CSCE - 689 NLP FOUNDATION TECHNIQUE
Programming Assignment #1 - SpamLord
Debanik Lahiri (UIN: 224008867)

How to Run:

1. Unzip the folder SpamLord.zip
2. Copy and paste the file SpamLord.py.
3. Go to the directory of SpamLord.py and enter in terminal:
   $ python SpamLord.py <Directory with webpages> <File with solutions>

   For example in the given assignment cases:
   <Directory with webpages>: data_dev/dev/
   <file with solutions>: data_dev/devGOLD
   Run:
   $ python SpamLord.py data_dev/dev/ data_dev/devGOLD

Results & Analysis:

The program SpamLord.py was able to achieve 0 errors on the training data. A summary of output on running:
Summary: tp=59, fp=0, fn=0

The program is trained with rules(regexes) based on the training data, which is chiefly university data. Moreover, I have also searched multiple websites to find out the techniques for hiding email addresses and phone numbers. Special regexes were required to handle different obfuscation techniques like:
<script type="text/javascript">obfuscate(’cse.tamu.edu’,’huangrh’)</script>

The regular expressions I have used are as follows:

For extracting email addresses:

Regular Expression	Purpose
'(([\w-]+|[\w-]+\.[\w-]+)'	Extract the local part of email address.
'(\s.?\(followed by.*)?'	Some email addresses have 'followed by' in the address
'(\s(at|where)\s|\s?(@|&.*;)\s?)'	Search for 'at' or 'where'
([\w-]+|[\w-]+([\.;]|\s(do?t)\s|\s)[\w-]+)	The domain of the address except the final extension like .com/.net
(\s(do?t|do?m)\s|\s|[\.;])	Different types of writing "."(Dot)
((-?e-?d-?u|-?c-?o-?m|-?n-?e-?t|-?o-?r-?g|-?g-?o-?v)\b)	Email address extension
(obfuscate\('(\w+\.(edu|com|net|org|gov))','(\w+)'\)))	Search for obfuscate() function on email address

For extracting phone numbers:

Regular Expression	Purpose
\(?(\d{3})\)?	Area Code of Phone number
[-]	Separator of Phone number
(\d{3})	Second part of the Phone number
(\d{4})	Last part of Phone number

Bugs/Problems/Limitations:

1. The program is trained on rules/regexes which have been generated based on the training data. Thus, there might be test cases or techniques of hiding data which have not been covered by the program.
2. One particular test case had the statement 'Server at <domain address>'. This is not an actual email address but refers to the physical location of the server. To circumvent this test case I hardcoded 'Server' string check. This might cause issues where there is actually an email address Server@address.com.
3. Any alternate obfuscate() function will not be handled.
4. I have covered the following domain extensions: edu, com, net, org, gov.