CIS 122

The Thrilling Conclusion

Making Markov Dictionaries

- Start with a list of words
- Initialize empty dictionary
- For each word in word list:
  - If it doesn't have an entry, add it to the dictionary
  - Append following word to associated list
- Let's check our code

Chaining Words Together

- We have a Markov Dictionary
  - List of possible following words for any first word
- Let's write a function constructSentence(markovDictionary)
  - Takes a Markov Dictionary as input
  - Produces a string of words forming a sentence
- Where do we start?

Chaining Words Together

- Find a word that could start a sentence
  - Look up words following '.' in our dictionary
  - Pick one
- Find a word that could follow that word
  - Look up words following current word in our dictionary
  - Pick one
- Repeat until we find another '.'
- How do we randomly select something from a list?
  - random.choice(myList)

Finishing Touches

- Given a string of text we can:
  - Construct a Markov Dictionary
  - Generate text of our own
- Now we just need to get that initial text
  - Could type it in ourselves
  - Easier to read it from a file
- But how do we interact with files?
  - For that matter, what is a file?

What is a file?

- At the lowest level, a file is just a bunch of 1's and 0's
  - Bits
- Different programs do different things with this data
  - Notepad interprets data as characters
  - Adobe Reader interprets data as a pdf
- So what differentiates a text file from a pdf?
  - Extensions
  - .txt
  - .pdf
- Tell computer which program should interpret this file

Python Files

- So how does Python interact with files?
  - With file objects

```
f = file("myFile.txt", "r")
```

filename open for reading

- What can we do with files?
  - f.read() # If file is open for writing
  - f.write(text) # If file is open for writing
  - f.close() # Close file when you're done with it

Python Files

```
def copy(inFilename, outFilename):
    infile = file(inFilename, 'r')
    text = infile.read()
    infile.close()

    outfile = file(outFilename, 'w')
    outfile.write(text)
    outfile.close()
```

Python Files

- Python file objects point to a position in a file
  - f = file("myFile.txt", "r")
  - f points to the beginning of the file
- As it reads through the file, its position changes
  - f.read()
  - Now f points to the end of the file
- What happens if you call f.read() a second time?

Python Files

- Python file objects point to a position in a file
  - f = file("myFile.txt", "r")
  - f points to the beginning of the file
- As it reads through the file, its position changes
  - f.read()
  - Now f points to the end of the file
- What happens if you call f.read() a second time?
- This allows Python to read through files bits at a time
  - f.readLine() # Read just one line
  - f.seek(charNum) # Move to a specific position in file

Put it all Together

- Try writing a function markov(filename, numSentences)
  - Takes a filename and a number of sentences to produce
  - Generates that number of sentences
- Mostly calling functions we've already written