Ling 570 HW1

Olga Whelan

Q2

The program is written in Python 2.7.3. Tokenizing is done with regular expressions.

Application: I tried to imitate the tokenizer, the output of which I regularly use for named entity tagging.

This tokenizer uses a set of 13 regex rules to separate the tokens by whitespace (all special characters are separated by whitespace by default) and work around some specific cases below:

<u>hyphenated words</u>: treated as one (*export-oriented*). Their treatment relies on spelling conventions: hyphen as opposed to dash is not supposed to be surrounded by spaces.

<u>words with apostrophe</u>: negative forms of verbs are treated as one token (*couldn't*). *'re, 's, 've* are treated as a separate token (*I 've*)

<u>abbreviations ending with a period</u>: one-letter abbreviations are treated as one token (*a. m.*). Abbreviations starting with a capital letter no more than four letters long are also treated as one token (*Mr.; U.S.; Ph.D.; Mass.*)

<u>&</u>: words connected by & are not separated (*AT&T*)

<u>numbers</u>: decimals and numbers with comma separators are considered one token (*0.5; 1,000*). Number sequences separated by hyphen are also kept together (550-1212)

<u>e-mails</u>: some most frequent patterns of e-mail addresses are recognized as one token (niceandsimple@example.com, very.common@example.com, very.common@example.dept.com)

<u>Problems:</u>

- Corp.'s and Inc.'s – in all my test files these cases are dealt with as I expect (Corp. 's and Inc. 's). But in ex1.tok I consistently get Corp. ' s and Inc. ' s; I didn't manage to resolve this.
- Possible conflict between abbreviations and capitalized 2-4-letter words in the end of the sentence. Might be resolved by including a word list containing such abbreviations.
- Urls have no special treatment – I could not come up with a good regex. Because of the randomness of special characters in urls, everything I tried conflicted with the other regexes. Possibly a different general approach is called for here: instead of giving rules for exceptions to whitespace separations, I should give rules for cases when this separation occurs.

I did not include a word list of abbreviations which could have been helpful, because I could not think of a good way to integrate it into my program.

Q4

tokens in ex1	39824	ex1.voc	10425
tokens in ex1.tok	46319	ex1.tok.voc	7919

Q5

(a) Binomial

P(A)=$\binom{n}{r} p^r (1-p)^r$

In our case the number of trials, n = 500;

the number of expected successes, r = 13 ;

the likelihood of success, p = $\frac{1}{38}$;

1-p = $\frac{37}{38}$ is the likelihood of failure.

Using a binomial distribution calculator, P(A) = **0.1113**

(b) Poisson

P(x;µ) = $\frac{e^{-\mu}(\mu^x)}{x!}$

When n is very large and p is small, binomial probability can be approximated by Poisson distribution:

P(x of n) = $\frac{e^{-np}(np^x)}{x!}$, where n = 500, p = $\frac{1}{38}$, x = 13

Using a Poisson distribution calculator, P(x of n) = **0.1098**