LING 575K HW1

Due 11PM on Apr 5, 2021

1 Setting Up Environment on Patas [10 pts]

Install Anaconda. This is a necessary component for running the code for most assignments in this course. We have setup a conda environment for the assignment, but you will need to install anaconda in order to use that environment. These are "free points". From your home directory, please execute the following steps:

1. `wget https://repo.anaconda.com/archive/Anaconda3-2020.11-Linux-x86_64.sh`
2. `sh Anaconda3-2020.11-Linux-x86_64.sh`

The first two lines of `run_hw9.sh` show how to now activate the environment that we have supplied with all of the necessary libraries.

2 Implementing a Vocabulary Class [30 pts]

A `Vocabulary` object provides the basic interface between raw text (strings) and integer IDs, which the models we will build require. In `/dropbox/20-21/575k/hw1` you will find:

- vocabulary.py: main file to edit
- main.py: running script, also to edit
- run_hw1.sh: main shell script, add your commands here
- toy-vocab.txt: output of running `python main.py --text_file /dropbox/20-21/575k/data/toy-reviews.tx --output_file toy-vocab.txt`

All places for you to edit are marked by '# TODO:' comments.

Q1: populate index_to_token Near the top of `Vocabulary.__init__`, you must use a Counter object to populate the list `self.index_to_token`. Some notes:

- You should iterate in the order that the Counter iterates.
- You should not include any tokens that occure below `min_freq` times.
- You should only add the first `max_size` tokens

Q2: populate token_to_index At the end of `Vocabulary.__init__`, you need to populate a token-to-string dictionary. Keys are tokens, and values are their integer indices.

Q3: implement two methods Two short methods require implementation: `tokens_to_indices` and `indices_to_tokens`.

Q4: implement main.py main.py is a simple script: it takes in a text input file, and an output file name. You should define a Vocabulary object based on that text file, and then save/write it to the output file. The Vocabulary class has methods to help with both of those steps.

Q5: build two vocabularies Finally, execute the following two commands (and add them to `run_hw1.sh`):

- `python main.py --text_file /dropbox/20-21/575k/data/sst/train-reviews.txt --output_file train_vocab_base.txt`
- `python main.py --text_file /dropbox/20-21/575k/data/sst/train-reviews.txt --output_file train_vocab_freq5.txt --min_freq 5`

These will produce two output files that should also be included in your tar.gz (see below).

3 Data Statement for Stanford Sentiment Treebank [35 pts]

For natural language processing applications, data plays a crucial role, since it largely shapes the resulting models and systems that are used in deployment. Emily Bender and Batya Friedman have recently been developing and advocating for the practice of "data statements": explicit documentation of the nature and origins of datasets used in NLP. For more information, please consult their paper:

- "Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science", *Transactions of the ACL*

For the first half of this course, we will make heavy use of the Stanford Sentiment Treebank. Before beginning to use this data in modeling, you will attempt to write a data statement for this dataset to the best of your ability given the documentation available. Afterwards, there will be space to reflect on what was missing. Please see:

- The paper presenting the data (§3 in particular): `https://www.aclweb.org/anthology/D13-1170/`
- The website for the dataset: `https://nlp.stanford.edu/sentiment/index.html`

Please answer, in a couple of sentences, the following questions. [Each question is 5 points.]

Q1: Curation Rationale Which texts were included and what were the goals in selecting texts?

Q2: Language Variety From what language variety are the texts? You may provide a prose description and/or a BCP47 code.

Q3: Speaker Demographics Who were the producers of the texts? Information may include (if available): age, gender, native language, socioeconomic status, number of speakers.

Q4: Annotator Demographics Who were the annotators of the data? Information may include (if available): age, gender, native language, socioeconomic status, number of annotators.

Q5: Speech Situation What were the conditions of text production? Details may include whether it was written or spoken, whether it was spontaneous or not, and who the intended audience was.

Q6: Text Characteristics What are the genre and topic of the texts?

Q7: Reflections Which of these questions were hard to answer with the information provided about the dataset? Why might it be helpful to have that information more explicitly documented?

Submission Instructions

In your submission, include the following:

- readme.(txt|pdf) that includes your answers to §3 Q1-Q7.
- `hw1.tar.gz` containing:
  - vocabulary.py
  - main.py
  - run_hw1.sh
  - train_vocab_base.txt
  - train_vocab_freq5.txt