Matthias Eyassu

CS 499

Project Baseline Report

Overview

The objective of this project is to implement a language model with the capacity to generate reams of science fiction in the creative and humorous style of authors like Adams and Colfer. In this baseline submission, a single layer LSTM along with a linear layer is used to learn a trigram model.

Data and Preprocessing

Fortunately, classic titles such as *Hitchiker's Guide to the Galaxy, The Restauarant at the End of the Universe, Life, the Universe, and Everything, So Long, and Thanks for all the Fish* are easily accessible in pdf form online. They are stored under `/data`. For the purposes of implementing a simple baseline model, I restricted myself to the first thirty percent of Adam's most famous novel, *Hitchiker's Guide to the Galaxy* which amounts to around 80,000 words. After reading the raw data from the pdf with `tika`, the program preprocesses the corpus by removing various symbols (html tags, punctuation, numbers, newlines, dashes), splitting hyphenated words, expanding contractions, lowercasing, tokenizing the text into a list of words, and converting the word list into a set of trigrams. These operations are mostly performed with `re` and `nltk`. I decided not to remove stopwords or apply stemming since I thought this might hurt the network's ability to write coherently. Although the program can train for any number of epochs and write an arbitrary number of words, I ran it for 20 epochs and set its output length to 1000 words to understand its bare-bones performance. The story it wrote is stored at `/output/stories/hgg_spinoff.txt`.

Evaluation Metric

Programs solving natural language problems such as part of speech tagging, translation, parsing, and question answering can be evaluated using simple metrics like accuracy but its significantly harder to assess the quality of text generation models without resorting to manual inspection and reading. Automatic methods have been developed and I definitely plan on using them in the future but for this baseline submission, I decided to rate the output with the guidance of the following scoring scheme:

Output Score: Syntax (1-5): 1 Semantics (1-5): 0.5 Novelty (1-5): 2

The baseline method performs poorly. The output is not pure gibberish but there is no plot development in any region of the text; it will be fun to try and push the model to its limit.

Resources:

Kaggle: Beginner's Guide to Text Generation with PyTorch (https://www.kaggle.com/ab971631/beginners-guide-to-text-generation-pytorch)

Assignment 3 Sample Code