Bot Literacy
Teaching Librarians to Make Twitter Bots

Mark Eaton, Kingsborough Community College, CUNY
Robin Camille Davis, John Jay College of Criminal Justice, CUNY

Computers in Libraries Conference • March 27, 2019
Today’s talk

● What are bots?
● What is code literacy?
● A pedagogical approach to code familiarity
● Takeaways
What are bots?
What’s a bot?

A little computer program that does the same thing over and over again, with some pre-programmed variation.

A Twitter bot usually sends out a thematic tweet on a regular basis. The tweet is created using a script (code) and some data (text).
@everyword
zucchinis

@everyword zucchinis · 6 Jun 2014
6 273 159

@everyword zucchini
11 1.1K 605

@everyword zucchettos
6 70 51

@everyword zucchetto
4 63 45

@everyword zoysias
4 36 39

@everyword
a strange voyage @str_voyage · 19h
The air feels warmer here. A cave.

a strange voyage @str_voyage · 20h
The blue here is thick with dead eels. A bad omen. The raft sails onward. Sheer crags.

a strange voyage @str_voyage · 20h
We consider our rations; plenty of dark green nuts and eelss.

a strange voyage @str_voyage · 21h
Shivering in the cold rain and the children hang on tight. Intimidating pillars.

a strange voyage @str_voyage · 21h
Something screeching in the dark. We fasten our belongings to the deck. We drift past a foreboding fortress.

@str_voyage
EmojiTetra @EmojiTetra • 1h

Next

Score
550

2% □ Left

90% ▸ Right

8% ◯ Rotate

0% ✈ Down

307 votes • Final results

@EmojiTetra
This user account is a bot that uses AutoWikiBrowser, operated by GreenC (talk). It is used to make repetitive automated or semi-automated edits that would be extremely tedious to do manually, in accordance to the bot policy. The bot is approved and currently active – the relevant request for approval can be seen here.

To stop this bot until restarted by the bot's owner, edit its talk page. If that page is a redirect, edit that original redirecting page, not the target of the redirect.
This tiny printer checks the weather and prints out a forecast each morning

Source: Blogmywiki
On Twitter, as on Facebook, Russian fingerprints are on hundreds or thousands of fake accounts that regularly posted anti-Clinton messages. Many were automated Twitter accounts, called bots, that sometimes fired off identical messages seconds apart — and in the exact alphabetical order of their made-up names, according to the FireEye researchers. On Election Day, for instance, they found that one group of Twitter bots sent out the hashtag #WarAgainstDemocrats more than 1,700 times.
2015–18: Build Your Own Twitter Bot Workshop (A Gentle & Fun Introduction to Python)
Led by Robin Davis & Mark Eaton

In this workshop, participants were introduced to the Python programming language with a “bot starter kit” provided by workshop leaders. Participants customized their whimsical bots and sent generated tweets out into the world. Along the way, participants became familiar with Python through the tried-and-true “change stuff and see what happens” pedagogical method.

- **2015**: LACUNY Emerging Technologies Committee, NYC
- **2018**: Code4Lib, Washington, DC
- **2018**: LITA pre-conference at ALA Annual, New Orleans
What is bot literacy?
President Obama at an Hour of Code event
There have been lots of claims that coding is a new literacy...


There have been other “new literacies”...
Others have suggested that rather than code being a “new” literacy, literacy itself is changing.
“Today, the definition of literacy has expanded from traditional notions of reading and writing to the ability to learn, comprehend, and interact with technology in a meaningful way.”

Others have made similar claims:


Or maybe programming is *not* a literacy?
Some counter-arguments:


Category fight!
A good approach might be to ask humanistic questions about programming and literacy...
What we want participants to learn

How are bots built?
What tools do I need to learn to make bots?
How do I deploy a bot?
What are the social implications of bot making?
How do we do this in practice?

Learn about APIs
Work with a programming language
Develop practical strategies to solve programming problems
Think about the social context of bot-making
What is an API?

“An application programming interface (API) is a particular set of rules ('code') and specifications that software programs can follow to communicate with each other. It serves as an interface between different software programs and facilitates their interaction, similar to the way the user interface facilitates interaction between humans and computers.”

StackOverflow. 2011. “What exactly is the meaning of an API?”
https://stackoverflow.com/questions/7440379/what-exactly-is-the-meaning-of-an-api
What are some useful APIs for librarians?

- Your catalog or discovery service (Primo, Summon…)
- Social media platforms
- OCLC
- DPLA
- Library of Congress
- LibGuides
- Zotero
- more…
An API call can look like this:

```bash
$ curl --request POST \
  --url https://api.twitter.com/1.1/tweets/search/30day/prod.json \
  --header 'authorization: Bearer iRBkKG5E2XzMDjRf176ZC9Ub0wnz4XsNiRVBChTYbJcE3F' \
  --header 'content-type: application/json' \
  --data '{
    "query":"from:TwitterDev lang:en",
    "maxResults": "100",
    "fromDate":"201811010000",
    "toDate":"201811062359"
  }
```
Or it can look much simpler:

api.update_status(status='Hello Twitter!')
Writing code is all about problem solving!
“Instead of proposing that students learn to code, we propose that they code to learn.”

Learning how to create a lawful evil Twitter bot with @robingram and @ocertatat at #c4l18 Thanks for hosting, @dcpl!
Our pedagogical approach:
Interesting
Quick-start
Scaffolded
“...it is difficult to consistently maintain a state of engagement during computer programming because the experience is punctuated by failure and its resultant negative emotions.”

### 1. Getting Started

<table>
<thead>
<tr>
<th>Task</th>
<th>Duration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installing Python</td>
<td>2m 21s</td>
</tr>
<tr>
<td>Choosing an editor or IDE</td>
<td>2m 55s</td>
</tr>
</tbody>
</table>

### 1. Installing Python

1. Which Python is right for you?
2. Python on Windows
3. Python on Mac OS X
4. Python on Mac OS 9
5. Python on RedHat Linux
6. Python on Debian GNU/Linux
7. Python Installation from Source
8. The Interactive Shell
9. Summary
A better first chapter

Python is a very simple language, and has a very straightforward syntax. It encourages programmers to program without boilerplate (prepared) code. The simplest directive in Python is the "print" directive - it simply prints out a line (and also includes a newline, unlike in C).

To print a string in Python 3, just write:

```python
print("This line will be printed.")
```

Source: LearnPython.org
A typical starter program for learners

Dive Into Python
Python from novice to pro

Chapter 2. Your First Python Program

2.1. Diving in

Here is a complete, working Python program.

It probably makes absolutely no sense to you. Don’t worry about that, because you’re going to dissect it line by line. But read through it first and see what, if anything, you can make of it.

```python
def buildConnectionString(params):
    """Build a connection string from a dictionary of parameters.

    Returns string.""
    return ";".join(["%s=%s" % (k, v) for k, v in params.items()])

if __name__ == "__main__":
    myParams = {"server":"mpilgrim", "database":"master", "uid":"sa", "pwd":"secret"}

    print buildConnectionString(myParams)
```
Day 1, Problem 1 from a beginner Python class I took:
Write a program that computes the future value of a specified amount $n$ invested yearly for $m$ years, given an annual interest rate $r$. Print a table to the screen that displays the results.
A better starter activity

How can I make this?
A better way to start learning code

- Aim for familiarity, not fluency, at first
- Make it easy to begin running code
  - PythonAnywhere
  - Bundle of pre-written scripts
- Make something fun that learners can show off to other people
  - Twitter bots
- Start simple, get complex…
Tip: Provide a glossary!

- Write your own plain-language definitions
- Helpful supplemental material

Glossary of terms

Python for Beginners, LITA Pre-Conference Workshop, ALA AC ’18

API: a tool that lets us use someone else’s platform. Today, we’re using the Twitter API, which lets us interact with our Python scripts instead of from Twitter.com or their API. It needs 4 keys to work. Stands for Applied Programming Interfaces, but you don’t need to know that.

Application: in the context of this workshop, we’re making an application program written in code for Twitter. We’re using the Twitter API in programs we’re creating.

Authorization: a step before we can use the Twitter API, they have to authenticate us to make sure we have the right to access our Twitter accounts. We’ll use keys and tokens to authenticate ourselves.

Bot: in the context of this workshop, a bot is a little computer program that does the same thing over and over again, with some pre-programmed variation. A Twitter bot usually does a rhythmic tweet on a regular basis.

Console: in PythonAnywhere, it’s behind the scenes where you can see the output of your code and important output, such as error messages. Same functionality as the command line and Terminal (on a Mac).

Credentials: a login usually just a username and password. In the context of this workshop, our special developer credentials allow us to use the Twitter API with our own app.

CSV: a file type that is a super-basic spreadsheet. It stands for comma-separated values. Visually, it looks like a bunch of text squished together, but it’s easy for a computer to read. Example of a 3-row, 4-column CSV.

Data: information. In the context of this workshop, the data we’ll be using is in the form of a list, a list-like, an array-like, or a dictionary. We’ll use this data and do fun stuff with it. We might say something like our object contains the data.

Database: a database is a collection of related data. A database can be a spreadsheet, a file, a table, a column, or a row. The different databases make it so that you can’t do stuff like adding a + sign, because what would that be?

Developer: person who writes code (i.e., a programmer). In Twitter, a developer is someone using their API that’s not today!

Directory: folder.

Encoding: which special character dictionary the computer uses to display text properly, so you don’t get weird characters. It doesn’t seem like this should be a huge problem, but it totally can! For example, some encoding systems can recognize “☺” or “😊”.

Function: a task. Sometimes this task requires parameters, or things to input. An example of a function is print(“Hello”), which makes the word “Hello” appear on your console.

GitHub: a company that will store your code online for free. GitHub is a great place to look for cool projects and projects that are open source. We’re using it for this workshop to have one place where you can download the code after today (see bit.ly/1ege-4). See also: repo

IDE: basically a program where you write and test code, it stands for Integrated Development Environment.

Input: a program where you write and test code, it stands for Integrated Development Environment.
Scaffolded learning

Progressively move learners toward comprehension and independent learning
Scaffolded learning, cont.

- Common goal
- Ongoing evaluation + adaptive support
- Dialogues + interactions
- Gradual transfer of responsibility
<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Run the provided bot code and listen to instructors describe what it’s doing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

START
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1.</strong></td>
<td>Run the provided bot code and listen to instructors describe what it’s doing</td>
</tr>
<tr>
<td><strong>2.</strong></td>
<td>Change one thing in the bot code to see how output changes</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>1.</strong></td>
<td>Run the provided bot code and listen to instructors describe what it’s doing</td>
</tr>
<tr>
<td><strong>2.</strong></td>
<td>Change one thing in the bot code to see how output changes</td>
</tr>
<tr>
<td><strong>3.</strong></td>
<td>Play with the bot code to make it do something new</td>
</tr>
</tbody>
</table>

**START**
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Run the provided bot code and listen to instructors describe what it’s doing</td>
</tr>
<tr>
<td>2.</td>
<td>Change one thing in the bot code to see how output changes</td>
</tr>
<tr>
<td>3.</td>
<td>Play with the bot code to make it do something new</td>
</tr>
<tr>
<td>4.</td>
<td>Write new bot code</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>1.</td>
<td>Run the provided bot code and listen to instructors describe what it’s doing</td>
</tr>
<tr>
<td>2.</td>
<td>Change one thing in the bot code to see how output changes</td>
</tr>
<tr>
<td>3.</td>
<td>Play with the bot code to make it do something new</td>
</tr>
<tr>
<td>4.</td>
<td>Write new bot code</td>
</tr>
<tr>
<td>5.</td>
<td>Write code for a non-bot task</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>5.</td>
<td>Write code for a non-bot task</td>
</tr>
<tr>
<td>4.</td>
<td>Write new bot code</td>
</tr>
<tr>
<td>3.</td>
<td>Play with the bot code to make it do something new</td>
</tr>
<tr>
<td>2.</td>
<td>Change one thing in the bot code to see how output changes</td>
</tr>
<tr>
<td>1.</td>
<td>Run the provided bot code and listen to instructors describe what it’s doing</td>
</tr>
</tbody>
</table>
The “change stuff and see what happens” approach

```python
# Housekeeping: do not edit
import tweepy
from credentials import *
auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
auth.set_access_token(ACCESS_TOKEN, ACCESS_SECRET)
api = tweepy.API(auth)

# What the bot will tweet
tweet = 'Hello ALA!'

# Send out the tweet and print it to the console
api.update_status(status=tweet)
print(tweet)
print("All done!")
```

RobinAtALA @robin_alal · 22 Jun 2018
Hello ALA!

RobinAtALA @robin_alal · 22 Jun 2018
Hello ALA. 😊

RobinAtALA @robin_alal · 22 Jun 2018
Hello ALA Again. 😊
Sends 1 tweet
Learner tinkers with a **string** and sees what an **API** can do

Sends 3 tweets
Learner tinkers with a **list** and a **for** loop

Sends 5 tweets from a text file
Learner tinkers with **external data**

Tweets randomly-selected rows from a spreadsheet
Learner tinkers with a **while** loop, **CSV** format

See the code yourself at [bit.ly/makeabot](http://bit.ly/makeabot)
Splices together ALA talk titles

Learner tinkers with **data** files and **string operations**

Tweets whenever another account tweets

Learner tinkers with **functions** from the Tweepy library

Generates new “This Is Just To Say” poem by substituting in random objects

Learner tinkers with **data** files and **JSON** format

See the code yourself at [bit.ly/makeabot](http://bit.ly/makeabot)
bot_u1f47e @u1f47e · 13 Feb 2018
of the earth. This was the latter! Here was a blatherskite.

Code4Lib Bots Retweeted

pwooldfn @closertodevnull · 13 Feb 2018
but I begin to get even? Yes, you it.

Code4Lib Bots Retweeted

Jesse @practiceecbot · 13 Feb 2018
YOU BET! As a limbuss, I have often surveyed!

Code4Lib Bots Retweeted

KellyBot @KellyBot8 · 13 Feb 2018
I have eaten the telomians that were in the BILLIARD ROOM

and which you were probably saving for breakfast

Forgive me they were delicious so tan and so DELUDED

Code4Lib Bots Retweeted

CodeForLib @C0D34L1B · 15 Feb 2018
Your next Code4Lib talk... "To the defeat of the organization without patron"

Code4Lib Bots Retweeted

CodeForLib @C0D34L1B · 13 Feb 2018
I have taken the digital currencies that were in the anatomical theatre

and which you were probably saving for Code4Lib

Forgive me they were useful so starlight and so instructed

Code4Lib Bots Retweeted

bot_u1f47e @u1f47e · 13 Feb 2018
had nothing but rabbits. It was a new the.
bot_u1f47e @u1f47e · 13 Feb 2018
of the earth. this was the latter! here was a blatherskite.

Code4Lib Bots Retweeted

pwooldfn @closetodevnull · 13 Feb 2018
but i begin to get even? yes, you it.

Code4Lib Bots Retweeted

Jesse @practiceebot · 13 Feb 2018
YOU BET! As a limbuss, I have often surveyed!

Code4Lib Bots Retweeted

KellyBot @KellyBot8 · 13 Feb 2018
I have eaten
the telomians
that were in
the BILLIARD ROOM
and which
you were probably
saving
for breakfast

Forgive me
they were delicious
so tan
and so DELUDED

Code4Lib Bots Retweeted

Code4Lib @C0D34L1B · 15 Feb 2018
Your next Code4Lib talk... "To the defeat of the organization without patron"

Code4Lib Bots Retweeted

Code4Lib @C0D34L1B
I have taken
digital currencies
that were in
the anatomical theatre

and which
you were probably
saving
for Code4Lib

Forgive me
they were useful
so starlight
and so instructed

Code4Lib Bots Retweeted

bot_u1f47e @u1f47e · 13 Feb 2018
had nothing but rabbits. it was a new the.
Feedback from workshops

“I really enjoyed this. … I feel more confident about continuing on with learning Python than times I’ve previously taken a crack at it.” —Code4Lib participant

“This session was a great blend of hands on, theory, and information on practical uses of Python.” —ALA/LITA participant

“I felt the session focused more on bots and sample code than introducing us to the python language. ... That said, I do know enough now to find other resources to learn on my own.” —ALA/LITA participant
Note about new Twitter bot accounts

- Sign-up process for new accounts is now more rigorous
- Not feasible to run this workshop again
- Alternatives
  - Slack bots
  - Mastodon bots
Takeaways
Bot literacy helps us to better understand our networked world
Making bots is an effective entry point into programming
A scaffolded approach is a good way to teach programming
Include creative time!
You can read a step-by-step walkthrough of our workshop in the Journal of Interactive Technology and Pedagogy:

https://jitp.commons.gc.cuny.edu/make-a-twitter-bot-in-python-iterative-code-examples/
And each iteration of the workshop is on GitHub:

https://github.com/robincamille/bot-tutorial (December 2015)
https://github.com/robincamille/bot-tutorial-code4lib (February 2018)
https://github.com/MarkEEaton/bot-tutorial-ala (June 2018)
Thanks to Olivia Jester for all of the adorable public domain robot drawings

publicdomainpictures.net/en/browse-author.php?a=121563
Thank you!

Mark Eaton
● Kingsborough Community College, CUNY
● Mark.Eaton@kbcc.cuny.edu
● Mastodon: @ocertat@mastodon.sdf.org

Robin Camille Davis
● John Jay College of Criminal Justice, CUNY
● RobDavis@jjay.cuny.edu
● Twitter: @robincamille