A Fast and Easy Way to Predict Words

README

Chen Lianghe

15 February 2020

Background & Algorithm

Natural Language Processing (NLP):

- The study of interactions between computers and human (natural) languages.
- Utilizes computer programming to process and analyze large amounts of natural language data.

N-Gram:

- A contiguous sequence of n items from a given sample of text or speech.
- Built our n-grams from a corpus named HC Corpora, which contains text from news, blogs and twitter.
- Built a text mining predictive model to determine the most likely next word from a user-specified text input.

Our Predictive Model:

- Based on the "Stupid Backoff" method described by Brants et. al, 2007.
- First, it checks for the highest order n-gram, a quadgram (n=4).
- If a match is not found, it proceeds to check for a trigram (n=3).
- Subsequently, it checks for a bigram (n=2) if there are still no matches.

Strengths of Our Application

1. Fast and returns the 2 most likely predictions within a second.
2. User-friendly app interface with clear instructions.
3. Accurate predictions that are logical and useful.

Areas of Improvement

1. Machine Learning based on User Input:

- N-grams can be updated by user input to provide predictions that are more accurate and specific for the user.

2. Using Higher Order N-Grams:

- With more data, our algorithm can extend to higher order n-grams for better predictions.

Resources

1. Our Presentation is available at RPubs.
2. Our Shiny App is available at shinyapps.io by RStudio.
3. Our Source Codes are available at GitHub.

Thank You Very Much!