Assignment 1-Preprocessing

Promothesh Chatterjee*

*Copyright© 2024 by Promothesh Chatterjee. All rights reserved. No part of this note may be reproduced, distributed, or transmitted in any form or by any means, including photocopying, recording, or other electronic or mechanical methods, without the prior written permission of the author.

Review Dataset

This corpus consists of truthful and deceptive hotel reviews of 20 Chicago hotels. This corpus contains:

400 truthful positive reviews from TripAdvisor 400 deceptive positive reviews from Mechanical Turk 400 truthful negative reviews from Expedia, Hotels.com, Orbitz, Priceline, TripAdvisor and Yelp 400 deceptive negative reviews from Mechanical Turk

Hotels included in this dataset affinia: Affinia Chicago (now MileNorth, A Chicago Hotel) allegro: Hotel Allegro Chicago - a Kimpton Hotel amalfi: Amalfi Hotel Chicago ambassador: Ambassador East Hotel (now PUBLIC Chicago) conrad: Conrad Chicago fairmont: Fairmont Chicago Millennium Park hardrock: Hard Rock Hotel Chicago hilton: Hilton Chicago homewood: Homewood Suites by Hilton Chicago Downtown hyatt: Hyatt Regency Chicago intercontinental: InterContinental Chicago james: James Chicago knickerbocker: Millennium Knickerbocker Hotel Chicago monaco: Hotel Monaco Chicago - a Kimpton Hotel omni: Omni Chicago Hotel palmer: The Palmer House Hilton sheraton: Sheraton Chicago Hotel and Towers sofitel: Sofitel Chicago Water Tower swissotel: Swissotel Chicago talbott: The Talbott Hotel

Assignment

1) Google how to randomly sample 500 observations from the dataset. Then select only the truthful reviews and check if the length of the reviews differ across positive and negative reviews. Do the same for deceptive reviews.

2) Tokenize the 'text' variable, removing punctuations, symbols, hyphens and numbers, and convert it into a Document Term Matrix (hint: use library quanteda).

3) 3. Check the dimensions and sparsity of the DFM, view the first five rows and columns.

4) Repeat the same analysis as 1) and 2) but now remove stopwords, stem the words, change to lower case. Do you see a difference in dimensions and sparsity?

5) Create a bar plot for most frequent words after preprocessing the original file with library tidytext

6) Perform a tokens-per-document analysis on the entire DTM created in step 4 and plot a histogram

7) Create a comparative wordcloud for the groups 'truthful' vs 'deceptive'

8) Create both unigrams and bigrams on the original dataset and compare the number of dimensions and sparsity across the datasets.

9) Create a wordcloud for the bigrams

10) Create a customized bigram wordcloud for the most frequently occurring word in the previous step.

11) Preprocess the original dataset and perform a TFIDF weighting on it, then create a tokens per document plot.

12) Perform collocation analysis on a slice of the dataset