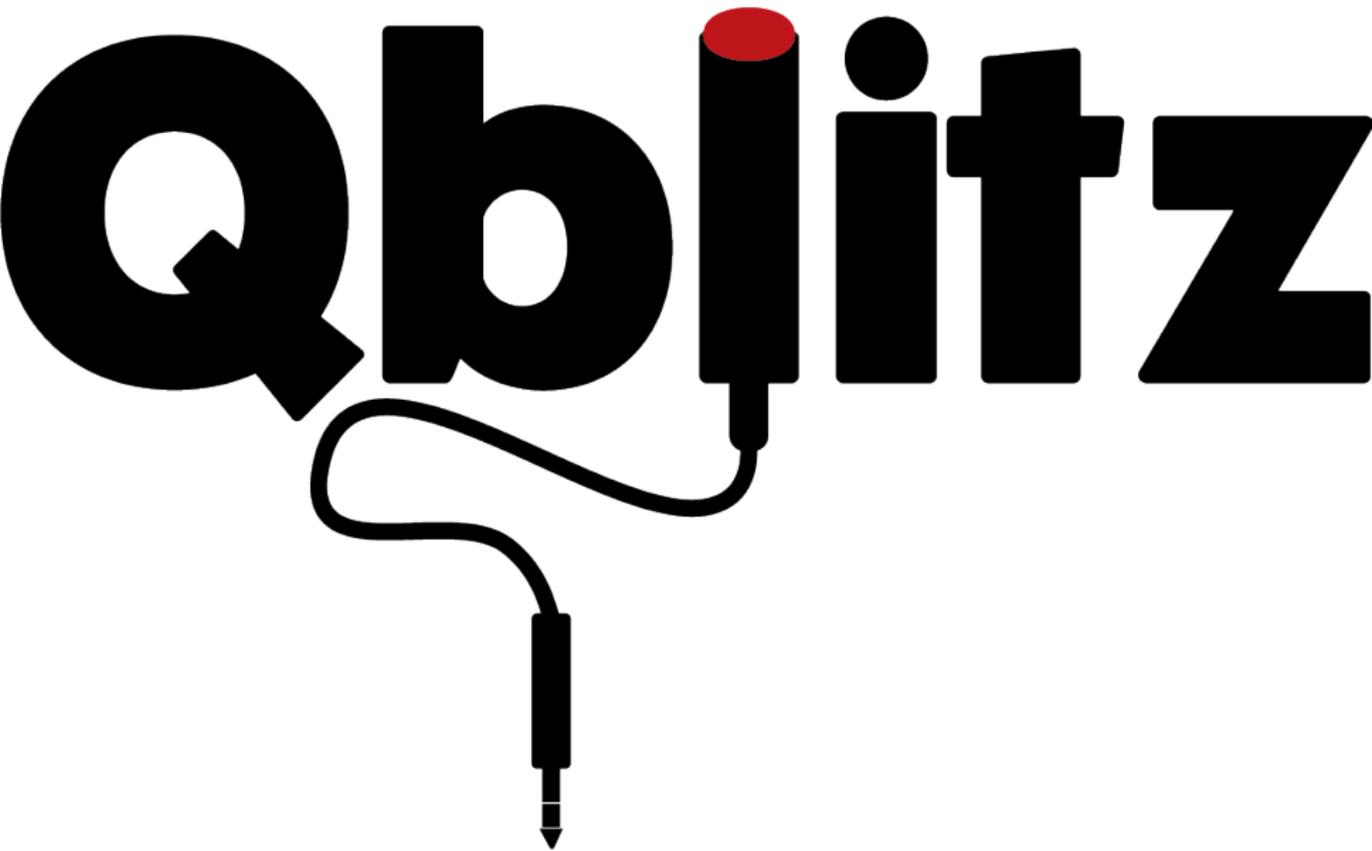




QANTA Competition

Natural Language Processing: Jordan
Boyd-Graber
University of Maryland
FRAMEWORK

Material adapted from Chen Zhao, Pedro Rodriguez, and Shi Feng



Playing Online

How the shared task works



How the shared task works



- I'm User 1. I'd like to play!



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]

- I'm User 1. I'd like to play!



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
- It's Extremism

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
- It's Extremism

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1
- I'd like to hear Word 2 of Question 1



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
- It's Extremism
- It's in

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1
- I'd like to hear Word 2 of Question 1



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
- It's Extremism
- It's in

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1
- I'd like to hear Word 2 of Question 1
- I'd like to hear Word 3 of Question 1



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
- It's Extremism
- It's in
- It's the

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1
- I'd like to hear Word 2 of Question 1
- I'd like to hear Word 3 of Question 1



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
- It's Extremism
- It's in
- It's the

- I'm User 1. I'd like to play!
- I'd like to hear Word 1 of Question 1
- I'd like to hear Word 2 of Question 1
- I'd like to hear Word 3 of Question 1
- I'd like to answer Question 1 with
Barry_Goldwater



How the shared task works



- Hi! Available questions are [1, 2, 3, 4]
 - It's Extremism
 - It's in
 - It's the
 - Got it! You've answered Question 1 at Position 3 with Barry_Goldwater
-
- I'm User 1. I'd like to play!
 - I'd like to hear Word 1 of Question 1
 - I'd like to hear Word 2 of Question 1
 - I'd like to hear Word 3 of Question 1
 - I'd like to answer Question 1 with Barry_Goldwater



Leaderboard

We evaluate each system with four metrics: accuracy at the end of the first sentence (first_acc) and at the end of the question (end_acc), and two new metrics: [expected wins](#) with system buzzer (EW) and with optimal buzzer (EW_OPT). Ranking is decided by EW.

| Rank | Model | first_acc | end_acc | EW | EW_OPT |
|-------------------|--|--------------|--------------|--------------|--------------|
| 1 Dec 10, 2018 | BitER_the_dusT <i>FYY</i> | 0.119 | 0.672 | 0.291 | 0.618 |
| 2 Dec 10, 2018 | SBQA <i>CMSC723 Technical Wizards</i> | 0.104 | 0.559 | 0.271 | 0.589 |
| 3 Dec 10, 2018 | DAN-TFIDF Buzzer <i>CMSC723 ForwardRethinking</i> | 0.0690 | 0.609 | 0.265 | 0.593 |

Computers Can Play Too!

How to do it: Webserver

```
$ http POST http://0.0.0.0:4861/api/1.0/quizbowl/act text='
HTTP/1.0 200 OK
Content-Length: 41
Content-Type: application/json
Date: Wed, 10 Oct 2018 01:12:27 GMT
Server: Werkzeug/0.14.1 Python/3.7.0

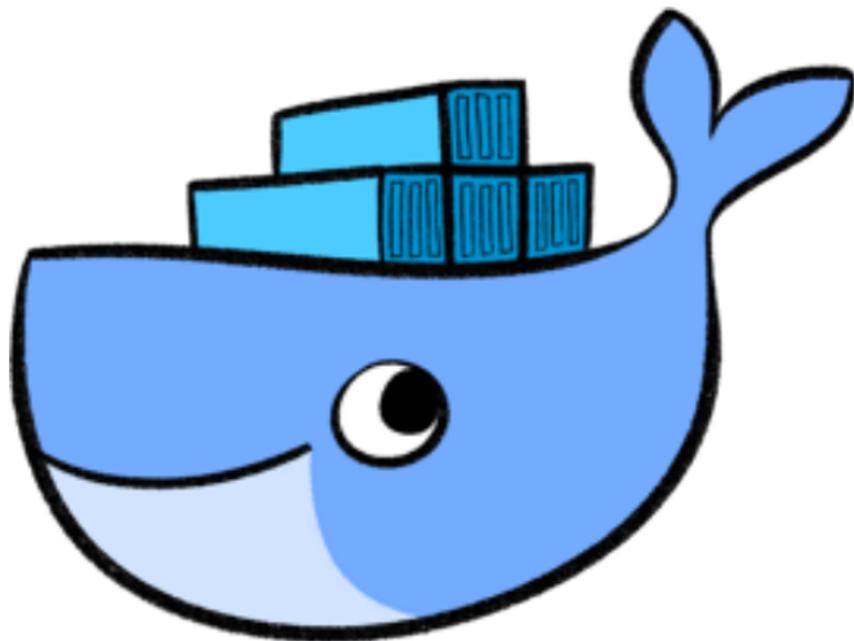
{
    "buzz": false,
    "guess": "Albert_Einstein"
}
```

What should questions look like?

1. **question_idx**: Question number in the current game
2. **char_idx**: How much of the question you've seen
3. **sent_idx**: The current sentence number.
4. **text**: Question text up to char_idx

```
{  
  "question_idx": 0,  
  "char_idx": 112,  
  "sent_idx": 0,  
  "text": "At its premiere, the librettist of this opera  
          portrayed a character who asks for a glass  
          of wine with his dying wish"  
}
```

How to get your system to us?



Docker: creating portable containers for software

How to get your system to us?



Docker: creating portable containers for software

Getting Started

- Make sure you have Python2 (Django) and Python3 (everything else)
- Install Docker (<http://docker.com>) and create account
- Install Codalab command line tools (<https://github.com/codalab/codalab-worksheets/wiki/CLI-Basics>) and create account

Most important piece of code ...

 Pinafore / **qanta-codalab**

[↔ Code](#) [! Issues 1](#) [🔗 Pull requests 0](#) [📁 Projects 0](#) [📖 Wiki](#)

Branch: master ▾ **qanta-codalab / src / qanta / tfidf.py**

 **aagohary** paragraphs in jsonl and downloaded from object store

3 contributors   

165 lines (130 sloc) | 4.92 KB

```
1 from typing import List, Optional, Tuple
2 from collections import defaultdict
3 import pickle
4 import json
5 from os import path
6
```

What it does

- **download**: get necessary data
- **train**: create a model, save in pickle
- **up**: Launch a webserver that can answer the questions
- **run eval**: Run evaluation script on provided test data

Running on Codalab



<http://codalab.qanta.org>

- Create a bundle (in a directory)
- Upload your bundle
- Evaluate it
 - You can specify a docker image
 - And ask for GPU
- Submit to the official competition
- Lets us run your code on new questions

Running on Codalab



<http://codalab.qanta.org>

- Create a bundle (in a directory)
- Upload your bundle
- Evaluate it
 - You can specify a docker image
 - And ask for GPU
- Submit to the official competition
- Lets us run your code on new questions
- More on this later

Things to Remember

- Incorporating new data in simple ways likely better than super-complicated models
- Code won't run on Codalab immediately
- Limits on size of docker container
- Determine limiting reagent: what is holding you back
- Error analysis, not just number