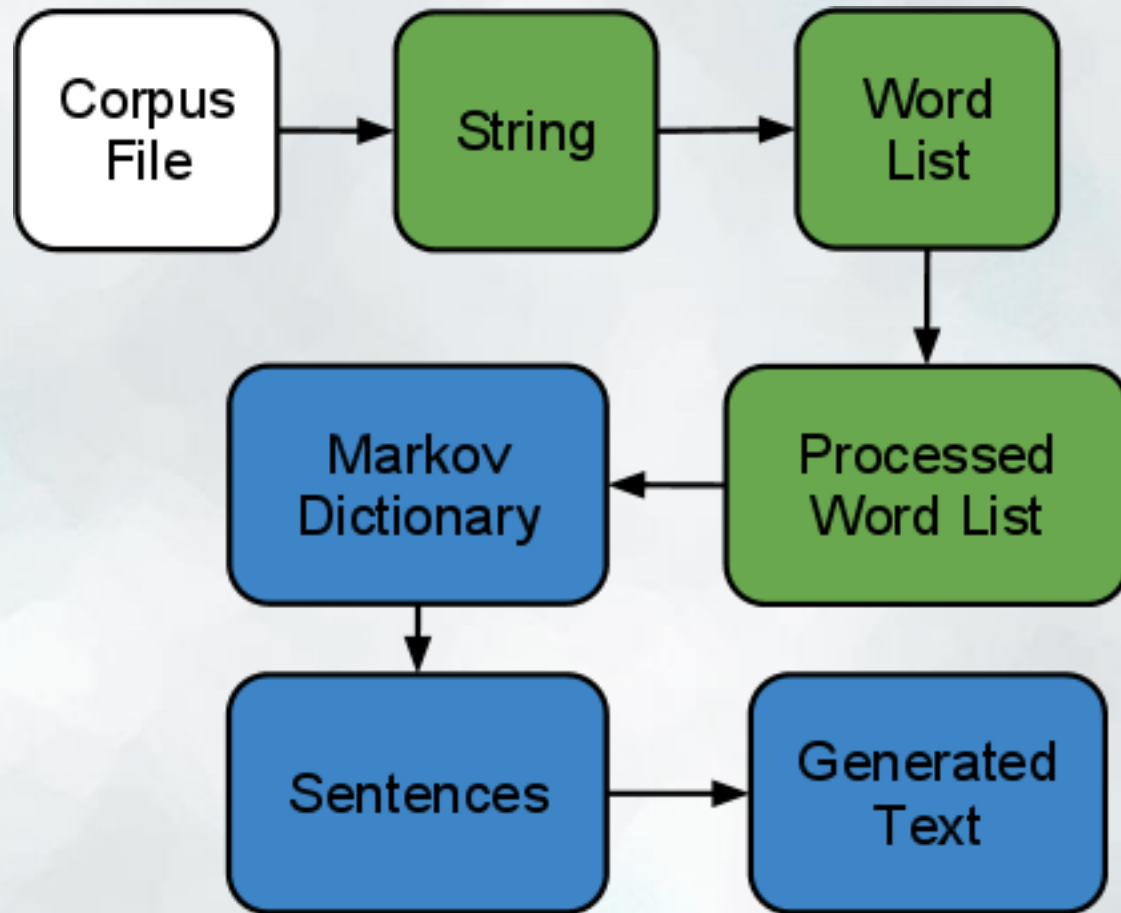


CIS 122

That's the Key

The Big Picture



Dictionary Review

- A dictionary is a set of key-value pairs
 - `myDict = {key1 : value1, key2 : value2, key3 : value3, ...}`
- We look up values in a dictionary by their keys
 - `myDict[key1] → value1`
- Dictionaries are mutable
 - We can reassign values after the fact
 - `myDict[key1] = 12`

Dictionary Quiz

```
scores = { }
```

```
scores[ 'red' ] = 3
```

```
scores[ 'blue' ] = scores[ 'red' ] + 1
```

```
scores[ 'red' ] += 1  (short for scores[ 'red' ] = scores[ 'red' ]+1)
```

```
scores[ 'yellow' ] = scores[ 'blue' ] + scores[ 'red' ]
```

```
print scores[ 'red' ]
```

```
print scores[ 'blue' ]
```

```
print scores[ 'yellow' ]
```

Have I Seen this Key Before?

- We can only look up keys already in our dictionary

```
>>> coinValue = { 'penny' : 1, 'nickel' : 5, 'dime' :10 }
```

```
>>> coinValue[ 'quarter' ]
```

```
<ERROR>
```

- How do we tell if a key is present?
 - Use the `in` keyword

```
>>> 'penny' in coinValue
```

```
True
```

```
>>> 'quarter' in coinValue
```

```
False
```

Have I Seen this Key Before?

- The `in` keyword works on any kind of sequence

```
5 in [1, 2, 3, 4, 5]
```

```
True
```

```
6 in [1,2,3,4,5]
```

```
False
```

```
'a' in 'lighthouse'
```

```
False
```

```
'light' in 'lighthouse'
```

```
True
```

Markov Time

- Let's use a Python dictionary to represent a Markov Dictionary
- What would our keys be?
- What would our values be?

Markov Time

- Let's write a function `makeMarkovDict(wordList)`
 - Takes a processed word list as input
 - Return a Markov Dictionary
 - Keys are words in list
 - Values are lists of words following that key
- Where do we start?