Language Detection

We will be writing a program that, given a body of text, determines what language the text is written in.

Installation

We are going to use python's Natural Language ToolKit (NLTK) so we can get access to some useful functions and data for language detection. Follow the instructions here to install NLTK depending on your operating system.

You'll need to `import nltk` in your program.

Algorithm

Stopwords are very common words in a language, generally words that are grammatically necessary but don't provide a ton of context about what's going on in the sentence. In English, common stopwords are "the", "and", "too", etc. Because they are so ubiquitous, you might be able to look at a body of text and determine what language it's written in based on its stopwords. We'll use this basic algorithm with stopwords to determine the language of the input text: if the most popular words in our input text are some of the most popular words in a language (i.e. a subset of that language's stopwords), we say the text is written in that language.

NLTK provides stopwords for Dutch, Finnish, German, Italian, Portuguese, Spanish, Turkish, Danish, English, French, Hungarian, Norwegian, Russian, and Swedish. You can retrieve the stopwords for each language with the `stopwords.words()` function provided by nltk, e.g:

```
stopwords.words('english')
```

will return:

```
[u'i', u'me', u'my', u'myself', u'we', u'our', u'ours', u'ourselves',
 u'you', u'your', u'yours', u'yourself', u'yourselves', u'he', u'him',
 u'his', u'himself', u'she', u'her', u'hers', u'herself', u'it', u'its',
 u'itself', u'they', u'them', u'their', u'theirs', u'themselves', u'what',
 u'which', u'who', u'whom', u'this', u'that', u'these', u'those', u'am',
 u'is', u'are', u'was', u'were', u'be', u'been', u'being', u'have', u'has',
 u'had', u'having', u'do', u'does', u'did', u'doing', u'a', u'an', u'the',
 u'and', u'but', u'if', u'or', u'because', u'as', u'until', u'while', u'of',
 u'at', u'by', u'for', u'with', u'about', u'against', u'between', u'into',
 u'through', u'during', u'before', u'after', u'above', u'below', u'to',
 u'from', u'up', u'down', u'in', u'out', u'on', u'off', u'over', u'under',
 u'again', u'further', u'then', u'once', u'here', u'there',
 u'when',u'where', u'why', u'how', u'all', u'any', u'both', u'each', u'few',
```

```
 u'more', u'most', u'other', u'some', u'such', u'no', u'nor', u'not',
u'only', u'own', u'same', u'so', u'than', u'too', u'very', u's', u't',
        u'can', u'will', u'just', u'don', u'should', u'now']
```

Play with this in the python interpreter to see what the other languages' stopwords are.

We can find the most common words in the input text by splitting the text into *tokens* and computing a *frequency distribution*. For example, in the sentence, "One ring to rule them all, one ring to find them" our tokens are [One, ring, to, rule, them, all, find]. You can write your own tokenizer, or you can use the one provided by nltk:

```
nltk.tokenize.word_tokenize(some text)
```

The frequency distribution over these tokens is just the number of times each token appeared in the original text. You can write your own function to compute the frequency distribution, or you can use the one provided by nltk:

```
nltk.FreqDist(tokens)
```

Then you can use this frequency distribution of the tokens to compare the frequently used words in the input text with the stopwords for each language, and subsequently determine what language the input text is likely written in.