Ling 555 — Programming for Linguists

Python — Magic methods and iterators

Robert Albert Felty

Speech Research Laboratory
Indiana University

Nov. 3, 2008

Prereqs

While we are waiting

Please download the following files from:
http://robfelty.com/teaching/L555Fall2008/
resources/py/
birdMagic.py, lexiconDict.py,fibs.py

Prereqs

While we are waiting

Please download the following files from:
http://robfelty.com/teaching/L555Fall2008/
resources/py/
birdMagic.py, lexiconDict.py,fibs.py

```
for file in
{lexiconDict.py,birdMagic.py,fibs.py}; do curl
-o $file
http://robfelty.com/teaching/L555Fall2008/
resources/py/$file; done
```

Outline

1 Homework questions and comments
   - Type-token
   - questions

2 Magic methods and iterators
   - init
   - super
   - Subclassing list, dict, etc. directly

3 Iterators

4 Resources

Type token ratio
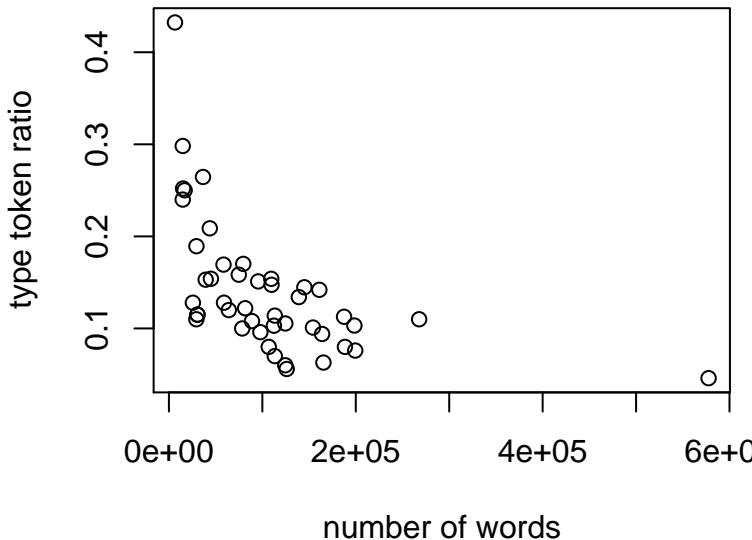
4

L555
Nov. 5

homework

Type-token

questions

magic

Iterators

Resources

4

Type token ratio

r=−.78, t= −8.392, p<.0001

type token ratio

log number of words
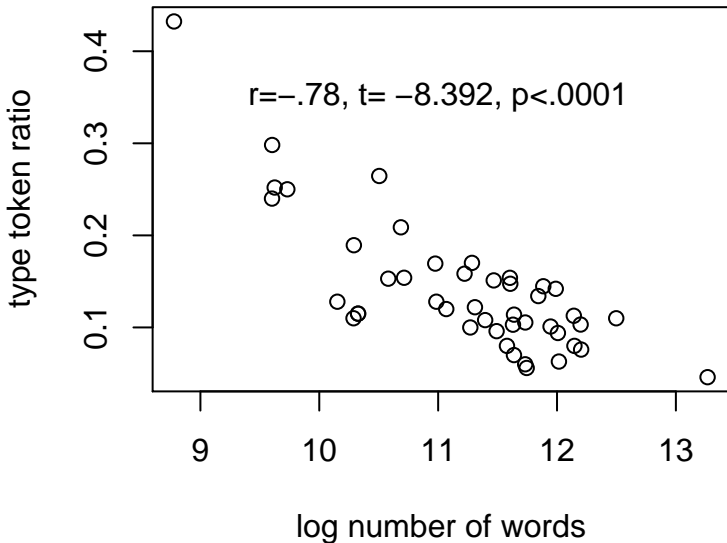
Any questions on homework 8?

Contructors — Init

- In our bird example, we created a method called init(), which we always called right after creating a bird object.
- Instead of doing this, we can use a magic method __init__, which is automatically called when we create an object.
- Let's look at birdMagic.py

New-style classes and super

- Python added new functionality for classes in version 2.2.
- So far, we have been using old-style classes.
- You can mix new and old-style classes, but it is probably better to stick with one.
- New style classes always sub-class another class. To create one from scratch, sub-class *object*

super

When we create a subclass, we want to make sure that the superclass gets initialized. We can do this directly, or we can use super. The advantage of super is that it will call the __init__ methods on all superclasses of a subclass.

Subclassing list, dict, etc.

- With new-style classes, we can also subclass built-in types directly.
- For example, our lexicon class is basically a dictionary with some extra functionality.
- Let's look at lexiconDict.py

Iterators

Definition

An iterator is any object which implements the next method. They can looped over like lists and dicts.

Example

Let's look at using iterators to generate fibonnaci numbers, in fibs.py

Resources for you:

Some of the examples we have covered in class today
can be found on the website at:
http://robfelty.com/teaching/L555Fall2008/resources/py
The files from today are:

1. birdMagic.py
2. lexiconDict.py
3. fibs.py