Learning-Word-Embeddings

Setup & Running

1. Clone this repository and `cd` into it.

```
git clone git@github.com:mallika2011/Learning-Word-Embeddings.git
cd Learning-Word-Embeddings
```

2. Create a virtual env and install requirements

```
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

3. There are 4 tasks that can be done : create vocabulary, train svd, train cbow, tokenize corpus

- To create the vocabulary run the following. This will create the files `word2count.json`, `word2ind.json`, `ind2word.json`, and `vocabulary.txt` in the `vocab_files` directory

```
mkdir vocab_files
python3 run.py Electronics_5.json 0
```

- To create the co-occurrence matrix and perform SVD run:

```
python3 run.py Electronics_5.json 1
```

- To train CBOW embeddings run the following. This by default runs CBOW using nn_based method. If the enc-dec method is preferred, comment L:16 and uncomment L:17 in `run.py`:

```
python3 run.py Electronics_5.json 2
```

- To tokenize and store the tokenized corpus run the following. Here the sampling boolean can be altered directly within the script.

```
python3 run.py Electronics_5.json 3
```

Model and Large File Links:

All vocabulary files and models for the CBOW method are present in the repo. The link to **co-occurrence matrix models** and the **tokenized_corpus** files (with and without sampling) can be found here:

All Trained Models

All Vocab Files

All Embedding Files

- Co-occurrence Matrix Models *with* Sampling
- Co-occurrence Matrix Models *without* Sampling
- Tokenized Corpus *with* sampling
- Tokenized Corpus *without* sampling

These are required to be able to directly run the model scripts (for CBOW) without having to create the tokenized corpus.

Corpus Statistics:

- Total Number of reviews in the corpus = 1.6M
- Reviews used to train models = 1M
- Total size of the vocabulary created = 67829 unique words

References:

SVD & Co-occ:

- https://medium.com/analytics-vidhya/co-occurrence-matrix-singular-value-decomposition-svd-31b3d3deb305
- Sparse matrices: https://www.geeksforgeeks.org/how-to-create-a-sparse-matrix-in-python/

CBOW:

- Stanford Class Notes: https://cs224d.stanford.edu/lecture_notes/notes1.pdf

TSNE & Dim Reduction:

- https://towardsdatascience.com/how-to-tune-hyperparameters-of-tsne-7c0596a18868
- https://distill.pub/2016/misread-tsne/

Misc:

- Text Pre-processing
- Sampling: https://cs.stackexchange.com/questions/95266/subsampling-of-frequent-words-in-word2vec
- Original Paper: https://papers.nips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf