Ling/CSC 439/539: Assignment #1 (75 pts)

Due by 11:59 P.M., August 27
(upload all materials to D2L)

Requirements for the submission

You must submit code and a written report for this assignment. The code and report must follow these requirements:

1. Each programming question must be answered through code that is **executed with a single command** in the terminal. Submissions that must be run through an IDE (e.g., Eclipse, IntelliJ, etc.) are not accepted.
2. Similarly, if your code requires compilation (e.g., it is written in Java), **a single command line for compiling the code must be provided**.
3. If your code requires certain dependencies (e.g., specific libraries, version of the Python language), these have to be clearly stated with instructions for installation.
4. Your report must include **clear instructions for all the above issues**.
5. The code for this assignment **cannot use other NLP libraries** such as NLTK, but may use libraries for linear algebra such as numpy.

Points will be taken off if the above requirements are not met. Additionally, your code must **compile** (if required by the programming language), **run**, and **produce the correct output**. Points will be taken off in any of these issues are violated.

Problem 1 (35 points)

Download the `brown_sample.txt` corpus from "Assignment 1" D2L folder. The corpus is formatted as a sequence of tokens separated by white space. For example, the text:

```
The/at Fulton/np-tl County/nn-tl Grand/jj-tl Jury/nn-tl said/vbd Friday/nr
```

```
an/at investigation/nn of/in Atlanta's/np$ recent/jj primary/nn election/nn
produced/vbd ``/`` no/at evidence/nn ''/'' that/cs any/dti irregularities/nns
took/vbd place/nn ./.
```

shows one sentence from this corpus. Each token contains a word and its part-of-speech (POS) tag separated by slash. For example, the token "investigation/nn" contains the word "investigation" and its POS tag "nn", which indicates that it is a noun ("nn*" POS tags indicate nouns, "vb*" indicate verbs, "jj*" indicate adjectives, etc.).

Write code that answers the following questions:

1. What are the top 10 most frequent words in this file (independent of POS tags)?
2. What are the top 10 most frequent POS tags?
3. What the the top 10 most frequent word-POS tag pairs?

Problem 2 (40 points)

The file `vectors_top3000.txt` in D2L projects the most common English words into a 200-dimensional space. That is, each word is assigned a 200-dimensional vector, called a word embedding vector. These vectors capture the semantics of language such that the vectors for related words (e.g., "car" and "bus") will be close to each other, whereas vectors for unrelated words will be far apart. (We will discuss later in class how to generate these vectors; for now, let's just use them.)

In particular the format of the file is the following: each line corresponds to a single word. Each line contains 201 tokens separated by white space, where: the first token is the word itself, and the remaining 200 tokens each correspond to coordinates of the corresponding embedding vector.

Please implement the following:

1. What are the top 10 most similar words to "home" in this 200-dimensional space? Compute similarity using the dot product formula, defined as $dot(\mathbf{x}, \mathbf{y}) = \sum_{i=0 \texttt{ to } 199} x_i \times y_i$, where $\mathbf{x}$ and $\mathbf{y}$ are two vectors, and x_i is coordinate i of the vector $\mathbf{x}$.
2. What are the top 10 most dissimilar words to "home"?