Backpacking or Travel?

Using Naïve Bayes Model and Random Forest Classifier to classify subreddit posts

Agenda

PROBLEM STATEMENT

DATA BACKGROUND

EXPLORATORY DATA ANALYSIS

MODELING PROCESS AND RESULTS

TAKEAWAYS AND RECOMMENDATION

Problem Statement

- An unfortunate power outage on some Reddit servers has caused some posts (from r/backpacking and r/travel) to be stored incorrectly within the servers
- As an employee of Reddit, my supervisor has tasked me to **correctly reclassify these posts** by training classifier models to solve this issue
- We will be training the models based on about **2000 reddit posts** (about 1000 posts from each subreddit)

Data Background

- **Pushshift API**
  - 100 posts per requests
  - Removed any duplicated posts
- **Cleaning**
  - Dropped 6 rows containing missing values
  - Checked that there are also no mod bot messages
  - Removed posts containing ‘[removed]’ (2 rows)
  - Lowercased all words and removed hyperlinks, white spaces, numbers
- **Preprocessing**
  - Lemmatize words (days -> day, nights -> night)
  - Added to stop words: ‘backpacking’, ‘travel’ plus other generic words

r/backpacking
(cleaned)
993 posts

r/travel
(cleaned)
1045 posts

Combined
2038 posts

Exploratory Data Analysis

r/backpacking

- **832** unique users
- **1.19** post per user
- Longest post by word count: **7,493** words (trip report)
- Shortest post by word count: **6** words (title of an image)

r/travel

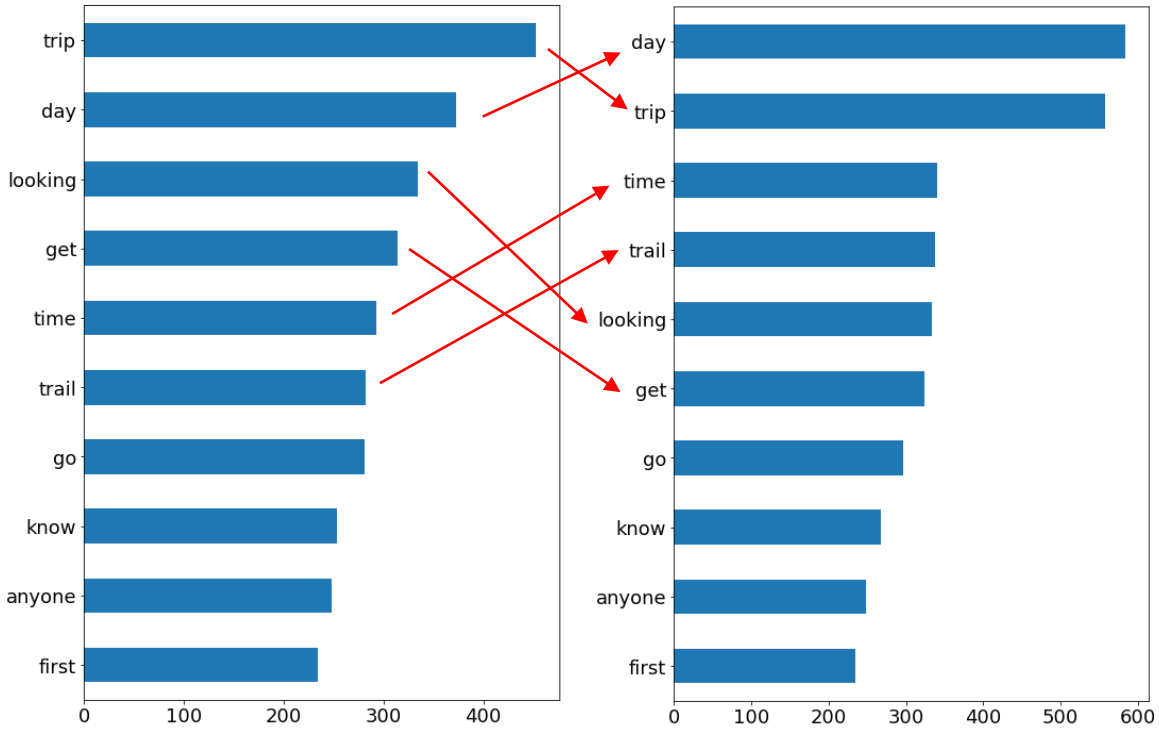
- **969** unique users
- **1.08** post per user
- Longest post by word count: **2,535** words (covid restriction discussion while traveling)
- Shortest post by word count: **3** words (user replying via post to thank someone)

Exploratory Data Analysis

- 10 most frequent words using CountVectorizer

r/backpacking

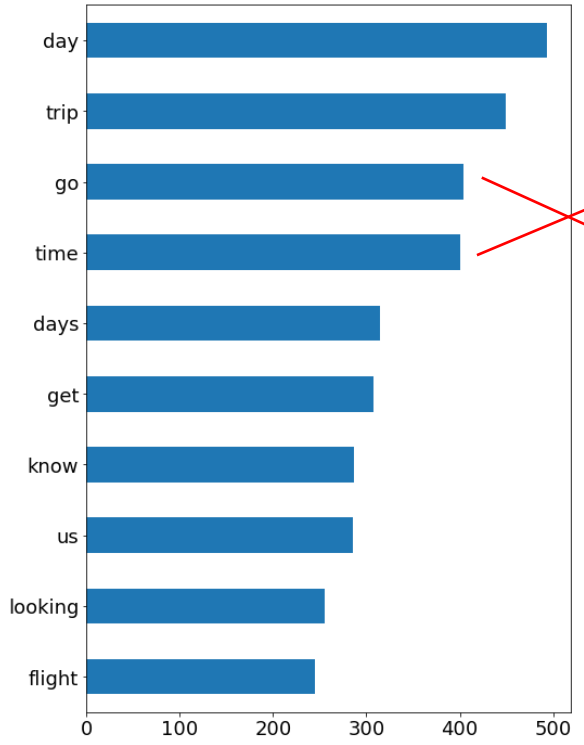
w/o lemmatize



w/ lemmatize

r/travel

w/o lemmatize



w/ lemmatize

Modeling Process and Results

1. Train test split: stratify y, setting a random state to rerun models
2. Fit and run models using Pipeline and GridSearchCV:
    1. 2 models: Naïve Bayes and Random Forest
    2. Baseline for each model is the default hyperparameters using CountVectorizer

Naïve Bayes

Tf – IDF Vectorizer

GridSearch best hyperparameters:

- 'nb__alpha': 0.5
- 'tvec__max_features': 6500
- 'tvec__ngram_range': (1, 2)

Train score: 0.8514

Test score: 0.8235

Random Forest

Tf – IDF Vectorizer

GridSearch best hyperparameters:

- 'rf__max_depth': None
- 'rf__n_estimators': 200
- 'tvec__max_features': 10000
- 'tvec__ngram_range': (1, 3)

Train score: 0.8220

Test score: 0.8059

Modeling Process and Results
Feature importance of Naïve Bayes

Top 10 word contributors to differentiate backpacking post from travel post

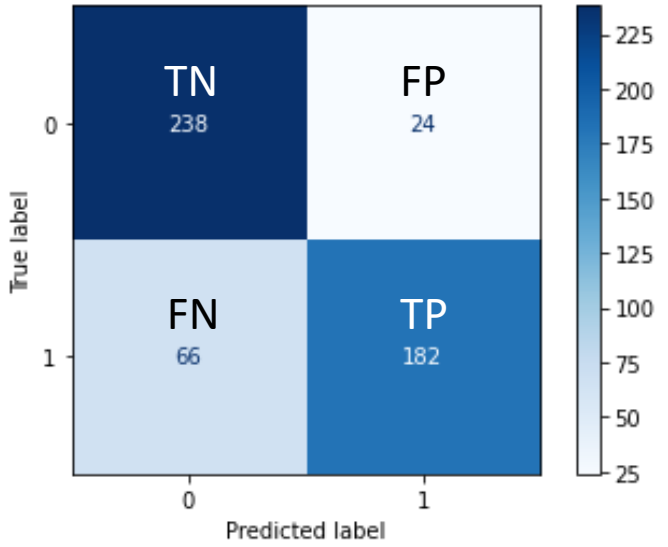
	log_prob_difference	
trail	2.811082	
mile	2.667665	
gear	2.655979	
sleeping	2.558983	
pack	2.463669	
tent	2.368062	
sleeping bag	2.324710	
wilderness	2.310666	
osprey	2.220558	*(a backpack brand)*
lb	2.201625	*(pounds in short form)*

Bottom 10 word contributors to differentiate backpacking post from travel post

	log_prob_difference
passport	-2.291786
airline	-2.262217
flight	-2.250027
paris	-1.926466
airport	-1.886119
madrid	-1.880912
ticket	-1.843540
american	-1.809327
euro	-1.788126
florence	-1.772215

Misclassification Analysis on Best Model: Naïve Bayes

- **Accuracy score:** 82.35%
- **Subreddit:**
  - 0: Backpacking
  - 1: Travel



- **False positives:** posts that incorrectly classified as backpacking
- **False positives:** posts that incorrectly classified as travel
- Most misclassified posts were **long posts**
  - Average word count: 91 words
  - The longest post being 721 words
- The most common misclassified words were: thanks, going, place

Takeaways and Recommendations

- For the 2 subreddits: Naïve Bayes **marginally performs better** than Random Forest.
- Surprisingly, the concern for the naïve assumption that all features are independent has minimal impact to the model's capability to classify the reddit posts accurately
- Naïve Bayes is easier to train and while Random Forest takes time to train and consumes more time to predict proportional to the number of trees (computationally more expensive)

- Potential commercial application for a company like TripAdvisor?

- **Next step:**
  - repeat model on other similar subreddits to further evaluate model performance
  - collect more training data
  - try more models like boosting or SVM