AUTOMATING HIV/AIDS GRANT CLASSIFICATION AND INTEGRATING TO FRONT-END APPLICATION

National Institute of Health

Office of AIDS Research | Summer 2022

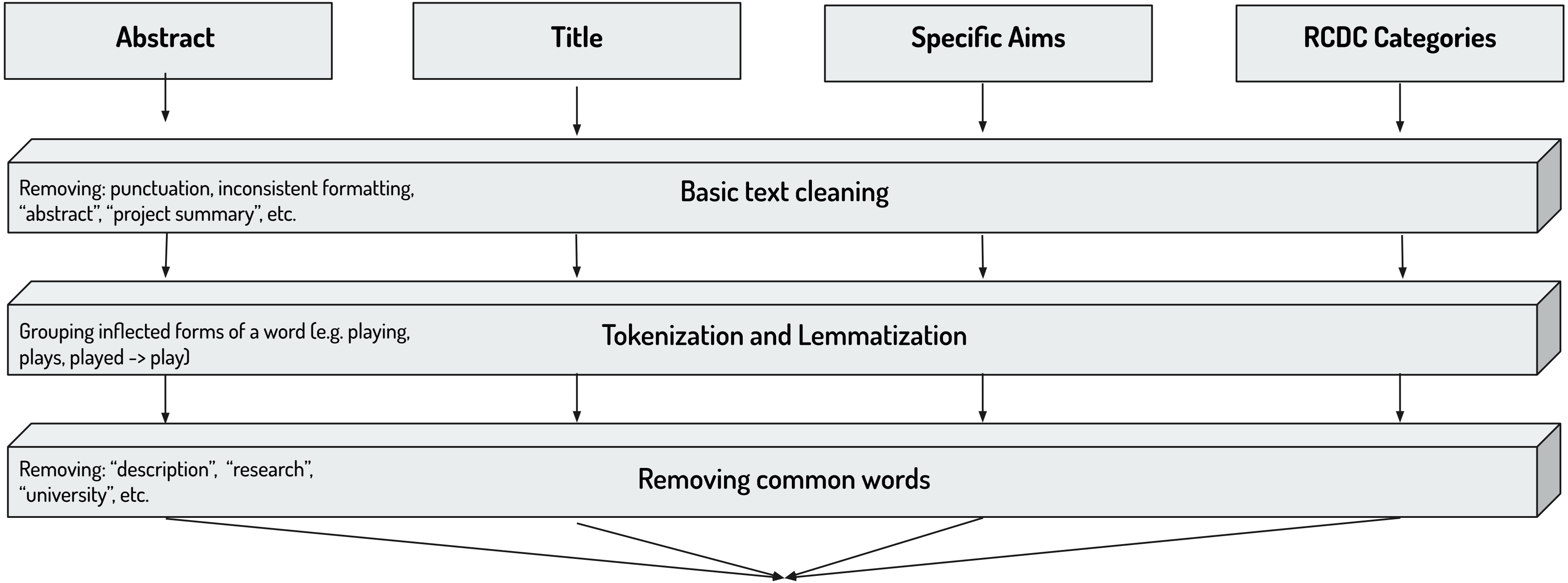
JEREMY LEE
Harvard University
Statistics and Mathematics

WARREN QUAN
Princeton University
Computer Science

Project Overview

- **Project Goal:** Explore if it's possible to build an automated algorithm to encode HIV-related grants under 1 or multiple of the 43 objective codes and 1 of the 9 areas of emphasis
- **Data:**
  - PQS data from 2012 to 2021
  - Abstracts, specific aims, and RCDC categories pulled from QVR
  - Filtered for grants with all funding toward a single objective code
  - Left with 22,823 unique Appl. Ids across all ICs
  - 43 unique objective codes

Text Cleaning



Abstract

Title

Specific Aims

RCDC Categories

Basic text cleaning

Removing: punctuation, inconsistent formatting, “abstract”, “project summary”, etc.

Tokenization and Lemmatization

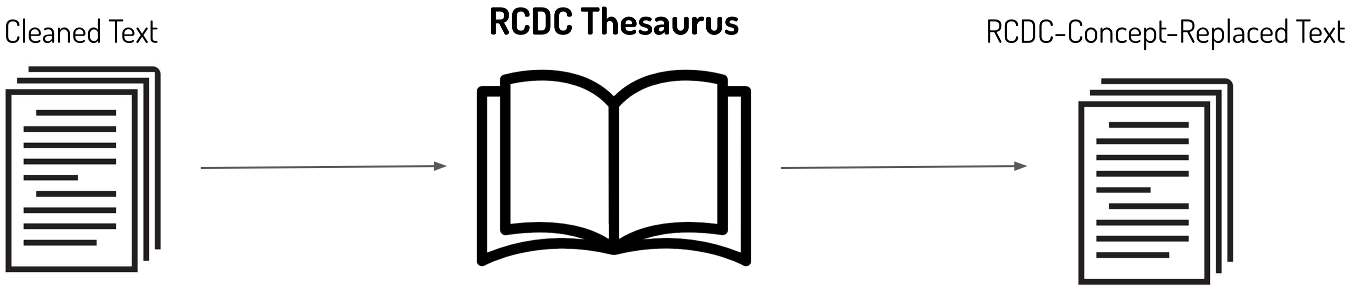
Grouping inflected forms of a word (e.g. playing, plays, played -> play)

Removing common words

Removing: “description”, “research”, “university”, etc.

['urban', 'congregation', 'capacity', 'hiv', 'prevention', 'care', 'religious', 'congregation', 'reach', ...]

RCDC Concept Replacement



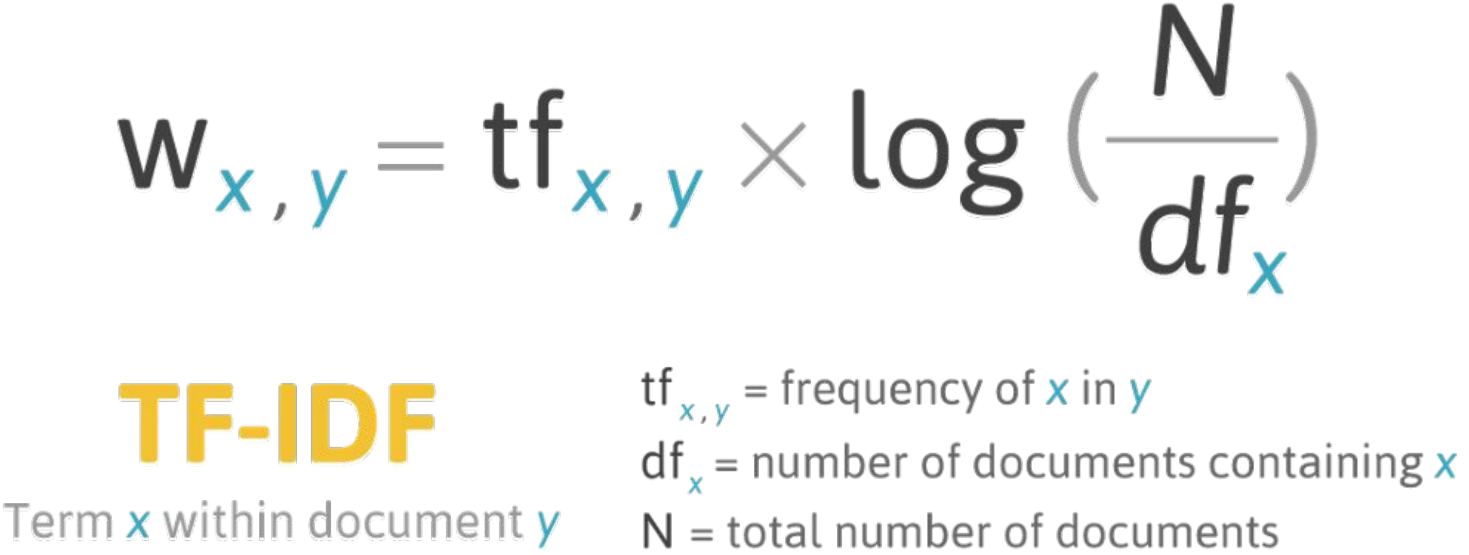
["1 year old", "one year of age", "1 year of age", "aged one year", "aged 1 year", "one year old", "age 1 year"] → 1 year old

["1,2-Dimethylhydrazine", "N,N'-Dimethylhydrazine", "1,2-Dimethyl-hydrazine"] → 1,2-Dimethylhydrazine

["1,3-Butadiene", "divinyl"] → 1,3-Butadiene

["1-Butanol", "n-Butanol", "n-Butyl Alcohol"] → 1-Butanol

Document Embedding



$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF

Term x within document y

$tf_{x,y}$ = frequency of x in y

df_x = number of documents containing x

N = total number of documents

Feature Selection

Performed chi square hypothesis tests with α = 0.05 to determine relevant features

2A: Biology of HIV Acquisition and Transmission, Biology of HIV Transmission

- **Top Features:** transmission, transmission process, penile, vfe, 47, genital system, genital, tf, seminal exosomes, hiv transmission

4A: Adaptive and Innate Host Defense Mechanisms, Host Defense Mechanisms

- **Top Features:** gc, lymphocytes, tfh, antibody, plasma cells, llpc, mbc, brwd1, memory lymphocytes, trm

6E: Approaches to Interrupt Vertical Transmission, Approaches To Interrupt Vertical Transmission and Preserve Maternal Health

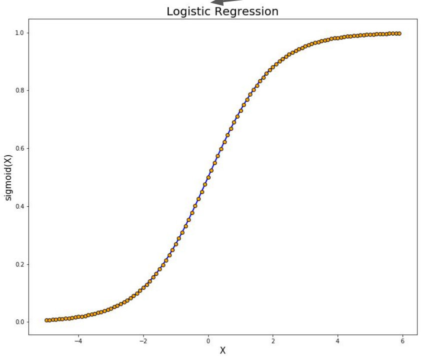
- **Top Features:** cr, hair, mother infant, cape town, town, pmtct, ctu, utero, cape, mother

Voting Classifier

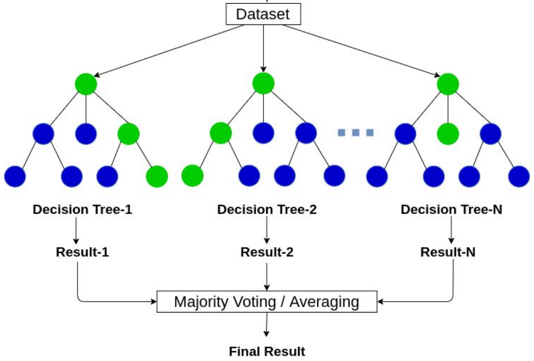
Objective Code Accuracy: ~68%
Area of Emphasis Accuracy: ~85%

3 classification models vote to generate a prediction

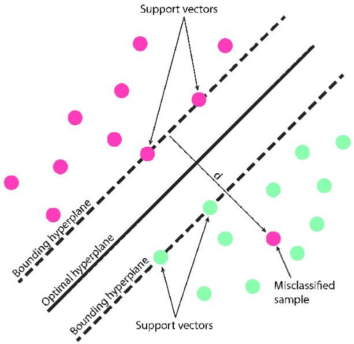
['urban', 'congregation', 'capacity', 'hiv', 'prevention', 'care', 'religious', 'congregation', 'reach', ...]



Logistic Regression



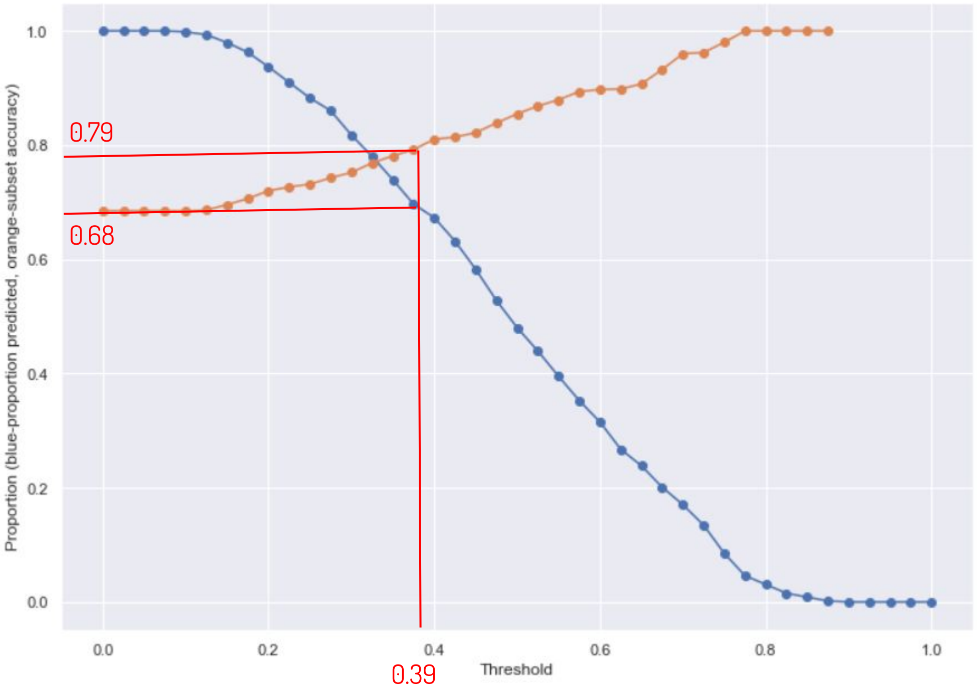
Random Forest



Support Vector Machine

Model Evaluation

Objective Code Accuracy: ~68%
Area of Emphasis Accuracy: ~85%



Interpretation:

- Whenever the model creates a prediction, it outputs a “confidence” along with it from 0 to 1 (e.g. predicts 5A with 0.45 confidence)
- This graph plots the model’s confidence (on the x-axis) against its accuracy (orange line) and the proportion of test samples that the model predicted with at least that level of confidence (blue line)

Example Interpretation (red lines):

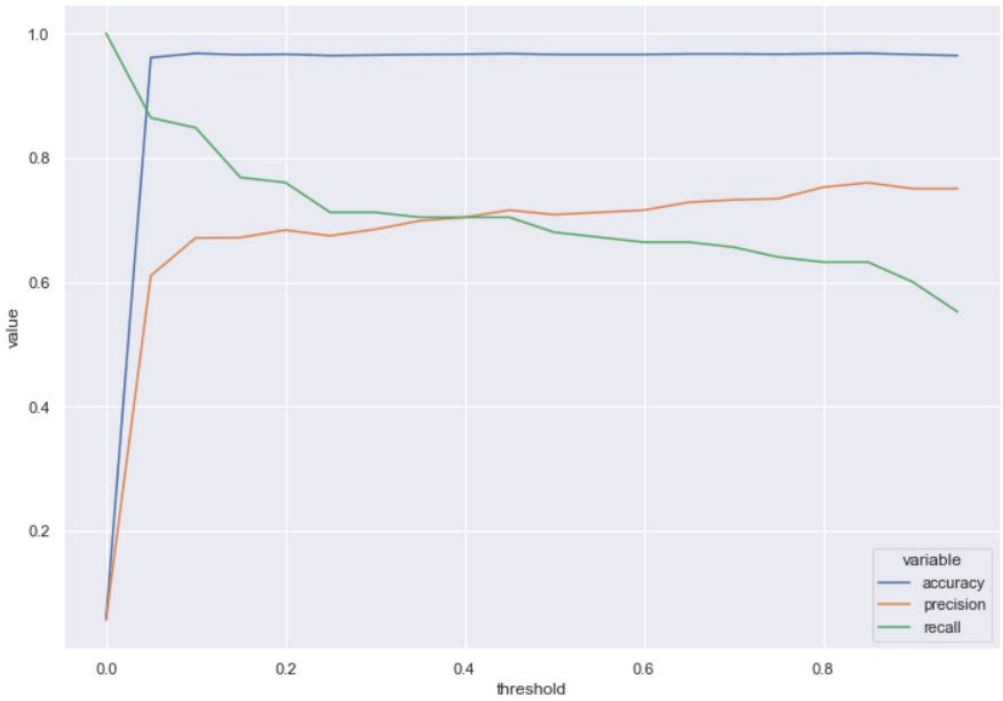
- If we filter for the predictions with confidence > 0.39, we’ll be left with about 68% of the original samples, which we can predict with 79% accuracy

Extending Text Classification Framework

- Using the same text cleaning and classification process, we can use grant text data to characterize other grant attributes, given a well-labeled, representative, and large enough training dataset
- When classifying grants as EHE / not EHE, this method delivered the following results:

Accuracy	Precision (TP / TP + FP)	Recall (TP / TP + FN)
97.1%	77.2%	70.4%

EHE Model Evaluation



Interpretation:

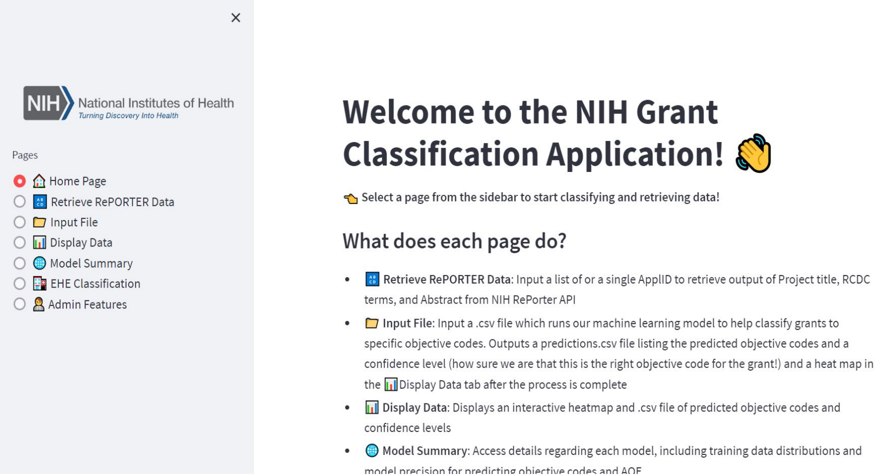
- Whenever the model creates a prediction, it outputs a value from 0-1, which represents the probability that the grant is EHE
- By default, the model predicts any grant with a value > 0.5 as EHE, but we can manually set this threshold to balance precision and recall

Example:

- If we set the threshold at 0.4 (i.e. classifying all grants with prediction > 0.4 as EHE), the precision and recall are about equal at ~70%

Website Goal

- **Website Goal:** Program a front-end application for scientific staff to input grant information which then runs an automated algorithm to classify HIV-related grants to objective codes
- Implemented webpage using Python Streamlit framework and SQLite3 for user authentication



Welcome to the NIH Grant Classification Application! 👋

👈 Select a page from the sidebar to start classifying and retrieving data!

What does each page do?

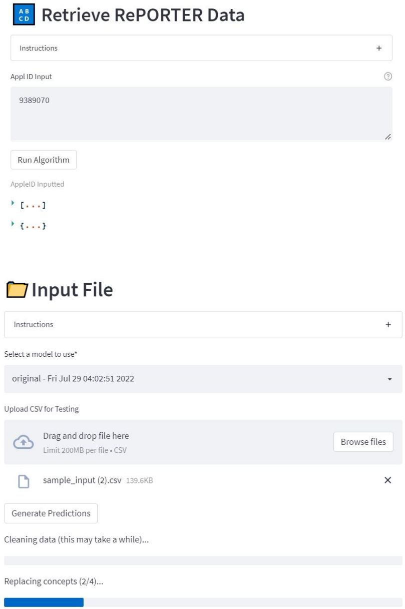
- 🔠 **Retrieve RePORTER Data**: Input a list of or a single ApplID to retrieve output of Project title, RCDC terms, and Abstract from NIH RePorter API
- 📁 **Input File**: Input a .csv file which runs our machine learning model to help classify grants to specific objective codes. Outputs a predictions.csv file listing the predicted objective codes and a confidence level (how sure we are that this is the right objective code for the grant!) and a heat map in the 📊Display Data tab after the process is complete
- 📊 **Display Data**: Displays an interactive heatmap and .csv file of predicted objective codes and confidence levels
- 🌐 **Model Summary**: Access details regarding each model, including training data distributions and model precision for predicting objective codes and AOF

Website Overview

- **Website Overview:** Functioning website allows OAR users to access the grant classifier tool, automating process of classifying grants to specific objective codes and now classifying EHE grants
    - Website also has the ability of retraining the model for admin users and contains a separate web page with details of each model's precision and accuracy
    - Platform contains accessible searching of grant information from NIH RePORTER API

Website Features

- 🔠 **Retrieve RePORTER Data**: Input a list of or a single Appl. ID to retrieve output of Project title, RCDC terms, and Abstract from NIH RePORTER API. (Made when trying to figure out other methods to run model with inputted Appl. ID only, but RePORTER has different items than ones we are training)
- 📁 **Input File**: Input a .csv file which runs our machine learning model to help classify grants to specific objective codes. Outputs a predictions.csv file listing the predicted objective codes and a confidence level and a heat map in the 📊Display Data tab after the process is complete



🔠 Retrieve RePORTER Data

Appl ID Input: 9389070

AppleID Inputted

```
[...]
{...}
```

📁 Input File

Select a model to use*: original - Fri Jul 29 04:02:51 2022

Upload CSV for Testing

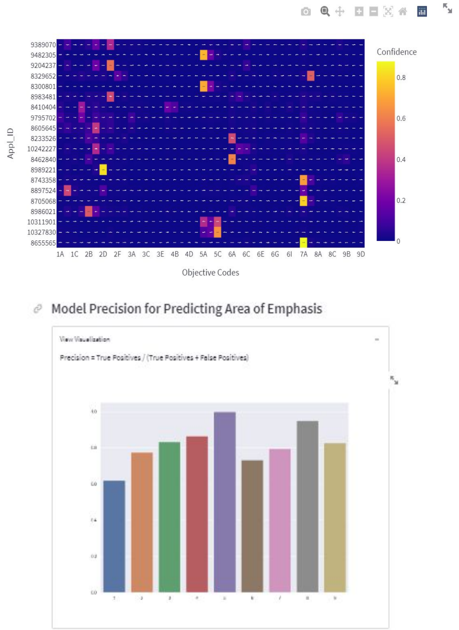
sample_input (2).csv 139.6KB

Cleaning data (this may take a while)...

Replacing concepts (2/4)...

Website Features Continued

- 📊 **Display Data**: Displays an interactive heatmap and .csv file of predicted objective codes and confidence levels
- 🌐 **Model Summary**: Access details regarding each model, including training data distributions and model precision for predicting objective codes and AOE
- 🏥 **EHE Classification**: Input .csv file and receive predictions .csv file containing Appl. Ids, EHE Predictions (1 = EHE Grant, 0 = Not EHE Grant), and prediction confidence



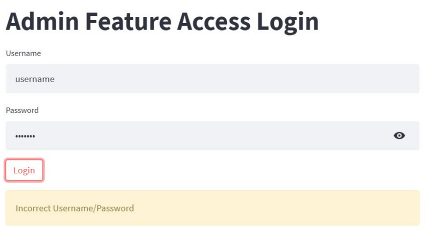
Model Precision for Predicting Area of Emphasis

View Visualization

Precision = True Positives / (True Positives + False Positives)

Admin Features

- 👨‍💼 **Admin Features:** Requires login to access admin features such as training model or making new user
- 🤖 **Train Model:** Input a .csv file with cleaned, tokenized text in column 'text_data' to train TF-IDF vectorizer and classification model.
- 👤 **Make New User:** Create new username and password for a new user (all passwords are hashed through a SHA256 encryption algorithm prior to storing, ensuring security)



Admin Feature Access Login

Username

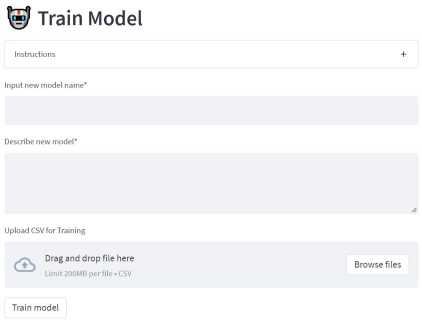
username

Password

•••••••

Login

Incorrect Username/Password



🤖 Train Model

Instructions +

Input new model name*

Describe new model*

Upload CSV for Training

Drag and drop file here
Limit 200MB per file • CSV

Browse files

Train model

Next Steps

- **Deployment**
  - Configured folder and compressed files of website code sent to SharePoint team
  - Website project was approved for deployment for Office of Aids Research internal use
  - Gather feedback post-deployment, then debug issues and improve user experience and features of website
- **Next Steps: Connection to IMPACII**
  - Integrate platform with IMPACII so users only need to input list of Appl. IDs, then website will query other necessary data needed from IMPACII with inputted Appl. IDs to run algorithm
  - Allows for more convenient input format instead of requiring a .csv file with data

ACKNOWLEDGEMENTS

- **National Institute of Health and Office of Aids Research**
  - Alison Lin
  - Danny Lumian
  - Danny Murphy
  - Robert Cregg
  - OAR Data Analytics Team
- **Coding It Forward Directors and Alumni Associates**

CIVIC DIGITAL FELLOWSHIP

NIH

JEREMY LEE
Harvard University
Statistics and Mathematics

WARREN QUAN
Princeton University
Computer Science