then the dot product $v \cdot w \in [-1,1]$, and hence $e^{v \cdot w} \in [1/e, e]$. With an affine transformation, we get this score to lie in $[0,5]$ and round the value to get a 5-star rating.

We randomly selected 5 cities and found 5-star ratings for these cities online. Since the cities are fixed, we have a fixed set of 5 sentiment score vectors (one for each city). These sentiment score vectors can be treated as constant vectors. The weight vector is now generated by optimizing w so that, the 5-star rating generated by our code is as close as possible to the rating found online (for these 5 cities). We use mean square error loss function as our measure of closeness.