Markov Model

Variables:

* data={}, format {string:string} = {message: sexual rating}
  + {‘this guy is a dick’ : ‘0’, …..etc}
* markov\_model = {}, format {string:{string: #,string:#, etc}, etc} = {first word: {second word: #, second word: # etc}, etc}
  + Markov model holds the edge value. To find the edge value of ‘like’ -> ‘spam’, first find the key if the outer dictionary in Markov Model that is ‘like’. Within that key is a value and that value is an inner dictionary. Find ‘spam’ as key in the inner dictionary. That value from the inner dictionary is the edge value ‘like’ towards ‘spam’.
  + This represents the co-occurrence of words after a word.
* newmarkov\_model = {}, similar format as above
  + Now, the edge value is changed. New Edge Value = Current Edge Value / Total Outward Value
* node\_model = {}, format {string: #} = {word: node value}
  + Node Model holds the node value of each word. Node value = Number of comments it was mentioned/total comments. Because score 2 has double value, make sure to double the count in total comments.
* newnodemodel = {}, similar format as above
  + Second Node Model holds the probability of a word (weight of how likely it would be sexual).
  + New Node Value = Current node + Edge TOWARDS Current Node \* Previous Node +…other edges and nodes TOWARD the current node + etc

Within the Main Method

1. Read in the file and store it into dictionary called ‘data’, the format is key: message (string) and value: sexual rating (string). Which means ‘data’ is the list of all messages. I can find the total number of messages by the size of the dictionary.
2. Create the Markov Model with just co-occurrence values. Calculate the edge values by forward messages. Parameter: data. Now, Professor Macropol mentions Bi-Directional Model, but only to forward messages, previous messages are not forgotten. If we do reverse messages we might as well double the edge values. We don’t always have two edges between two nodes because whenever one edge is formed, the opposite edge appears! That does not make sense! She did change her ways, but the last time she did an example, (a) was the way.
   1. For example w/ only forward: I -> like -> cheese -> and -> spam -> and -> like -> cheese
      1. {I:{like:1}, like: {cheese:2}, cheese: {and:1}, spam: {and:1},and:{spam:1,like:1} }
   2. With forward and backward: I -> like -> cheese -> and -> spam -> and -> like -> cheese, cheese -> like -> and-> spam -> and -> cheese -> like -> I
      1. {I:{like:1}, like: {cheese:2,and:1}, cheese: {and:1,like:2}, spam: {and:2},and:{spam:2,like:1,cheese:1} }
      2. Notice that it was tedious that it was the same value for both edges? Whenever ‘spam’ mentioned after ‘and’, the edge from ‘spam’ to ‘and’ was also incremented.
3. Create a new Markov Model.
   1. Now to find all the edge values pointing OUTWARDS a word. For example, find all the values pointing out of ‘and’. Loop every value in the inner dictionary of ‘and’ and calculate the outward value.
      1. and:{spam:1,like:1}
      2. outwards value of ‘and’ = 1 + 1 = 2
      3. ‘and’ -> ‘spam' Current edge value = 1
      4. New edge value = current edge value / outward value = ½
   2. Must make another copy, because outward value of a node will not be consistent
4. Create the initial node model.
   1. Denominator: Find total messages: number of keys of data = number of messages
      1. Total messages = number of messages + number if messages scored 2
   2. Numerator: Count Words Occurrence in 2 and 1. If 2, count it as 2. If 1, count as 1.
   3. Node value = numerator / denominator
5. Create a new node model with the new Markov Model and initial node model.
   1. If the word is not on the list, the default value is zero. That is why there is only 300 something words because it only shows words found in 1 or 2 sexual rating.

Results: each line present the node value of the word from beginning to end, result is the average\*1000

0

0.004057789735028529

0.008357555454784182

0

0

0

0

0

go for the kill cuz she cant kill= 1.551918148726589 (Actual Score 0, Predicted Score 2)

0.0015234783454514506

0.0004234767440688049

0.0034413016104558494

0

0

Kaceytron makes a hospital porno= 1.077651339995221(Actual Score 1, Predicted Score 1)

0

0.00466211866196364

0.0033207236295039752

0

0.0009194883936512709

0

0

0

0

0.0015225626309014176

0

Bitch are u live if ye then take out your clothes= 0.9477175741836639 (Actual Score 2,Predicted Score 0)

In conclusion, the co-occurrence values does affect the sexual rating prediction. For example, the first comment gained a lot of weight because ‘for’ and ‘the’ were the heaviest. We need normalize the second node model to fix our prediction. With three comments for samples, it seems that 1/3 was correct.