Markov Text Generation

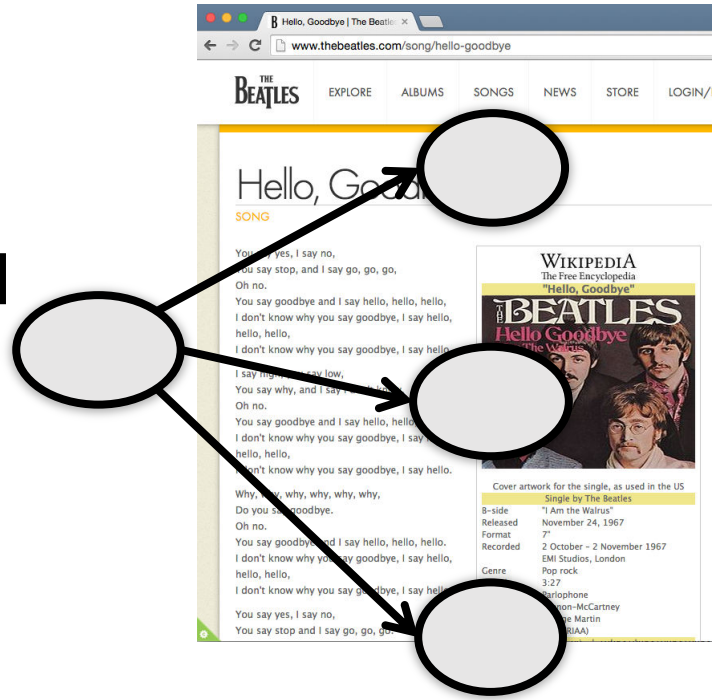
Implementation

By the end of this video you will be able to…

- Describe the class design of a Markov text generator
- Implement the Markov text generator using lists of lists

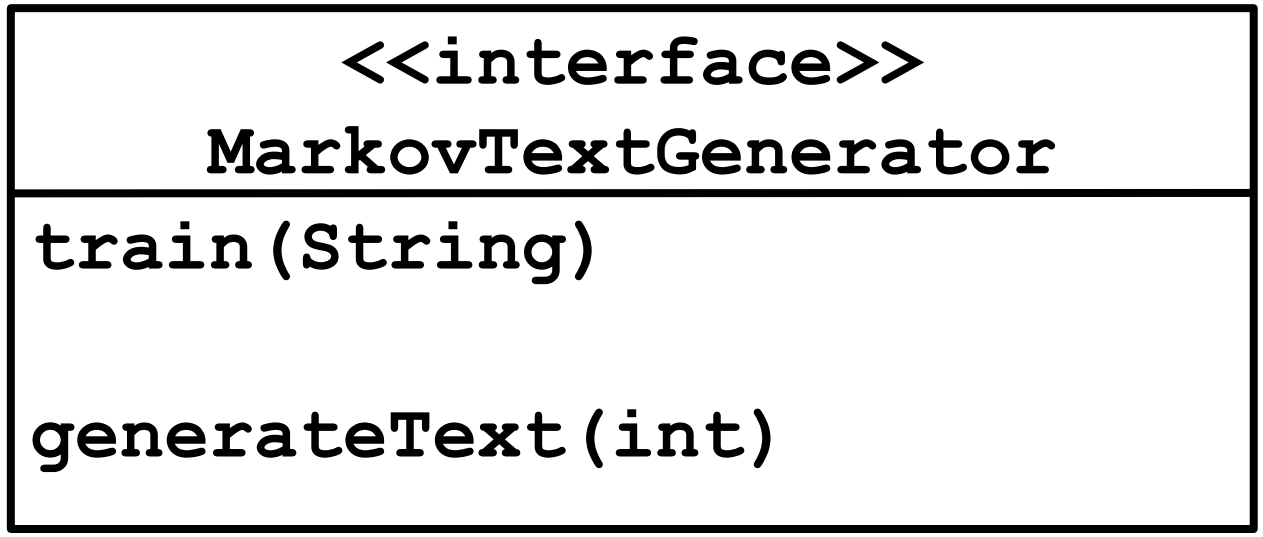
Stage 1: Train

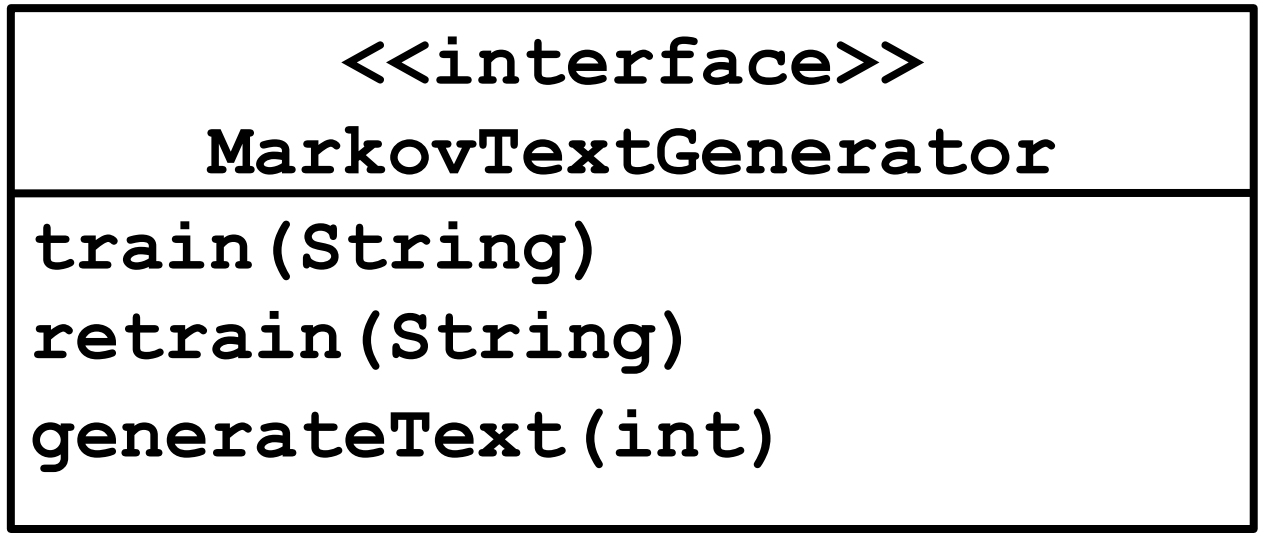
Build model based on data



Stage 2: Generate

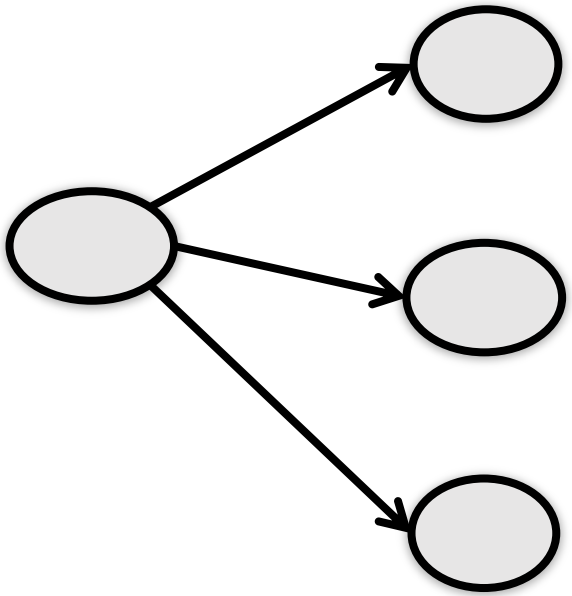
Use model to predict next text


<<interface>>
MarkovTextGenerator
train(String)
generateText(int)


<<interface>>
MarkovTextGenerator
train(String)
retrain(String)
generateText(int)

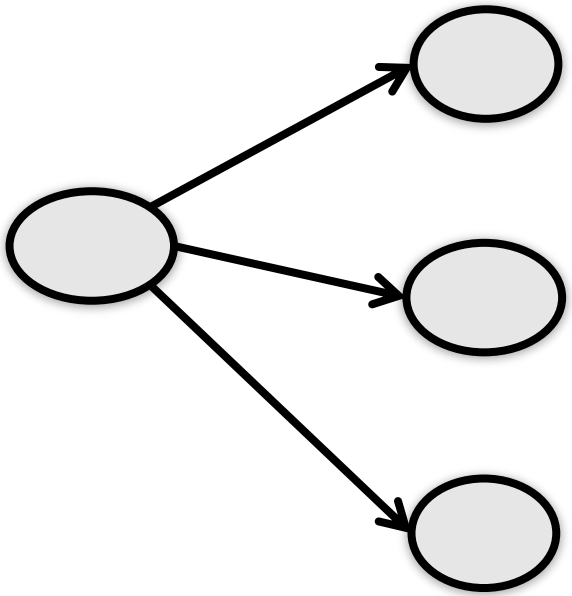
Stage 1: Train

Build model based on ~~data~~ input String



Stage 1: Train

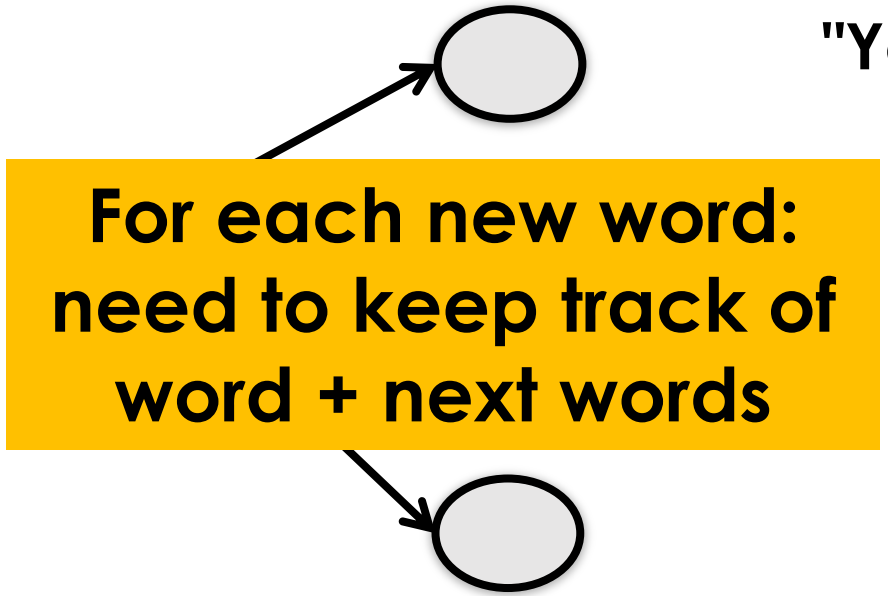
Build model based on ~~data~~ input String



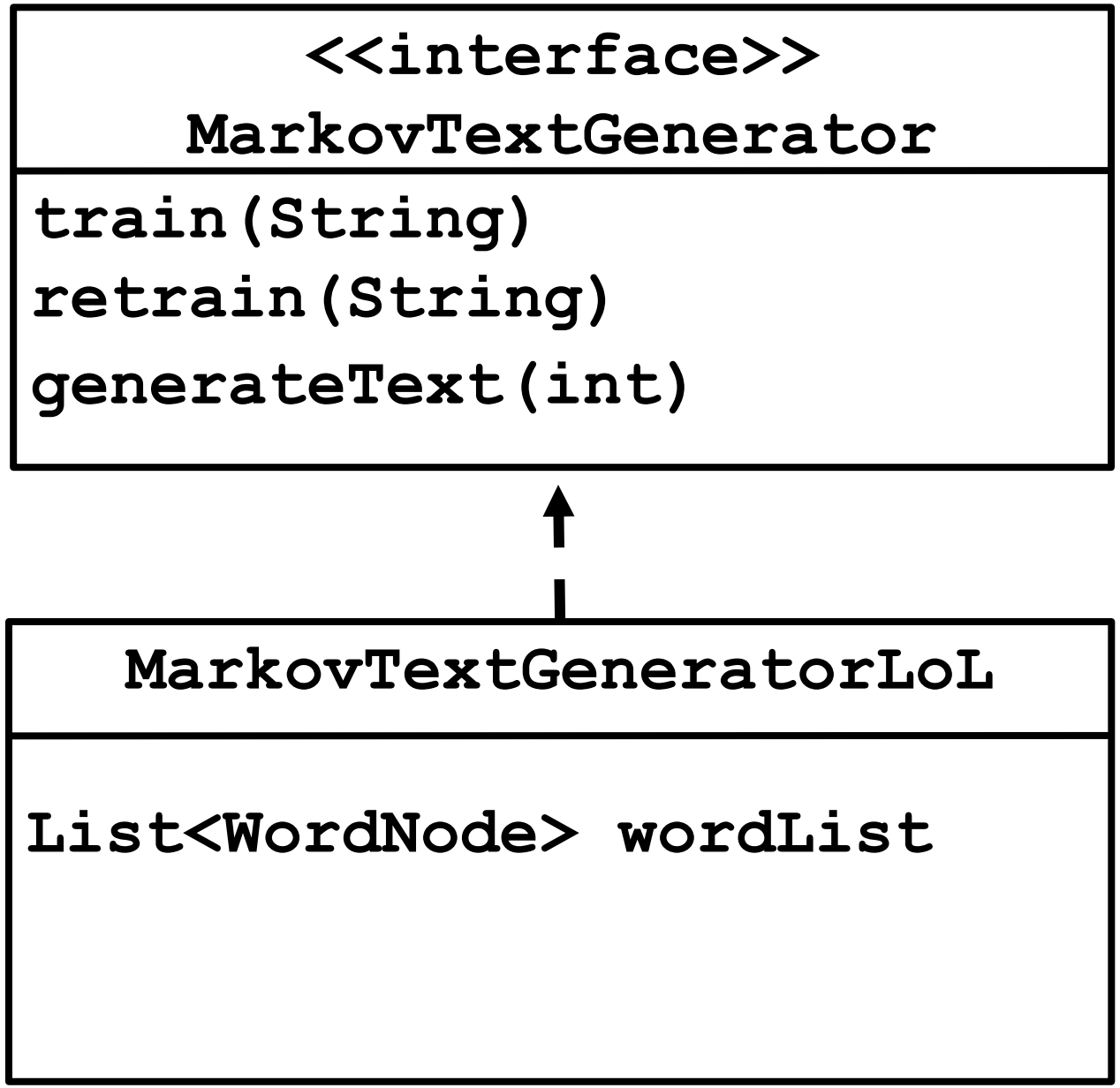
"You say yes, I say no, You say …"

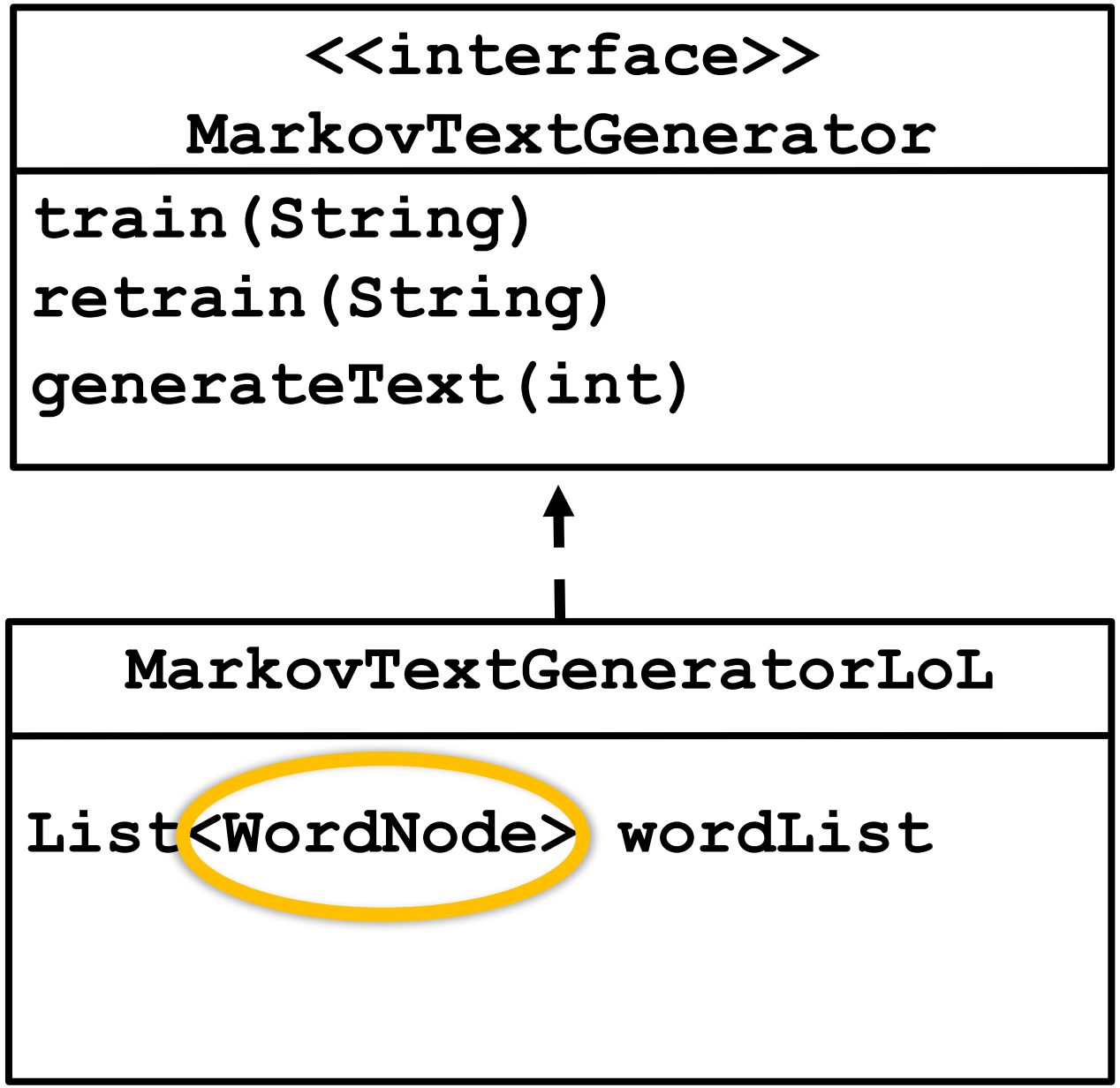
Stage 1: Train

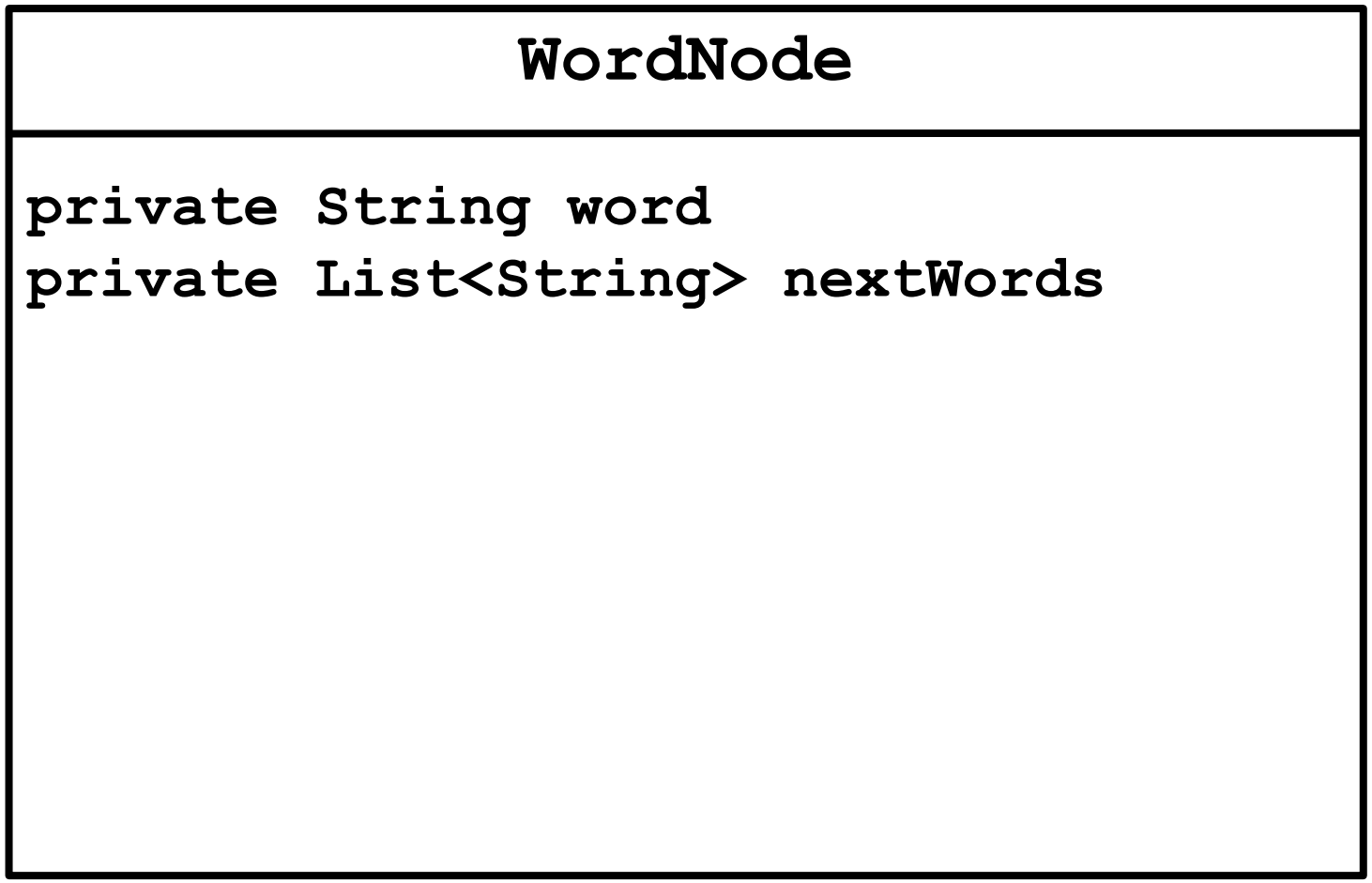
Build model based on ~~data~~ input String

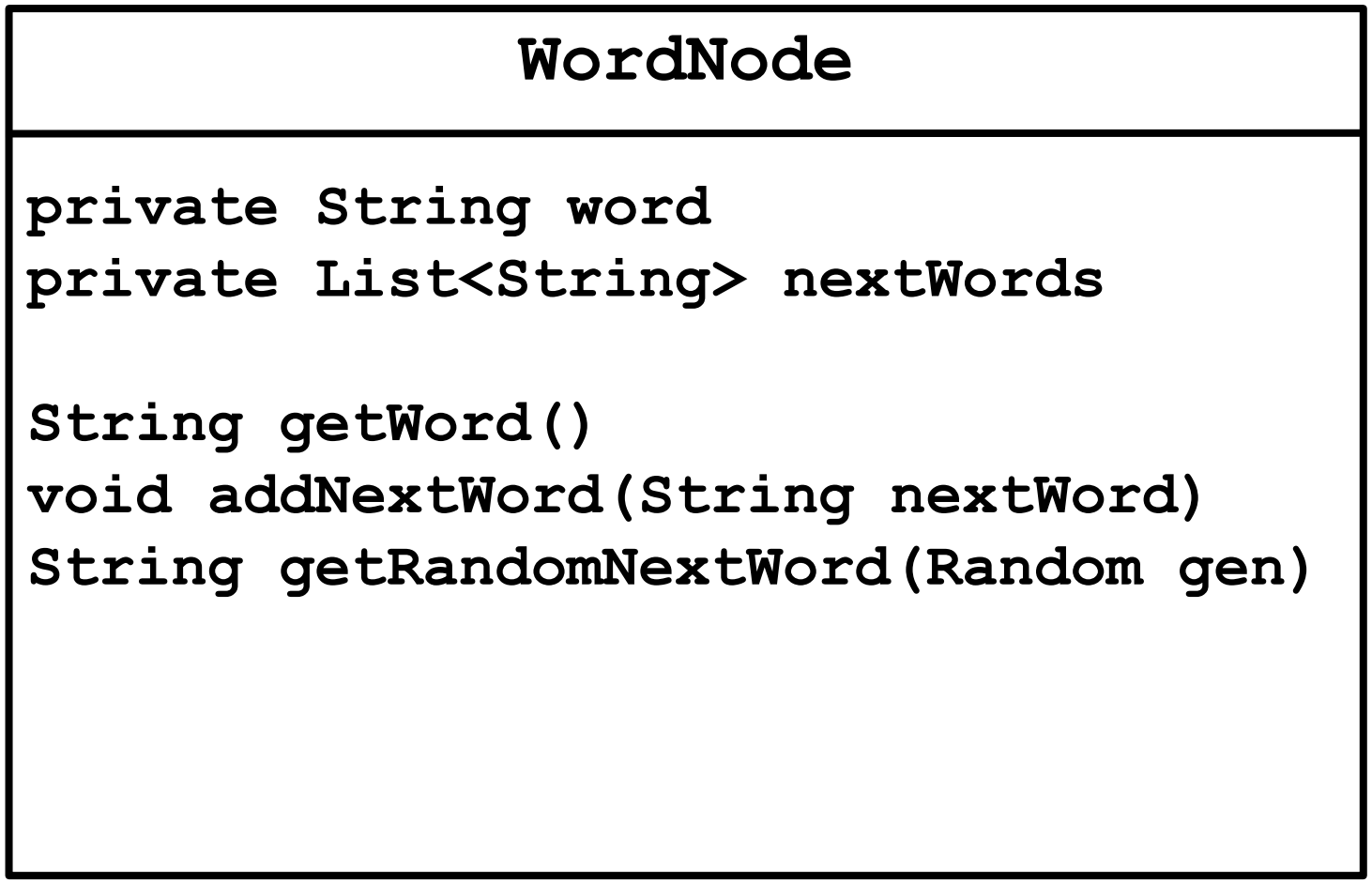


"You say yes, I say no, You say …"


<<interface>>
MarkovTextGenerator
train(String)
retrain(String)
generateText(int)
MarkovTextGeneratorLoL
List<WordNode> wordList

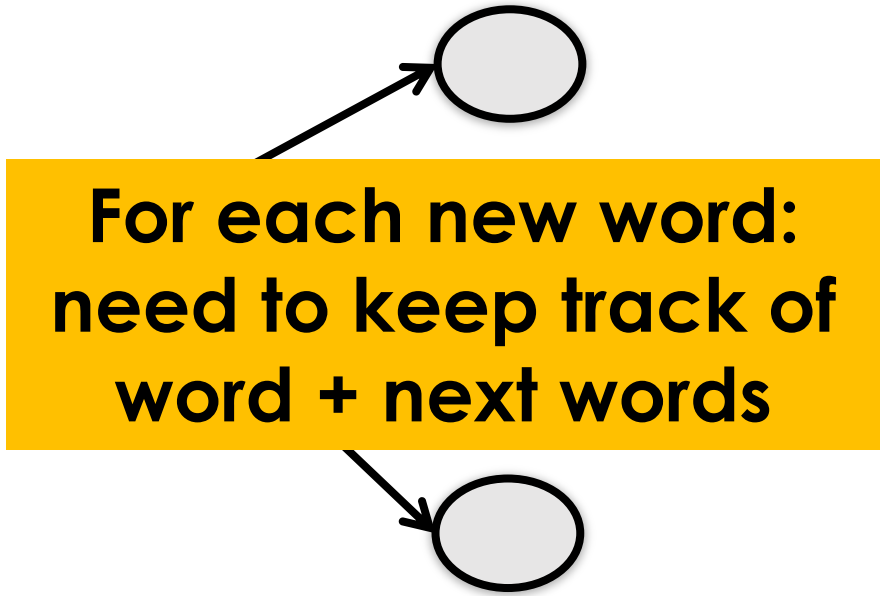

<<interface>>
MarkovTextGenerator
train(String)
retrain(String)
generateText(int)
MarkovTextGeneratorLoL
List<WordNode> wordList


WordNode
private String word
private List<String> nextWords


WordNode
private String word
private List<String> nextWords
String getWord()
void addNextWord(String nextWord)
String getRandomNextWord(Random gen)

Stage 1: Train

"You"	"say"	"yes,"	"I"	"say"	"no,"	"You"	"say"



Stage 1: Train

For each new word: need to keep track of word + next words

"You"	"say"	"yes,"	"I"	"say"	"no,"	"You"	"say"

`word` → "You"

`nextWords` → "say"

Stage 1: Train

For each new word: need to keep track of word + next words

"You"	"say"	"yes,"	"I"	"say"	"no,"	"You"	"say"

`word` → "You"

`nextWords` → "say"

`word` → "say"

`nextWords` → "yes,"

Stage 1: Train

For each new word: need to keep track of word + next words

"You"	"say"	"yes,"	"I"	"say"	"no,"	"You"	"say"

`word` → "You"

`nextWords` → "say"

`word` → "say"

`nextWords` → "yes,"

...

Stage 1: Train

For each new word: need to keep track of word + next words

"You"	"say"	"yes,"	"I"	"say"	"no,"	"You"	"say"

`word` → "You"

`nextWords` → "say"

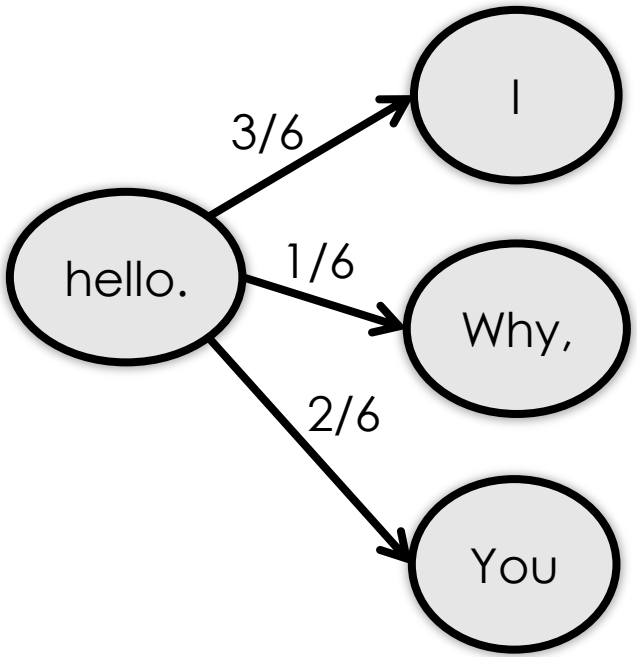
`word` → "say"

`nextWords` → "yes,", "no," …

Stage 1: Train

What about probabilities?

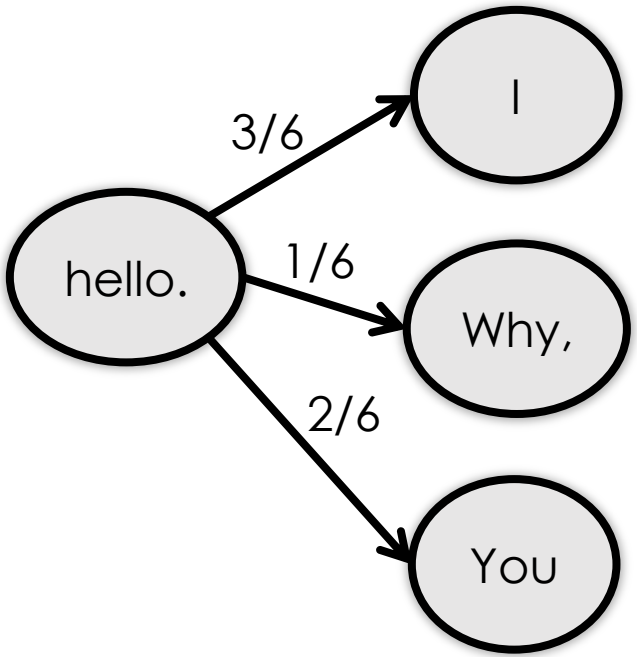
"I don't know why you say goodbye, I say hello. I say high, you say low, …



Stage 1: Train

What about probabilities?

"I don't know why you say goodbye, I say hello. I say high, you say low, …



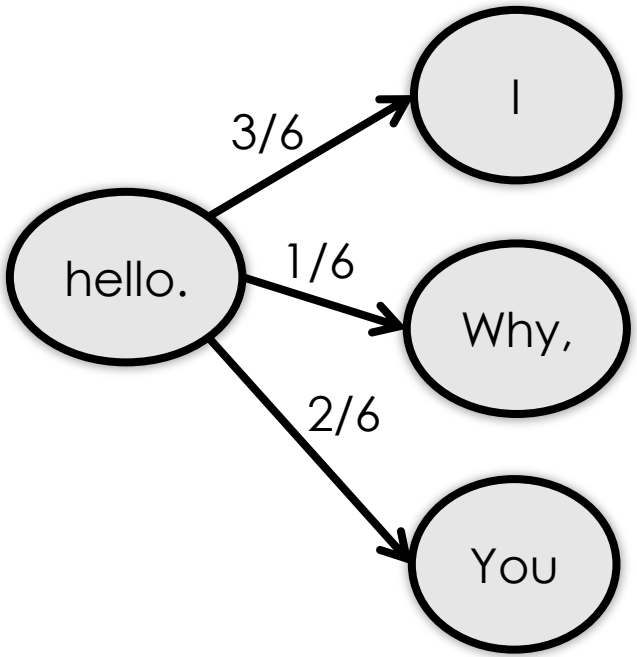

```
word      ──→  "hello."

nextWords →    "I"
```

Stage 1: Train

What about probabilities?

… I say hello, hello, hello. I don't know why you say goodbye …



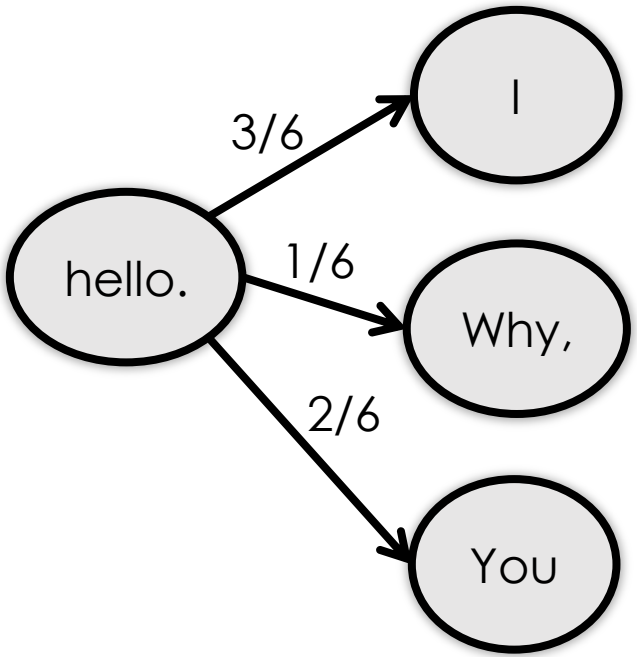

```
word      ⟶  "hello."

nextWords ⟶  "I" , "I"
```

Stage 1: Train

What about probabilities?

… I say hello. Why, why, why, why, why, why, …

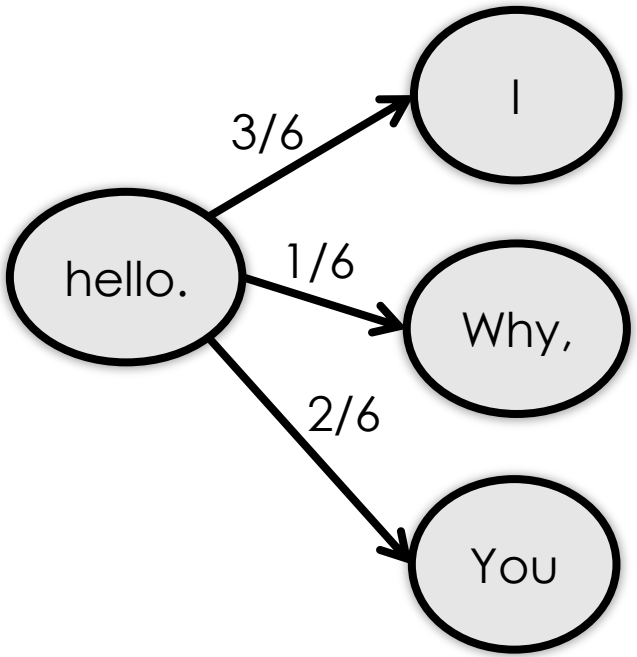



```
word      ⟶  "hello."
nextWords ⟶  "I" , "I" , "Why,"
```

Stage 1: Train

What about probabilities?

… hello, hello, hello. I don't know …



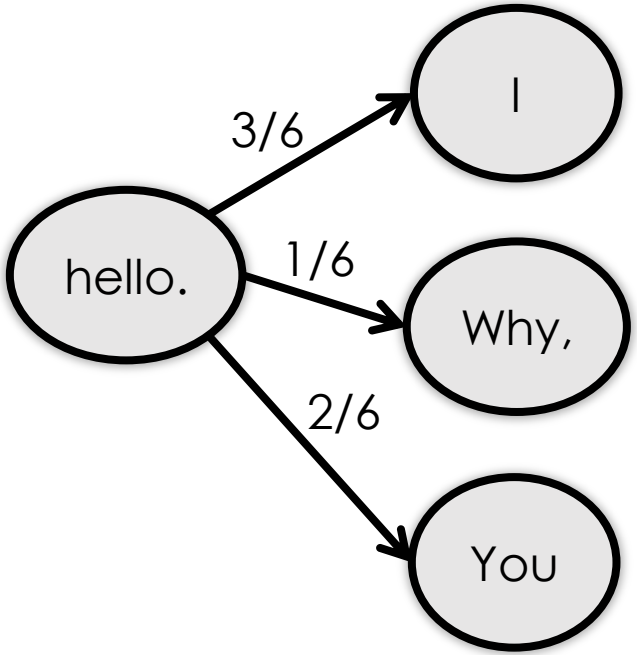
`word` → "hello."

`nextWords` → "I" , "I" , "Why," , "I"

Stage 1: Train

What about probabilities?

... I say hello. You say yes, I say no, ...



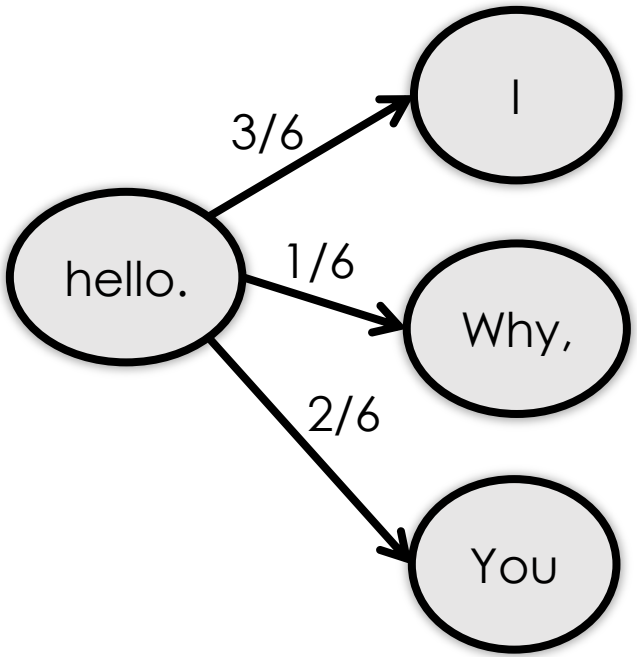

```
word ──────→  "hello."

nextWords →  "I" , "I" , "Why," , "I" , "You"
```

Stage 1: Train

What about probabilities?

etc.

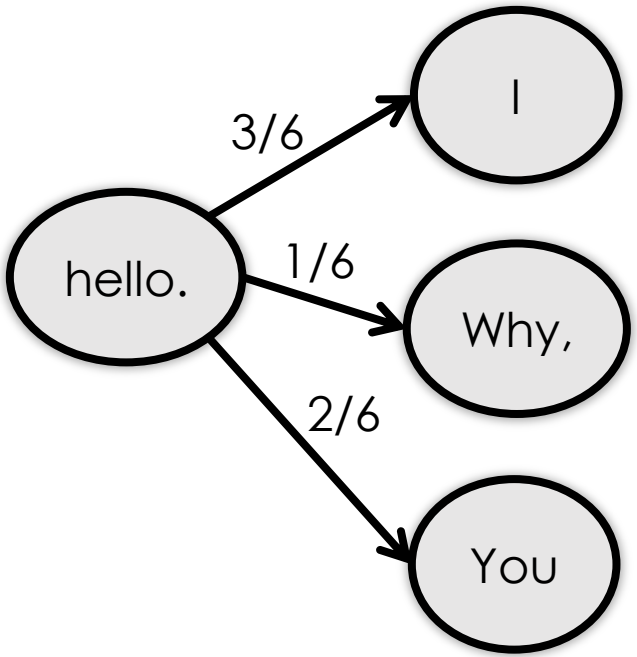



```
word      ──────→  "hello."

nextWords →  "I" , "I" , "Why," , "I" , "You" , "You"
```

Stage 2: Generate

What about probabilities?

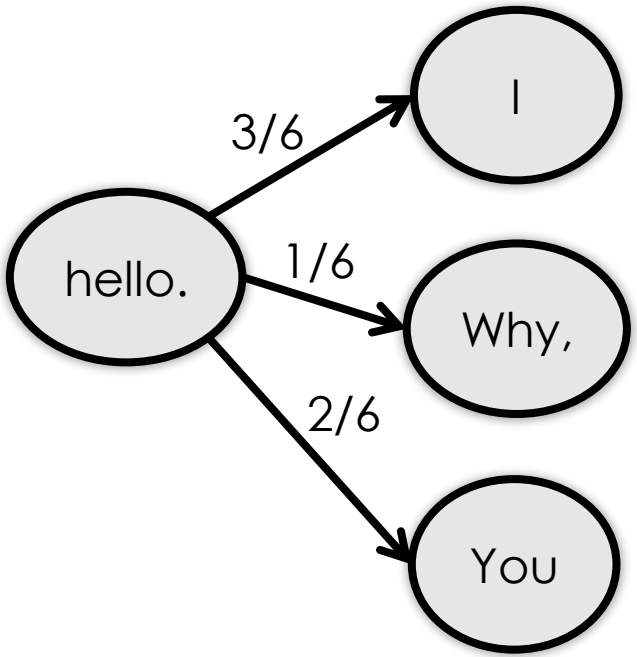



```
word      ──→  "hello."

nextWords →  "I" , "I" , "Why," , "I" , "You" , "You"
```

Stage 2: Generate

What about probabilities?




```
word      ──→  "hello."
nextWords ──→  "I" , "I" , "Why," , "I" , "You" , "You"
```

Pick randomly!

Stage 2: Generate

Until we have enough words:

- Find currentWord as the word of some node in wordList
- Generate a random number between 0 and the size of nextWords list of this node
- Print the word at that index,
- Repeat