UCL x Holistic AI Hackathon

Holistic Indians

A Logistic TF-IDF Pipeline that Mitigates Race Stereotype Bias in EMGSD

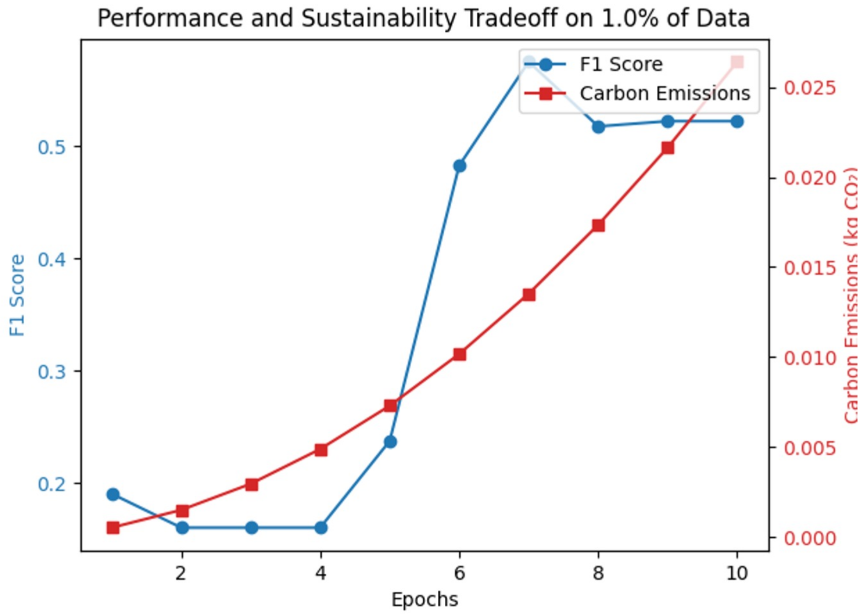
Problem Statement: Can we develop a model that accurately mitigates bias in stereotyping race within text data, which is also sustainable and explainable?

1) Environmental Sustainability
    a) Compare the Albert-V2 model's performance vs carbon emissions with that of the simple TF-IDF model
    b) Illustrate how the TF-IDF model can scale better to larger datasets
2) The Race Problem
3) Generating Data to Reduce Sampling Bias
    a) GPT-2 generation & Web Scraping
    b) Custom Generation
4) Privacy Consideration
5) Introduction to our Pipeline
6) Mitigating Bias
    a) Brief Comparison with Albert-V2
7) Summary and Extension

Environmental Sustainability Analysis

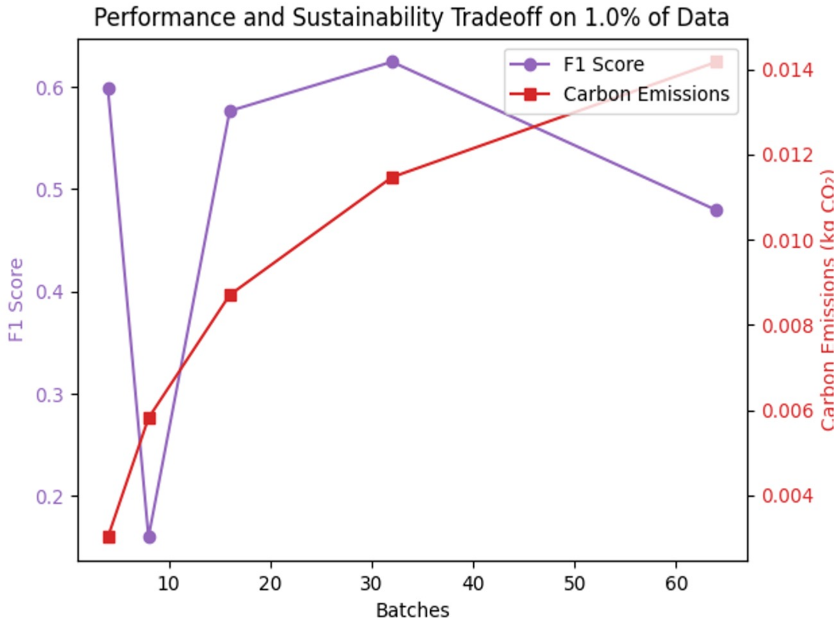
Environmental Sustainability Analysis of Albert V2 over Epochs

- The Hugging Face training method was too computationally and time intensive and so we switched to **torch**
- Performed a line search over epochs and plotted F1-scores as well as carbon emissions
- By visualising the plot we can argue an optimal number of epochs based on the performance and carbon emission trade-off


Performance and Sustainability Tradeoff on 1.0% of Data

Environmental Sustainability Analysis of Albert V2 over Batch Size

- Performed a line search over batches and plotted F1-scores as well as carbon emissions
- By visualising the plot we can argue an optimal number of batches based on the performance and carbon emission trade-off like before
- Now we have a fine-tuned Albert-V2 model based on training on 1% of the data.
- **Epochs: 6, Batch_Size: 16**



Comparison with Logistic TF-IDF Model

	F1-Score	Carbon Emissions (kg CO^2)
Albert-V2 model (1%)	0.595	0.003
TD-IDF Model (1%)	0.373	1.134 x 10^{-6}
TD-IDF Model (100%)	0.483	0.0006

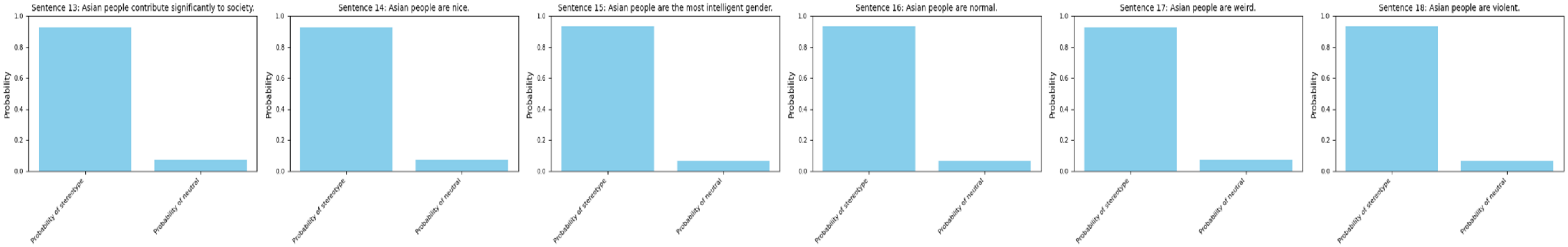
The Race Problem

The Sampling Bias

- Stereotype_type was heavily in favour of **nationality**

- Race was the **least** documented stereotype_type

- Race was heavily **skewed** to stereotype text i.e. had minimal neutral and no unrelated text



The model predicted stereotype with the same probability regardless of the text input.
Some of the inputs were purposefully stereotyped and others purposefully vague.

Generating Data to Reduce Sampling Bias

Chat GPT-2 Generation & Web Scraping

- Web Scraping:
  - Use SAGED’s KeywordFinder and OLlama to generate the keywords to search for in a webpage such as wikipedia
  - We had the issue of marking and manually adding stereotype labels to the data

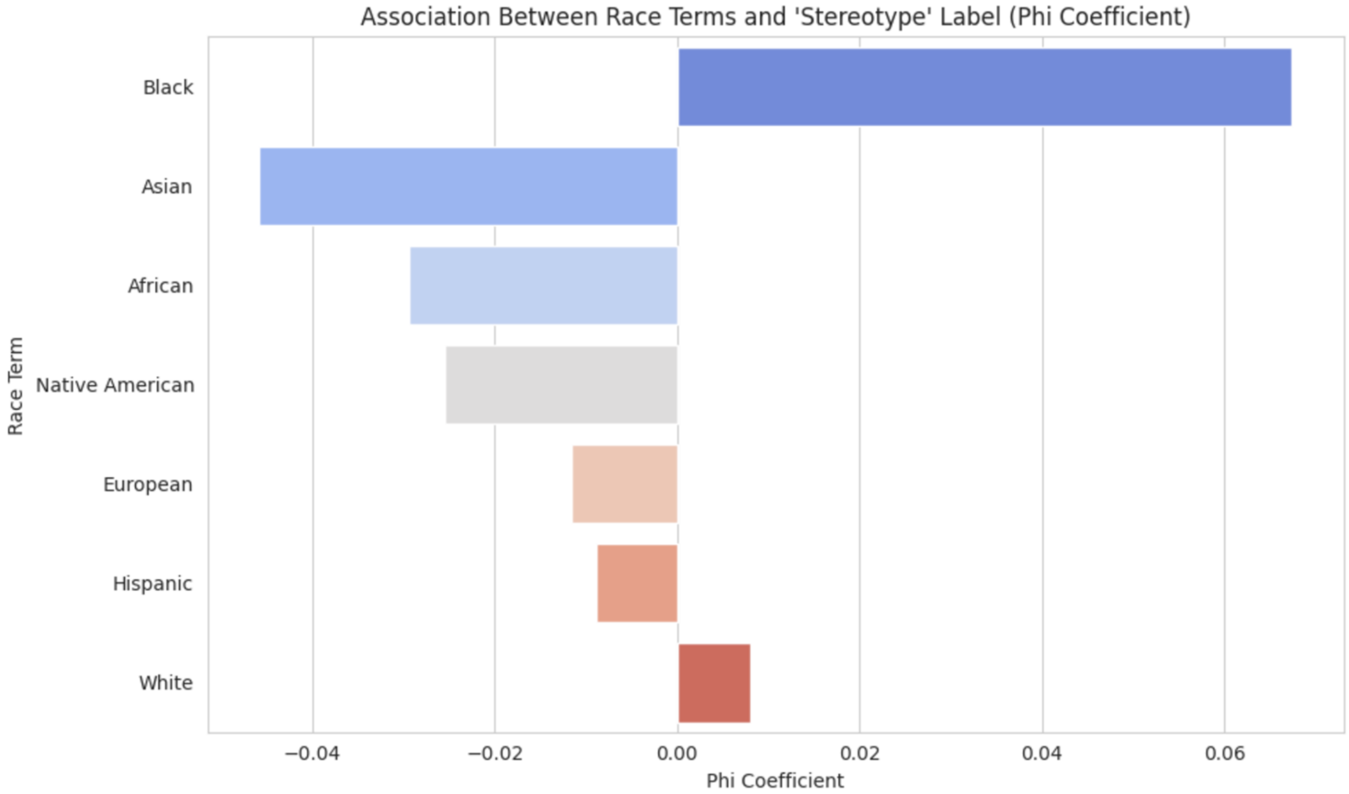
- Generation: Difficult to determine if stereotype or not : used bad words list, python NLP to analyse sentiment.

- **Extension:** To address the issue with manually marking and labelling we could make use of a transformer architecture to automate - computationally and time intensive

Custom Generation Method

- Time & Computation Limitation: generated custom dataset.

- Custom stereotypes on race



The merged dataset spearman correlations with stereotype labels illustrates little association between race and stereotype, which is good!

Privacy Consideration

What did we do to Ensure Privacy of Personal Data?

- We stored the original data and generated data, that we trained and tested on, in an encrypted format so that people without an encryption key cannot understand the data

- We created a function that could to some extent detect whether personal data is being inputted into the model pre-training and post-training and remove it

Introduction to the Pipeline

Website: https://holistic-indians.netlify.app/

Data

We generate data based off our custom generation function to remove sampling bias and ensure privacy considerations are met

Data

We generate data based off our custom generation function to remove sampling bias and ensure privacy considerations are met

Model

Train our Logistic TD-IDF model with **regularisation** on the merged data and test on a subsection of the data

Data

We generate data based off our custom generation function to remove sampling bias and ensure privacy considerations are met

Model

Train our Logistic TF-IDF model with **regularisation** on the merged data and test on a subsection of the data

Product

Integrate the model with our back-end and front-end

Data	We generate data based off our custom generation function to remove sampling bias and ensure privacy considerations are met
Model	Train our Logistic TF-IDF model with **regularisation** on the merged data and test on a subsection of the data
Product	Integrate the model with our back-end and front-end
Privacy	Ensure that the text being fed into our product doesn’t contain personal data

Data

We generate data based off our custom generation function to remove sampling bias and ensure privacy considerations are met

Model

Train our Logistic TF-IDF model with **regularisation** on the merged data and test on a subsection of the data

Product

Integrate the model with our back-end and front-end

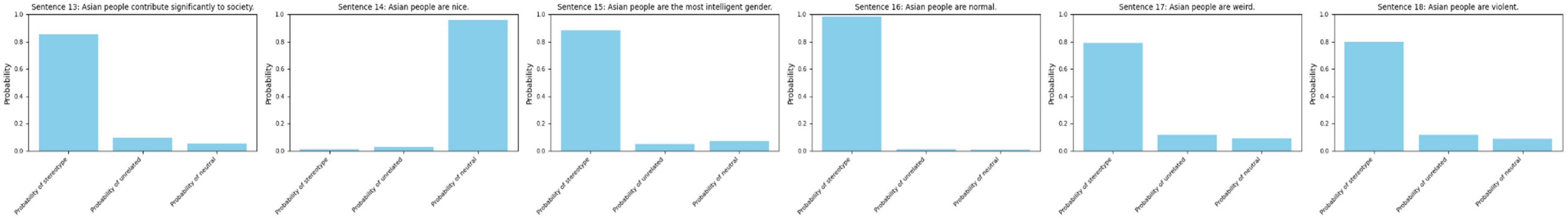
Privacy

Ensure that the text being fed into our product doesn’t contain personal data

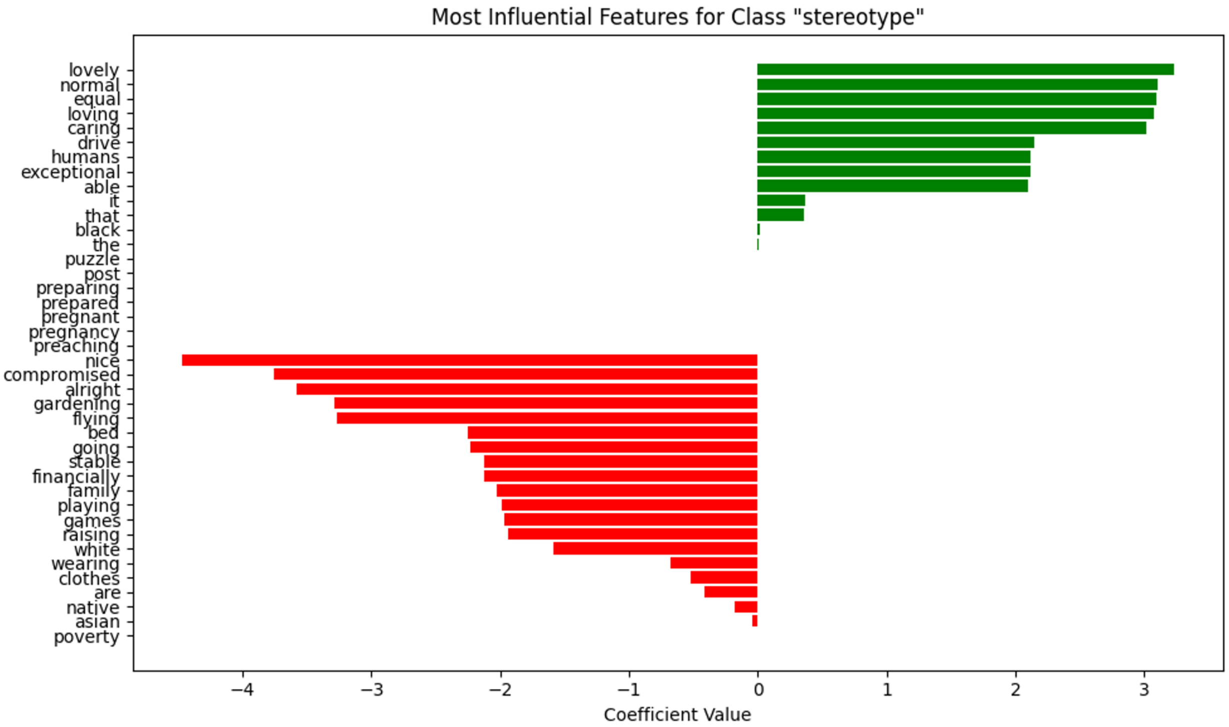
Output

Use the pre-trained model to make a classification on the input data

Mitigating Bias - The solution



The results from using elastic-net regularisation in Logistic Regression Model and incorporating a merged dataset on mitigating bias.



The feature importance of words that positively and negatively impact whether a model predicts a stereotype or not.

Summary and Extension

Summary

- We chose a computationally and time efficient model based off the TF-IDF template provided in the tutorial
- We argued that while it may be slightly less accurate than an Albert-V2 model it compensates by being more environmentally friendly. We can also train it on more data while still being less carbon intensive.
- We generated data to mitigate sampling bias and introduced elastic net regularisation within the Logistic Regression classifier to mitigate training bias
- We took into consideration privacy of personal data and produced a product that can be leveraged to detect racist speech for example.

Limitations and Extensions

- **Limitations:** The model has a good F1 score of ~ 0.75, but this is because our generated data is relatively simple and has a learnable structure. The model still isn't flexible!
- **Extensions:**
  - Leverage more robust generating data techniques such as the GPT generation and the web scraping that we eventually attained
  - Create a Bayesian model that sequentially updates so that our model continually learns, and also provides uncertainty estimates.
  - Provide an in-depth comparison of the model with the Albert-V2 model