Lab 1

James Lowrey

In this lab two different methods were used to parse Reuter’s articles in order to find “interesting” or meaningful words. The Reuter articles are encoded in a sgm format with tags identifying the topics, places, title, and body of each article. Words are interesting/meaningful if they indicate information about the article they are in. Stop words like “and”, “a”, or “the” do not indicate information as they are very common in the English language. Extremely rare words also do not indicate information, as they can be noise in the text extraction process. Thus different processes were employeed to find words that match this middle of the line criteria.

The first step in this information retrieval task was finding the relevant text in the sgm files. The BeautifulSoup4 Python library was utilized to turn the sgm files into a navigable tree. The BeautifulSoup4 library lacks a sgm parser, so an html parser was used instead. This ended up being a beneficial design choice, as html parsers are generally more lenient than xml or sgm parsers when it comes to poorly formatted articles. The given data files had some poorly formatted tags (not closed properly, slight mis-spellings) which BeautifulSoup4 took care of in its parsing.

After the tree is created it can be traversed using BeautifulSoup4’s API. This tree contains a series of Reuter nodes, and each Reuter may contain topics, places, and title nodes, as well as an associated text (representing the body text of the article). From this tree a class label and count of relevant words from the article’s text are created for each article.