Information Theoretic Modeling – Exercise 6

Haibo Jin
Student number: 014343698

1 Problem 1

It has been implemented in *multi_nml.py*. And Figure 1 shows the running result of the program.

```
local@tktl-2013:~/Nick/Master/courses/Information Theoretic Modeling/exercise/6$ python multi_nml.py
code-length: 408.928413692
```

Figure 1: Multinomial NML.

2 Problem 2

(a)

This problem has been implemented in *bayes_network.py*. Figure 2 shows the running result of the program.

```
local@tktl-2013:~/Nick/Master/courses/Information Theoretic Modeling/exercise/6$ python bayes_network.py
The total conde-length: 2899.86350867
```

Figure 2: The total fNML code-length of the given bayesian network.

(b)

3 Problem 3

I first see the distributions of the word and bigrams, then simply replace the frequent ones. *data* has the original file that to be compressed and *data.py* has the program that can generate the file. Please see Figure 3 for the total word count before and after the compression, though it is not so efficient.

```
local@tktl-2013:~/Nick/Master/courses/Information Theoretic Modeling/exercise/6$ wc data
  48  535 3146 data
local@tktl-2013:~/Nick/Master/courses/Information Theoretic Modeling/exercise/6$ wc data.py
   7  452 2983 data.py
local@tktl-2013:~/Nick/Master/courses/Information Theoretic Modeling/exercise/6$
```

Figure 3: Total word count before and after the compression.

4 Problem 4

Please see the implementation in *google_distance.py*.

(a)

Figure 4 shows the pairwise distance of words in the given set.

```
local@tktl-2013:~/Nick/Master/courses/Information Theoretic Modeling/exercise/6$ python google_distance.py
Andrey,Kolmogorov: 0.584197452151
Andrey,complexity: 0.23666204457
Andrey,stochastic: 0.495309905214
Andrey,porridge: 0.549378877674
Andrey,Rissanen: 0.419326502131
Andrey,breakfast: 0.425667233491
Andrey,omelette: 0.503620854393
Andrey,broccoli: 0.444005830717
Kolmogorov,complexity: 0.528048397588
Kolmogorov,stochastic: 0.526363464197
Kolmogorov,porridge: 0.547442697396
Kolmogorov,Rissanen: 0.305033745037
Kolmogorov,breakfast: 0.745822725586
Kolmogorov,omelette: 0.617540412773
Kolmogorov,broccoli: 0.5836028253
complexity,stochastic: 0.539475215383
complexity,porridge: 0.500627327005
complexity,Rissanen: 0.525211389524
complexity,breakfast: 0.655400858094
complexity,omelette: 0.507430883654
complexity,broccoli: 0.476362539207
stochastic,porridge: 0.627649811291
stochastic,Rissanen: 0.485525581779
stochastic,breakfast: 0.83909037207
stochastic,omelette: 0.698662342907
stochastic,broccoli: 0.613751705603
porridge,Rissanen: 0.587758316384
porridge,breakfast: 0.621732836027
porridge,omelette: 0.255365040053
porridge,broccoli: 0.509156208555
Rissanen,breakfast: 0.78418091874
Rissanen,omelette: 0.675991789118
Rissanen,broccoli: 0.552953800559
breakfast,omelette: 0.60182824153
breakfast,broccoli: 0.667086258651
omelette,broccoli: 0.489692702262
```

Figure 4: Normalized google distance for every pair of words.

(b)

Figure 5 shows the heatmap of the 9×9 distance matrix. The part with cold colors indicates the distance between the two words is small while warm colors means the two words are not so close. As we can see, the distance between *Andrey* and *complexity* is small. The same for *Kolmogorov* and *Rissanen*, *porridge* and *omelette*. So, the NGD does indicate the semantic relatedness. However, it does not indicate all the pairs that are correlated, such as *Andrey* and *Kolmogorov*.

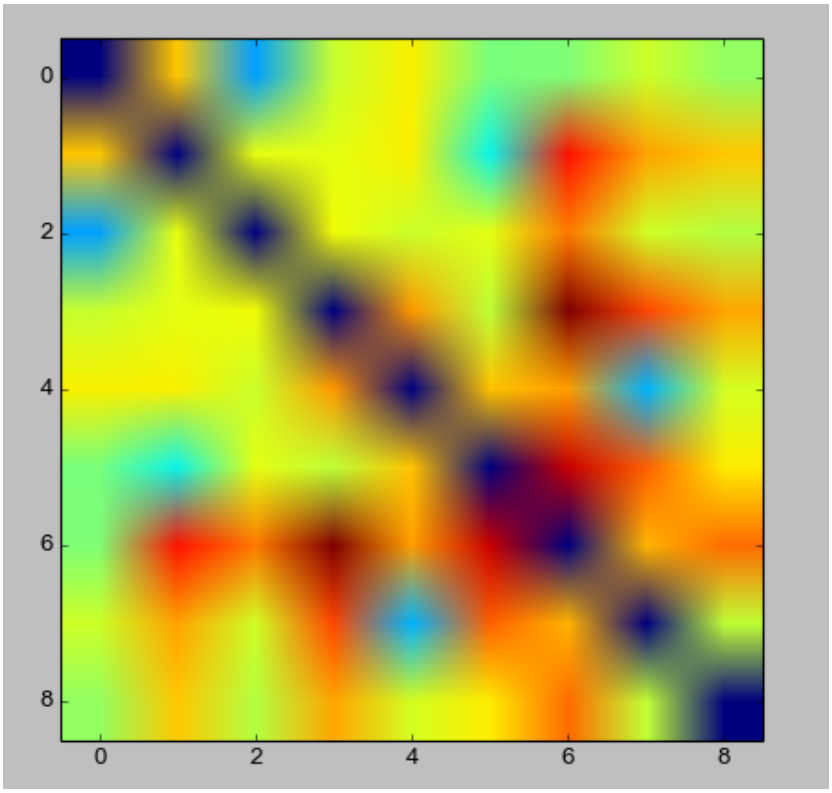

Figure 5: Heatmap of the 9×9 distance matrix.