COGS 108 Week 8

Slide Credit: Akshay N (FA22)

Deadlines

DUE DATES

- Quiz 7 is due Monday, May 19
- A3 is due Wednesday, May 21
- Discussion lab 7 is due Friday May 23

COMING UP

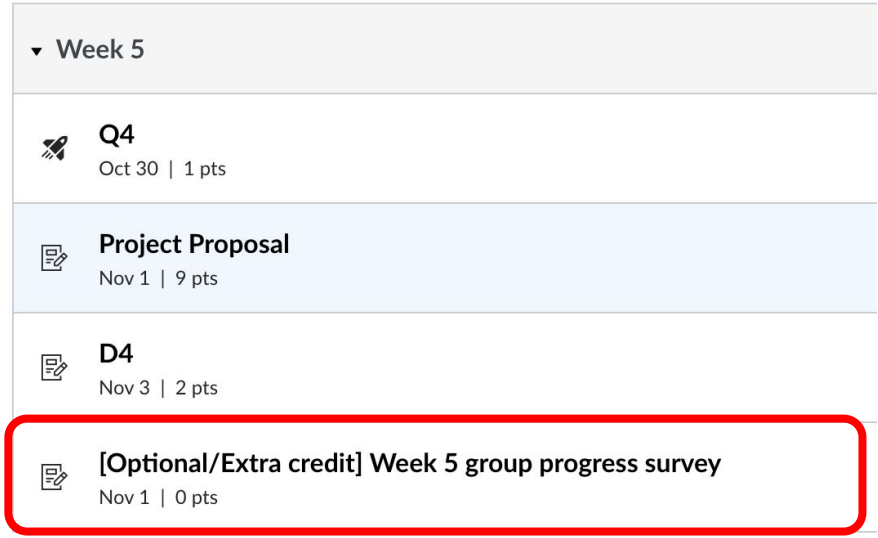
- EDA Checkpoint is due next Wednesday, May 28

Announcements

- Your Data Checkpoints will be graded by Wednesday or Thursday of this week
  - We won't give additional feedback for the proposal stuff, we'll just update your grade in your proposal issue if we see that you fixed it
  - We will give feedback on the Data part of the checkpoint
  - If there are still things you need to fix from the proposal, we'll put those comments into the checkpoint issue

Project - Weekly Check-ins

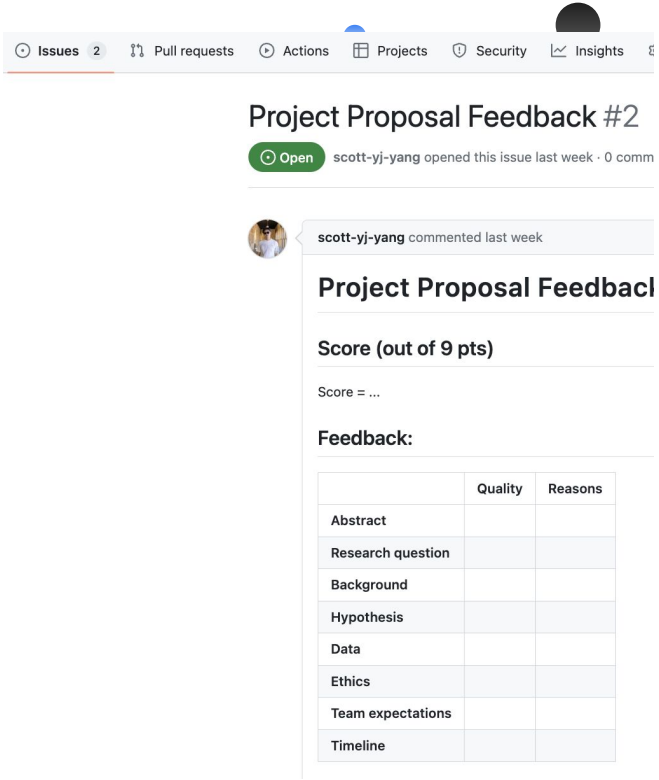
- Every week you can fill out the weekly group progress survey
- If you fill them all out you get **Extra Credit!!!**
- It's a chance for you to let us know how your project is going
  - Questions?
  - Concerns about groupmates?
  - Challenges you're facing



Checkpoint Feedback

- Feedback will be released on Issues!
- There is an issue in your repo with your assigned TA/IA ⇒ Reach out to them with any questions.

- Start thinking about your EDA Checkpoint!!!



Project Updates

- Remember to check Edstem for all updates on projects
- Note: For very large datasets that cannot be uploaded to GitHub, you may use either Git LFS, or just upload to google drive or another dropbox and paste the link.
  - Your TA MUST be able to see the raw data in some format…

DISCUSSION LAB 7

Text Analysis

TF-IDF

- What is TF-IDF?
  - term frequency-inverse document frequency.
  - a statistical measure that evaluates how relevant a word is to a document in a collection of documents.

TF-IDF

- How does it work?
  - Multiplies two terms:
  - how many times a word appears in a document
  - the inverse document frequency of the word across a set of documents.
  - So, if the word is very common and appears in many documents, this number will approach 0. Otherwise, it will approach 1.

TF-IDF

- How does it work?

$$\text{TF-IDF}(t, d, D) = \log(1 + \text{frequency of } t \text{ in } d) \quad \log\left(\frac{\text{Total \# of documents}}{\text{\# of documents that contain the word } t}\right)$$

Term frequency

Inverse document frequency

TF-IDF

- Implementing in python

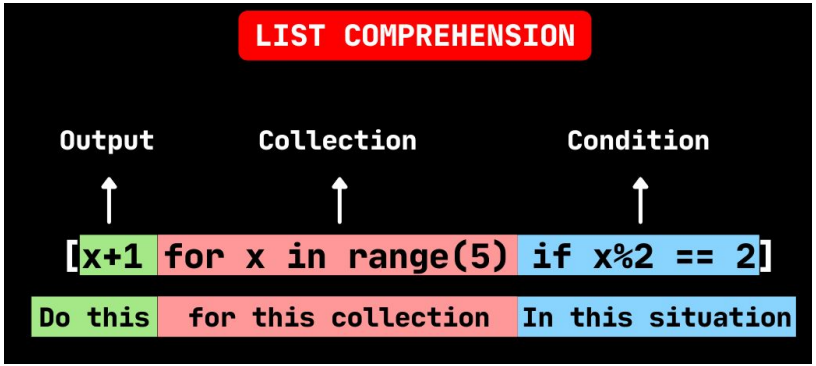
```
tfidf = TfidfVectorizer(sublinear_tf=True,
                        analyzer='word',
                        max_features=2000,
                        tokenizer=word_tokenize,
                        stop_words=stopwords.words("english"))
```

Sublinear_tf: Apply sublinear tf scaling, i.e. replace tf with 1 + log(tf).

max_features: If not None, build a vocabulary that only consider the top max_features ordered by term frequency across the corpus. Otherwise, all features are used.

List Comprehension

- Use List Comprehension in python to get list of items that are a manipulation of another collection
- Ex:
  - years = [____ for speech in inaugural.fileids()]



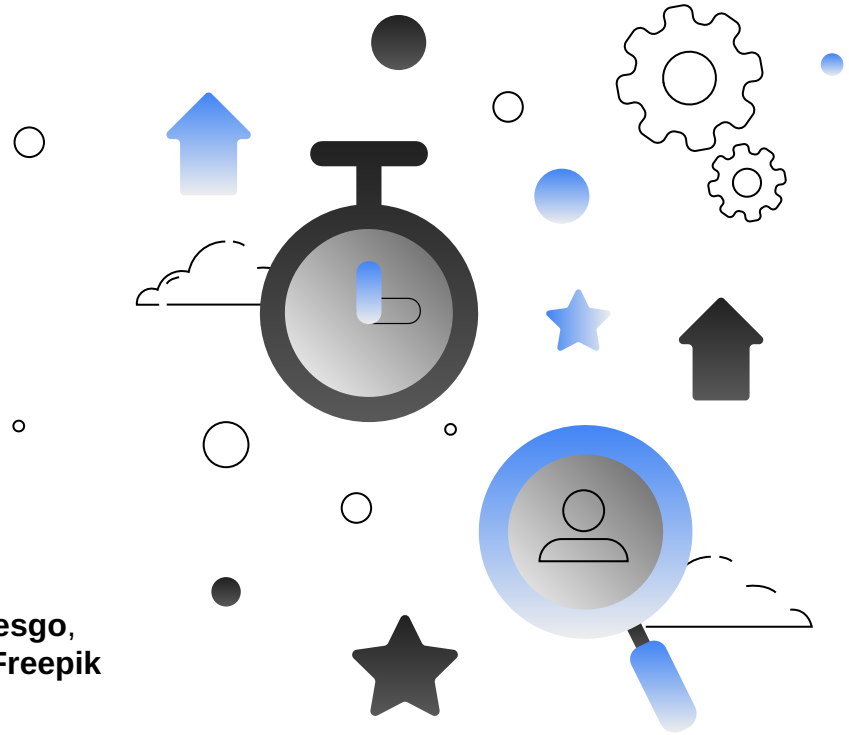
Plotting

- In general, a linear distribution can be plotted using plt.plot()
  - plt.plot(x = ?, y = ?, label = ?)
  - plt.xlabel(?)
  - plt.ylabel(?)
  - plt.legend(?)

- plt.plot(inaug_tfidf.index.to_numpy(), inaug_tfidf['british'].to_numpy())

THANKS!

Questions on Edstem or
Office hours: Friday 8-10 am



CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon** and infographics & images by **Freepik**