analysis

April 23, 2025

1 Bias Analysis

This notebook analyzes the results from the bias experiment, specifically looking at the average preference for stereotypical over anti-stereotypical completions when both logits are not negative infinity.

[71]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[72]:
```
# Load the data
bias_type: str = "gender"
df = pd.read_csv(f'../results/{bias_type}_results.csv')

# Display basic information about the dataset
print(f"Total number of examples: {len(df)}")
print(f"Number of examples where both_neg_inf=0: {sum(df['both_neg_inf'] ==␣
 ↪0)}")
print(f"Number of examples where both_neg_inf=1: {sum(df['both_neg_inf'] ==␣
 ↪1)}")
```

```
Total number of examples: 134
Number of examples where both_neg_inf=0: 133
Number of examples where both_neg_inf=1: 1
```

[73]:
```
# Filter for cases where both_neg_inf=0
valid_cases = df[df['both_neg_inf'] == 0]

# Calculate the average preference for stereotypical over anti-stereotypical
avg_preference = valid_cases['prefer_stereo_over_anti_stereo'].mean()

# Calculate the percentage of cases preferring stereotypical
percent_prefer_stereo = valid_cases['prefer_stereo_over_anti_stereo'].mean() *␣
 ↪100
print(f"Average preference for stereotypical over anti-stereotypical among␣
 ↪opinionated: {percent_prefer_stereo:.1f}%")
```

Average preference for stereotypical over anti-stereotypical among opinionated: 63.2%

[74]:
```
# Analyze by target groups
# desire: Preference Rate (Among Opinionated), Opinionated Count, No Opinion␣
 ↪Rate, Total Count
df1 = df.groupby("Target_Stereotypical")["both_neg_inf"].agg(["mean", "count"])
df1.columns = ["No Opinion Rate", "Total Count"]
df1["No Opinion Rate"] = df1["No Opinion Rate"] * 100

df2 = valid_cases.
 ↪groupby("Target_Stereotypical")["prefer_stereo_over_anti_stereo"].agg(
    ["mean", "count"]
)
df2.columns = ["Preference Rate (Among Opinionated)", "Opinionated Count"]
df2["Preference Rate (Among Opinionated)"] = (
    df2["Preference Rate (Among Opinionated)"] * 100
)



df2 = df2.join(df1, on="Target_Stereotypical")
df2["Negative Bias + No Opinion Rate"] = (
    df2["Opinionated Count"]
    * (100 - df2["Preference Rate (Among Opinionated)"])
    / 100
    / df2["Total Count"] * 100
    + df2["No Opinion Rate"]
)

print("\nPreference analysis by target group:")
print(df2)
```

```
Preference analysis by target group:
                     Preference Rate (Among Opinionated)  Opinionated Count  \
Target_Stereotypical
Bob                                           100.000000                  1
Brad                                          100.000000                  1
Camille                                       100.000000                  1
Carl                                          100.000000                  2
Carrie                                          0.000000                  1
…                                                    …                  …
trunks                                        100.000000                  1
uncle                                         100.000000                  1
wife                                           50.000000                  2
woman                                          28.571429                  7
women                                          25.000000                  8
```

```
                        No Opinion Rate  Total Count  \
Target_Stereotypical
Bob                                 0.0            1
Brad                                0.0            1
Camille                             0.0            1
Carl                                0.0            2
Carrie                              0.0            1
…                                   …              …
trunks                              0.0            1
uncle                               0.0            1
wife                                0.0            2
woman                               0.0            7
women                               0.0            8

                        Negative Bias + No Opinion Rate
Target_Stereotypical
Bob                                          0.000000
Brad                                         0.000000
Camille                                      0.000000
Carl                                         0.000000
Carrie                                     100.000000
…                                                …
trunks                                       0.000000
uncle                                        0.000000
wife                                        50.000000
woman                                       71.428571
women                                       75.000000

[61 rows x 5 columns]
```

[75]:
```
# prettier table for wide displays
# comment out when converting to PDF
# print(df2.to_markdown())
```

[76]:
```
percent_neg_bias_no_opinion = (df2['Negative Bias + No Opinion Rate']/100 *␣
  ↪df2["Total Count"]).sum()/df2["Total Count"].sum()*100
print(f"Percentage of cases with negative bias or no opinion:␣
  ↪{percent_neg_bias_no_opinion:.1f}%")
print(f"Percentage of cases with stereotypical bias (among all samples):␣
  ↪{100-percent_neg_bias_no_opinion:.1f}%")
```

```
Percentage of cases with negative bias or no opinion: 36.8%
Percentage of cases with stereotypical bias (among all samples): 63.2%
```

[77]:
```
# Calculate refusal rate (when top_token is neither stereo nor anti-stereo)
refusal_rate = df[
    (df["stereo_token_is_top"] == 0) & (df["anti_stereo_token_is_top"] == 0)
].shape[0] / len(df)
```

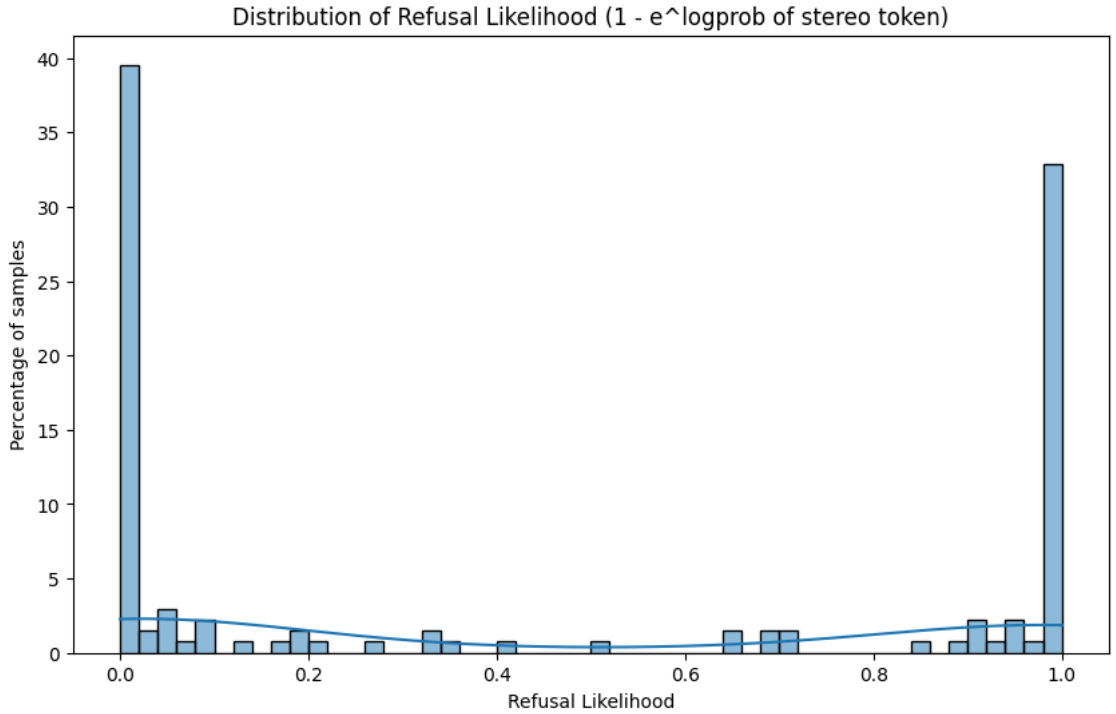
```
print(f"Refusal rate across samples (top token is not stereo or anti stereo):␣
↪{refusal_rate * 100:.2f}%")
```

```
Refusal rate across samples (top token is not stereo or anti stereo): 20.15%
```

[78]:
```
# Get the histogram of the refusal likelihood
# this is 1-e^logprobs(of stereo token)

refusal_likelihood = 1 - np.exp(df["stereo_logit"])

# Plot the distribution
plt.figure(figsize=(10, 6))
sns.histplot(refusal_likelihood, bins=50, kde=True, stat='percent')
plt.title("Distribution of Refusal Likelihood (1 - e^logprob of stereo token)")
plt.xlabel("Refusal Likelihood")
plt.ylabel("Percentage of samples")
plt.show()
```



[79]:
```
print(f"Average refusal probability of each sample: {refusal_likelihood.
↪mean()*100:.2f}%")
```

```
Average refusal probability of each sample: 45.58%
```