CS6320, Spring 2018
Dr. Mithun Balakrishna
Homework 2
Due Wednesday, February 21st, 2018 11:59pm

B. Problems:

1. Regular Expression (20 points)

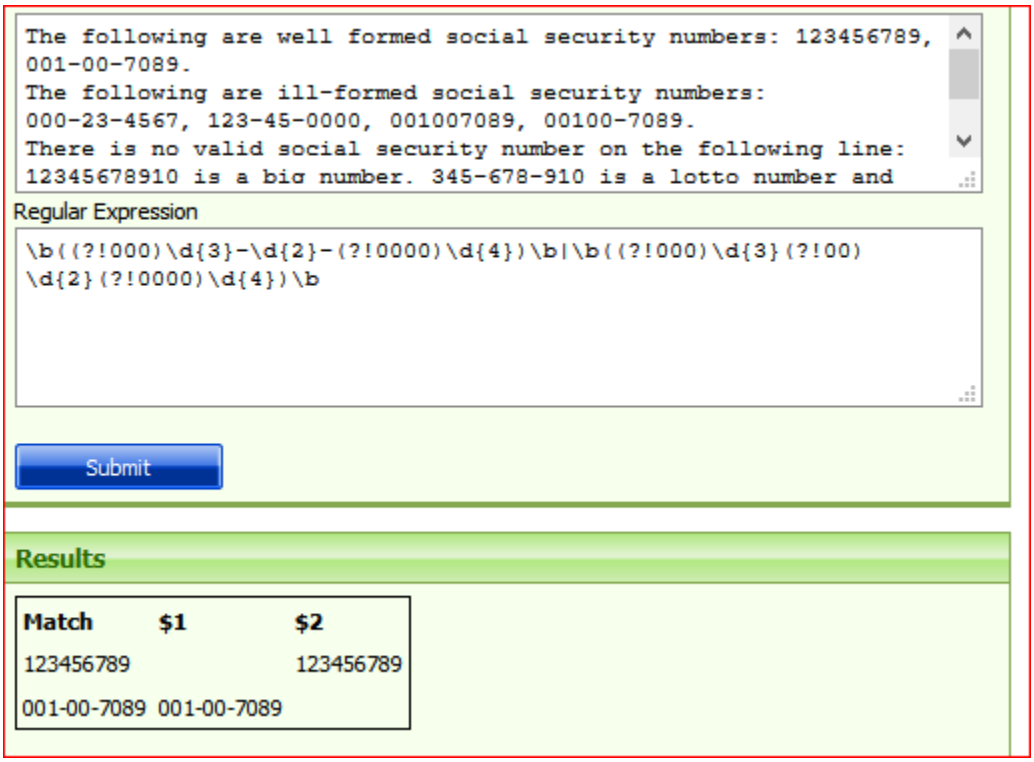
Write a single regular expression for identifying social security numbers in text. The social security numbers consists of:

- 9 digits
- must be preceded by one or more spaces or beginning of line
- must be followed by one or more spaces or end of line In addition there are certain restrictions:
- first three digits cannot be all zeros
- last four digits cannot be all zeros
- nine digits can all be next to each other or there can be a hyphen between:
  - third and fourth digit, and
  - fifth and sixth digit

The following are well formed social security numbers: 123456789, 001-00-7089.
The following are ill-formed social security numbers: 000-23-4567, 123-45-0000, 001-007089, 00100-7089.

There is no valid social security number on the following line:
12345678910 is a big number, 345-678-910 is a lotto number and 3333333334 is a 10 digit number.

ANS.: \b((?!000)\d{3}-\d{2}-(?!0000)\d{4})\b|\b((?!000)\d{3}\d{2}(?!0000)\d{4})\b



2. Bigram Probabilities (40 points):

An automatic speech recognition system has provided a written sentence as the possible interpretation to a speech input.

Compute the probability of a written sentence using the bigram language model trained on *HW2_F17_NLP6320-NLPCorpusTreebank2Parts-CorpusA.txt* (provided as Addendum to this homework on eLearning).

Note: Please use whitespace (i.e. space, tab, and newline) to tokenize the corpus into words/tokens that are required for the bigram model. Do NOT perform any type of word/token normalization (i.e. stem, lemmatize, lowercase, etc.). Creation and matching of bigrams should be exact and case-sensitive. Do NOT split the corpus into sentences. Please consider the entire corpus as a single string for tokenization and computation of bigrams.

Compute the sentence probability under the three following scenarios:

i. Use the bigram model without smoothing.
ii. Use the bigram model with add-one smoothing

iii. Use the bigram model with Good-Turing discounting.

Your computer program should do the following:

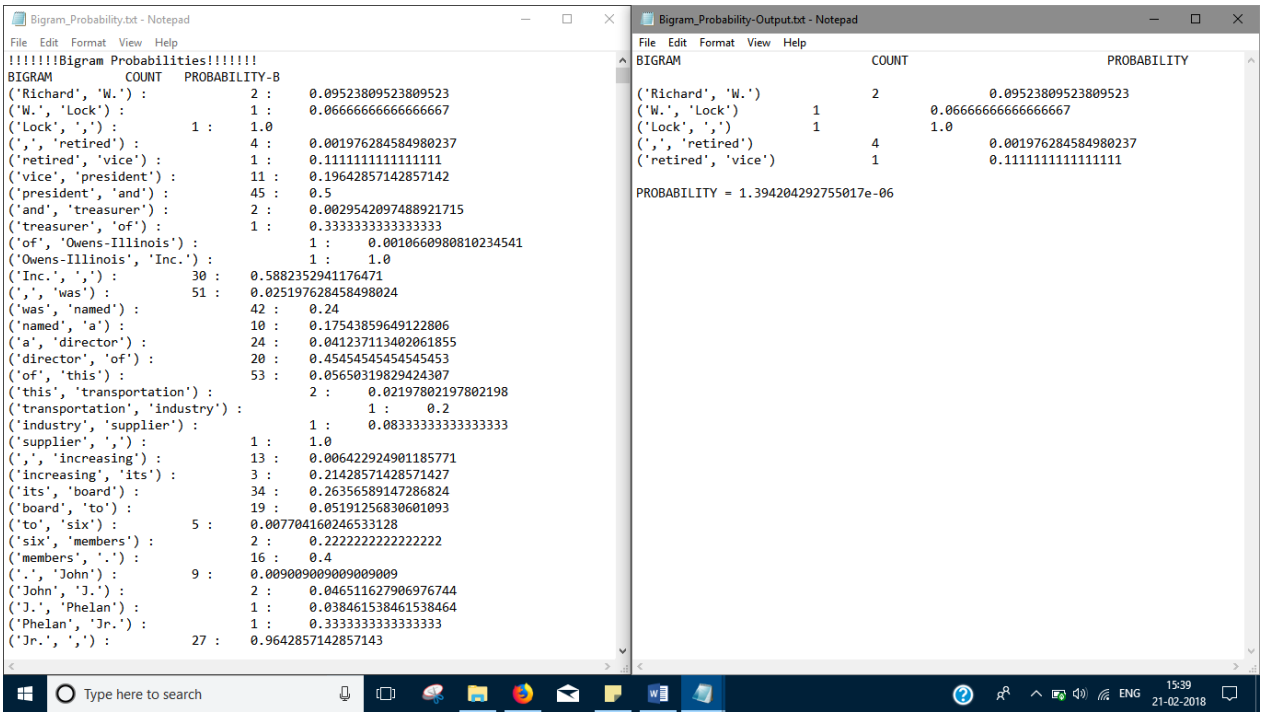
1. Compute the bigram counts on the given corpus (*HW2_F17_NLP6320NLPCorpusTreebank2Parts-CorpusA.txt*).
2. For a given input written sentence:
   a. For each of the three scenarios, construct a table with the bigram counts for the sentence.
   b. For each of the three scenarios, construct a table with the bigram probabilities for the sentence.
   c. For each of the three scenarios, compute the total probability for the sentence.

ANS:

Please see the attached zip file named problem2

Some Screenshots on the OUTPUT:

i. BIGRAM WITHOUT SMOOTHING



Bigram_Probability.txt - Notepad

```
!!!!!!!Bigram Probabilities!!!!!!!
BIGRAM          COUNT   PROBABILITY-B
('Richard', 'W.') :             2 :     0.09523809523809523
('W.', 'Lock') :                1 :     0.06666666666666667
('Lock', ',') :         1 :     1.0
(',', 'retired') :              4 :     0.001976284584980237
('retired', 'vice') :           1 :     0.1111111111111111
('vice', 'president') :         11 :    0.19642857142857142
('president', 'and') :          45 :    0.5
('and', 'treasurer') :          2 :     0.0029542097488921715
('treasurer', 'of') :           1 :     0.3333333333333333
('of', 'Owens-Illinois') :              1 :     0.0010660980810234541
('Owens-Illinois', 'Inc.') :            1 :     1.0
('Inc.', ',') :         30 :    0.5882352941176471
(',', 'was') :          51 :    0.025197628458498024
('was', 'named') :              42 :    0.24
('named', 'a') :                10 :    0.17543859649122806
('a', 'director') :             24 :    0.041237113402061855
('director', 'of') :            20 :    0.45454545454545453
('of', 'this') :                53 :    0.05650319829424307
('this', 'transportation') :            2 :     0.02197802197802198
('transportation', 'industry') :                1 :     0.2
('industry', 'supplier') :              1 :     0.08333333333333333
('supplier', ',') :             1 :     1.0
(',', 'increasing') :           13 :    0.006422924901185771
('increasing', 'its') :         3 :     0.21428571428571427
('its', 'board') :              34 :    0.26356589147286824
('board', 'to') :               19 :    0.05191256830601093
('to', 'six') :         5 :     0.007704160246533128
('six', 'members') :            2 :     0.2222222222222222
('members', '.') :              16 :    0.4
('.', 'John') :         9 :     0.009009009009009009
('John', 'J.') :                2 :     0.046511627906976744
('J.', 'Phelan') :              1 :     0.038461538461538464
('Phelan', 'Jr.') :             1 :     0.3333333333333333
('Jr.', ',') :          27 :    0.9642857142857143
```

Bigram_Probability-Output.txt - Notepad

```
BIGRAM                          COUNT                           PROBABILITY

('Richard', 'W.')               2               0.09523809523809523
('W.', 'Lock')          1               0.06666666666666667
('Lock', ',')           1               1.0
(',', 'retired')                4               0.001976284584980237
('retired', 'vice')             1               0.1111111111111111

PROBABILITY = 1.394204292755017e-06
```

ii. ADD ONE SMOOTHING

Add_One_Smoothing.txt - Notepad

```
!!!!!!!Add One Smoothing!!!!!!!
BIGRAM          COUNT   PROBABILITY-AOS
('Richard', 'W.') :             2 :      0.0005331437711036076
('W.', 'Lock') :                1 :      0.0003558085749866572
('Lock', ',') :                 1 :      0.0003566969859104691
(',', 'retired') :              4 :      0.000655307994757536
('retired', 'vice') :           1 :      0.00035618878005342833
('vice', 'president') :                 11 :     0.0021193924408336277
('president', 'and') :          45 :     0.008075842696629214
('and', 'treasurer') :          2 :      0.0004774789113480821
('treasurer', 'of') :           1 :      0.0003565697985380638 5
('of', 'Owens-Illinois') :              1 :      0.0003056234718826406
('Owens-Illinois', 'Inc.') :            1 :      0.0003566969859104691
('Inc.', ',') :                 30 :     0.005479936362029344
(',', 'was') :          51 :     0.006815203145478375
('was', 'named') :              42 :     0.007438159487977858
('named', 'a') :                10 :     0.0019424333392194949
('a', 'director') :             24 :     0.004040077569489335
('director', 'of') :            20 :     0.0037168141592920354
('of', 'this') :                53 :     0.008251833740831296
('this', 'transportation') :            2 :      0.000526592943654555
('transportation', 'industry') :                1 :      0.00035644427018356799
('industry', 'supplier') :              1 :      0.000355998576005696
('supplier', ',') :             1 :      0.0003566969859104691
(',', 'increasing') :           13 :     0.001834862385321101
('increasing', 'its') :                 3 :      0.0007117437722419929
('its', 'board') :              34 :     0.006102877070619006
('board', 'to') :               19 :     0.003348961821835231
('to', 'six') :                 5 :      0.0009592326139088729
('six', 'members') :            2 :      0.0005342831700801424
('members', '.') :              16 :     0.0030109812256464753
('.', 'John') :                 9 :      0.001514004542013626
('John', 'J.') :                2 :      0.0005310674455655868
('J.', 'Phelan') :              1 :      0.0003355113636363636364
('Phelan', 'Jr.') :             1 :      0.0003565697985380638 5
('Jr.', ',') :          27 :     0.004969826056088037
```

Add_One_Smoothing-Output.txt - Notepad

```
BIGRAM                          COUNT                           PROBABILITY

('Richard', 'W.')               3               0.0005331437711036076
('W.', 'Lock')          2               0.0003558085749866572
('Lock', ',')           2               0.0003566969859104691
(',', 'retired')                5               0.000655307994757536
('retired', 'vice')             2               0.00035618878005342833

PROBABILITY = 1.5793772972190785e-17
```

iii. GOOD-TURING DISCOUNTING

Good_Turing_Discounting.txt - Notepad

```
!!!!!!!Good Turing Discounting!!!!!!!
BIGRAM          COUNT   PROBABILITY-GTD
('Richard', 'W.') :             2 :      3.424561752430911e-05
('W.', 'Lock') :                1 :      7.418169862907512e-06
('Lock', ',') :                 1 :      7.418169862907512e-06
(',', 'retired') :              4 :      8.918617614269788e-05
('retired', 'vice') :           1 :      7.418169862907512e-06
('vice', 'president') :                 11 :     0.0003836317135549872 3
('president', 'and') :          45 :     0.001568627450890392
('and', 'treasurer') :          2 :      3.424561752430911e-05
('treasurer', 'of') :           1 :      7.418169862907512e-06
('of', 'Owens-Illinois') :              1 :      7.418169862907512e-06
('Owens-Illinois', 'Inc.') :            1 :      7.418169862907512e-06
('Inc.', ',') :                 30 :     0.0
(',', 'was') :          51 :     0.001773231031543052
('was', 'named') :              42 :     0.0014663256606990622
('named', 'a') :                10 :     0.0002857954776113344
('a', 'director') :             24 :     0.0008525149190110827
('director', 'of') :            20 :     0.0005967604433077579
('of', 'this') :                53 :     0.0
('this', 'transportation') :            2 :      3.424561752430911e-05
('transportation', 'industry') :                1 :      7.418169862907512e-06
('industry', 'supplier') :              1 :      7.418169862907512e-06
('supplier', ',') :             1 :      7.418169862907512e-06
(',', 'increasing') :           13 :     0.0001404142219547665 5
('increasing', 'its') :                 3 :      6.4364102778332 2e-05
('its', 'board') :              34 :     0.0011935208866155158
('board', 'to') :               19 :     0.0010230179028132991
('to', 'six') :                 5 :      0.0001369038664058974
('six', 'members') :            2 :      3.424561752430911e-05
('members', '.') :              16 :     0.0005797101449275362
('.', 'John') :                 9 :      0.0001989201477692526 2
('John', 'J.') :                2 :      3.424561752430911e-05
('J.', 'Phelan') :              1 :      7.418169862907512e-06
('Phelan', 'Jr.') :             1 :      7.418169862907512e-06
('Jr.', ',') :          27 :     0.0006365444728616084
```

Good_Turing_Discounting-Output.txt - Notepad

```
BIGRAM                          COUNT                           PROBABILITY

('Richard', 'W.')               1.0042527339003646              3.424561752430911e
('W.', 'Lock')          0.21753783122976278             7.418169862907512e-06
('Lock', ',')           0.21753783122976278             7.418169862907512e-06
(',', 'retired')                2.6153846153846154              8.918617614269788e
('retired', 'vice')             0.21753783122976278             7.418169862907512e

PROBABILITY = 1.2467887269916974e-24
```

3. Transformation Based POS Tagging (40 points)

For this question, you have been given a POS-tagged training file, *HW2_F17_NLP6320_POSTaggedTrainingSet.txt* (provided as Addendum to this homework on eLearning), that has been tagged with POS tags from the Penn Treebank POS tagset (Figure 1).

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two, three*	TO	"to"	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential 'there'	*there*	VB	verb, base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb, past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb, gerund	*eating*
JJ	adjective	*yellow*	VBN	verb, past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb, non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb, 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, singular	*IBM*	$	dollar sign	$
NNPS	proper noun, plural	*Carolinas*	#	pound sign	#
PDT	predeterminer	*all, both*	"	left quote	' or "
POS	possessive ending	*'s*	"	right quote	' or "
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one's*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

Figure 1. Penn Treebank POS tagset

Use the POS tagged file to perform:

a. Transformation-based POS Tagging: Implement Brill's transformation-based POS tagging algorithm using ONLY the previous word's tag to create transformation rules.

b. Naïve Bayesian Classification (Bigram) based POS Tagging:

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(t_1^n | w_1^n) \approx \underset{t_1^n}{\operatorname{argmax}} \prod_{i=1}^{n} P(w_i | t_i) P(t_i | t_{i-1})$$

c. Apply model (a) and (b) on the sentence below, and show the difference in error rates.

Sentence: *The president wants to control the board 's control*

Manual POS Tagged Sentence: *The*_DT *president*_NN *wants*_VBZ *to*_TO *control*_VB *the*_DT *board*_NN *'s*_POS *control*_NN

ANS: Please see Problem3.zip

Screenshot of OUTPUT-

Brills Rules

Brills_Rules.txt - Notepad

File Edit Format View Help

PREVIOUS WORD	FROM	TO	SCORE
MD	VBP	VB	54
MD	NN	VB	45
VBZ	VBD	VBN	38
PRP	VBN	VBD	35
PRP	VB	VBP	32
NNP	VBN	VBD	31
PRP	POS	VBZ	31
NNS	VB	VBP	30
TO	VBP	VB	28
TO	NN	VB	27
VB	VBD	VBN	22
VBD	VBD	VBN	17
VBP	VBD	VBN	17
DT	VB	NN	17
VBN	VBZ	NNS	12

Brills output

brills-output.txt - Notepad

File Edit Format View Help

```
Word            Brills Tag
The                     DT
president                       NN
wants                   VBZ
to                      TO
control                 VB
the                     DT
board                   NN
's                      POS
control                 NN

Brills Tag Error Rate: 0.0
```

Naïve Byes Tags

Naive_Bayes_Tags.txt - Notepad

File Edit Format View Help

```
Word            Most Probable Tag

Brainpower                      NNP
,                       ,
not                     RB
physical                        JJ
plant                   NN
,                       ,
is                      VBZ
now                     RB
a                       DT
firm                    NN
's                      POS
chief                   JJ
asset                   NN
.                       .
Brainpower                      NNP
,                       ,
not                     RB
physical                        JJ
plant                   NN
,                       ,
is                      VBZ
now                     RB
```

Naïve bayes Output

naive-bayes-output.txt - Notepad

File Edit Format View Help

```
Word            Naive Bayes Tag
The                     DT
president                       NN
wants                   VBZ
to                      TO
control                 NN
the                     DT
board                   NN
's                      POS
control                 NN

Naive Bayes Tag Error Rate: 0.1111111111111111
```

Command Line output:

```
GENERATING RULES..........
Please wait it will takes 5 min to generate rules!!!
Rule  1
Rule  2
Rule  3
Rule  4
Rule  5
Rule  6
Rule  7
```