CSCE-689: Natural Language Processing
Programming Assignment: 1 SpamLord

Aniket Sanjiv Bonde (UIN - 825009631)
E-mail: bondeanikets@tamu.edu

January 2018

1 Problem Statement

To write regular expressions that extract phone numbers and regular expressions that extract email addresses

2 Solution: Python function (Note: I have used python 3.6)

```
def process_file(name, f):

    email_normal = '(\w+)\s*(?:@|\s+at\s+|&#x40\.|\(at\))' \
    '\s*([a-zA-Z0-9\.]+)\s*\.\s*(edu|com|co\...)'
    email_hidden = "obfuscate\('([a-zA-Z0-9\.]+)','([a-zA' \
    '-Z0-9\.]+)'\)"

    phone = '\(?(\d{3})\s*(?:\)|\s|-| )\s*(\d{3})\s*' \
    '(?:\s|-| )\s*(\d{4})'

    res= []
    for line in f:
        numbers = re.findall(phone, line)
        for n in numbers:
            number = '%s-%s-%s' % n
            res.append((name,'p',number))

        line = line.replace('-','')
        line = re.sub(r'\s*([\s+\(]dom[\s+\)]|[\s+\(]dot' \
        '[\s+\)]|;)\s*', '.', line, 1000, re.IGNORECASE)
        line = re.sub(r'\s*([\s+\(]where[\s+\)])\s*', '@',
                      line, 1000, re.IGNORECASE)

        matched = re.findall(email_normal, line, re.IGNORECASE)
        for match in matched:
            if match[0].lower() != 'server':
                email = '%s@%s.%s' % match
                res.append((name,'e',email))

        matched = re.findall(email_hidden, line, re.IGNORECASE)
        for match in matched:
            email = '%s@%s' % (match[1],match[0])
            res.append((name,'e',email))

    return res
```

For email, I have designed 2 regexes:

1. Normal emails: `(\w+)\s*(?:@|\s+at\s+|&#x40\.|\(at\))\s*([a-zA-Z0-9\.]+)\s*\.\s*(edu|com|co\...)`
2. Hidden emails: `obfuscate\('([a-zA-Z0-9\.]+)','([a-zA-Z0-9\.]+)'\)`

For phone-nos, I have designed 1 regex:
`\(?(\d{3})\s*(?:\)|\s|-| )\s*(\d{3})\s*(?:\s|-| )\s*(\d{4})`

3 Results and Analysis

As can be seen from the results, accuracy of 100% was obtained on the dev data. It has been ensured that the regex is as generalizable as possible.

```
>>>> python SpamLord.py data_dev\dev data_dev\devGOLD
True Positives (59):
{('ashishg', 'e', 'ashishg@stanford.edu'),
 ('ashishg', 'e', 'rozm@stanford.edu'),
 ('ashishg', 'p', '650-723-1614'),
 ('ashishg', 'p', '650-723-4173'),
 ('ashishg', 'p', '650-814-1478'),
 ('balaji', 'e', 'balaji@stanford.edu'),
 ('bgirod', 'p', '650-723-4539'),
 ('bgirod', 'p', '650-724-3648'),
 ('bgirod', 'p', '650-724-6354'),
 ('cheriton', 'e', 'cheriton@cs.stanford.edu'),
 ('cheriton', 'e', 'uma@cs.stanford.edu'),
 ('cheriton', 'p', '650-723-1131'),
 ('cheriton', 'p', '650-725-3726'),
 ('dabo', 'e', 'dabo@cs.stanford.edu'),
 ('dabo', 'p', '650-725-3897'),
 ('dabo', 'p', '650-725-4671'),
 ('dlwh', 'e', 'dlwh@stanford.edu'),
 ('engler', 'e', 'engler@lcs.mit.edu'),
 ('engler', 'e', 'engler@stanford.edu'),
 ('eroberts', 'e', 'eroberts@cs.stanford.edu'),
 ('eroberts', 'p', '650-723-3642'),
 ('eroberts', 'p', '650-723-6092'),
 ('fedkiw', 'e', 'fedkiw@cs.stanford.edu'),
 ('hager', 'e', 'hager@cs.jhu.edu'),
 ('hager', 'p', '410-516-5521'),
 ('hager', 'p', '410-516-5553'),
 ('hager', 'p', '410-516-8000'),
 ('hanrahan', 'e', 'hanrahan@cs.stanford.edu'),
 ('hanrahan', 'p', '650-723-0033'),
```

```
 ('hanrahan', 'p', '650-723-8530'),
 ('horowitz', 'p', '650-725-3707'),
 ('horowitz', 'p', '650-725-6949'),
 ('jks', 'e', 'jks@robotics.stanford.edu'),
 ('jurafsky', 'e', 'jurafsky@stanford.edu'),
 ('jurafsky', 'p', '650-723-5666'),
 ('kosecka', 'e', 'kosecka@cs.gmu.edu'),
 ('kosecka', 'p', '703-993-1710'),
 ('kosecka', 'p', '703-993-1876'),
 ('kunle', 'e', 'darlene@csl.stanford.edu'),
 ('kunle', 'e', 'kunle@ogun.stanford.edu'),
 ('kunle', 'p', '650-723-1430'),
 ('kunle', 'p', '650-725-3713'),
 ('kunle', 'p', '650-725-6949'),
 ('lam', 'e', 'lam@cs.stanford.edu'),
 ('lam', 'p', '650-725-3714'),
 ('lam', 'p', '650-725-6949'),
 ('latombe', 'e', 'asandra@cs.stanford.edu'),
 ('latombe', 'e', 'latombe@cs.stanford.edu'),
 ('latombe', 'e', 'liliana@cs.stanford.edu'),
 ('latombe', 'p', '650-721-6625'),
 ('latombe', 'p', '650-723-0350'),
 ('latombe', 'p', '650-723-4137'),
 ('latombe', 'p', '650-725-1449'),
 ('levoy', 'e', 'ada@graphics.stanford.edu'),
 ('levoy', 'e', 'melissa@graphics.stanford.edu'),
 ('levoy', 'p', '650-723-0033'),
 ('levoy', 'p', '650-724-6865'),
 ('levoy', 'p', '650-725-3724'),
 ('levoy', 'p', '650-725-4089')}
False Positives (0):
set()
False Negatives (0):
set()
Summary: tp=59, fp=0, fn=0
```

4 How to run my code

NOTE: All the print statement have been replaced by print(), as python 3.6 needs parentheses. Apart from that all other code is the same. Just changing the path of input data changes data on which program runs on.

5 Any known bugs, problems, or limitations of the program

I have made the program as generalized as possible but some edge cases might be/can be missed by the program. Currently there are no known bugs, just that this program needs python 3.6 to run.