Predicting Dating Advice Subreddit Classification

Predicting which dating subreddit is best fit for user posts and conversations

Bob Adams
General Assembly - Data Science Immersive

Problem Statement

Dating is complicated - the right advice is key to navigating relationships at all stages. Two popular subreddits for dating advice are r/dating and r/datingoverthirty. This project aims to leverage key portions of conversations (the post and top comment) from these two communities to predict where a post originated. In practice, an effective model could be used to prompt posters into communities based on their post - to get the best advice for them!

Thank you, volunteer Data Scientists for your review of the models evaluated - and for your consideration in expanding the findings to a new service (if u/spez allows it.)

Process Overview

01

Community Overview

Community Overlap and Characteristics

02

Data Sourcing via API

Posts and Comments from Reddit via praw

03

EDA and Evaluation

Post and comment characteristics, sentiment analysis, data usage

04

Model Speed Dating

Fitting and evaluating 7 different models

05

Model Evaluation

Accuracy is easier than chemistry

06

Recommendations

Findings and next steps

If you came here looking for dating advice...

After 12 years in a relationship... and reading over 100 conversations to catch up to speed...

Don't. (Let's hope this model makes it a bit easier)

Community Overview

r/dating
Subscribers: **2.3M**
Daily Posts: **440**
Daily Comments: **4054**

r/datingoverthirty
Subscribers: **1.1M**
Daily Posts: **5** (extrapolated)
Daily Comments: **50** (extrapolated)

Similar mission with a more niche audience

Subscribers of r/dating are **46.2x** as likely to subscribe to r/datingoverthirty than the average redditor.

Source: Subredditstats.com

Data Sourcing via praw

Limits: Data sourced prior to access updates to r/datingoverthirty with a per-request limit of 1k posts.

Remediation: Additional information is available for r/dating, but would create a *class imbalance*.

Recommendation: Refit any selected model leveraging a broader dataset from both communities.

Title

The short title of each post per subreddit.

Selftext

The contextualized description (body) of the post.

Conversations happen in comments and threads.

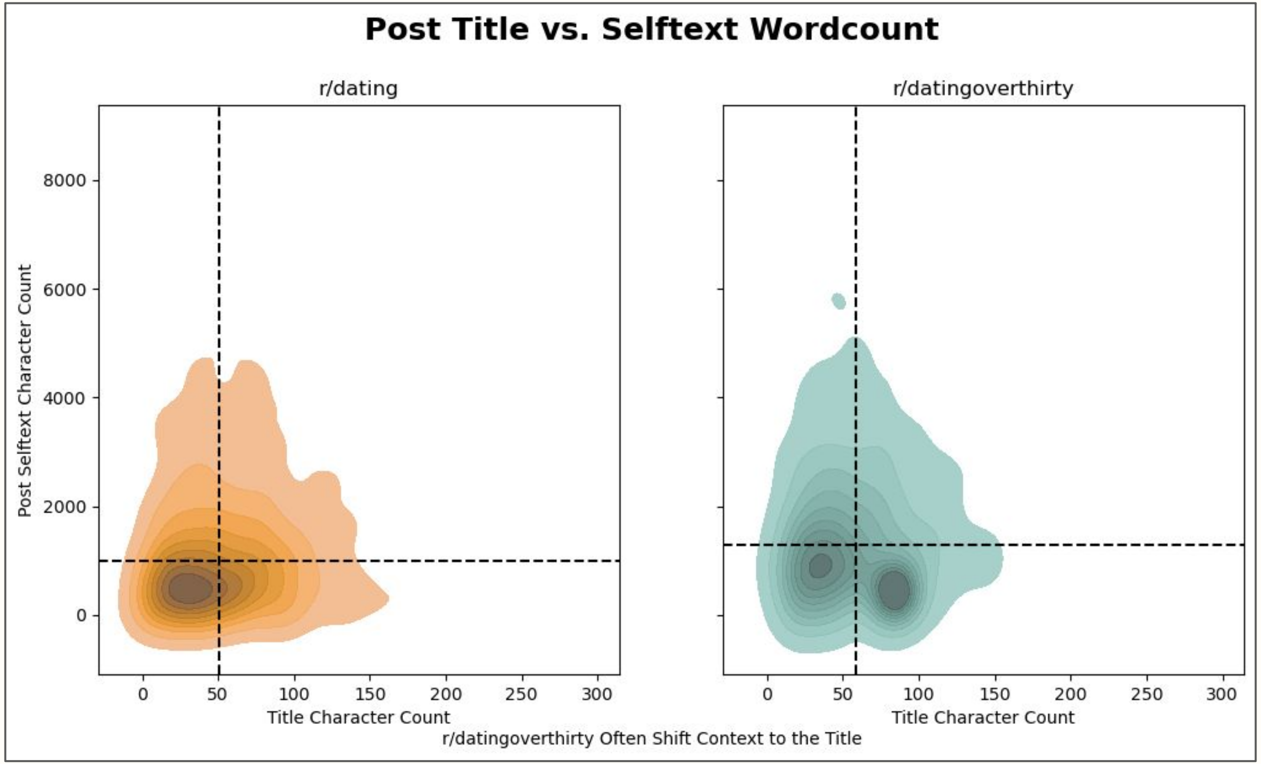
Top Comment

The top upvoted comment per thread (excluding ads, auto-moderator posts).

Generally the most popular thread by value and engagement.

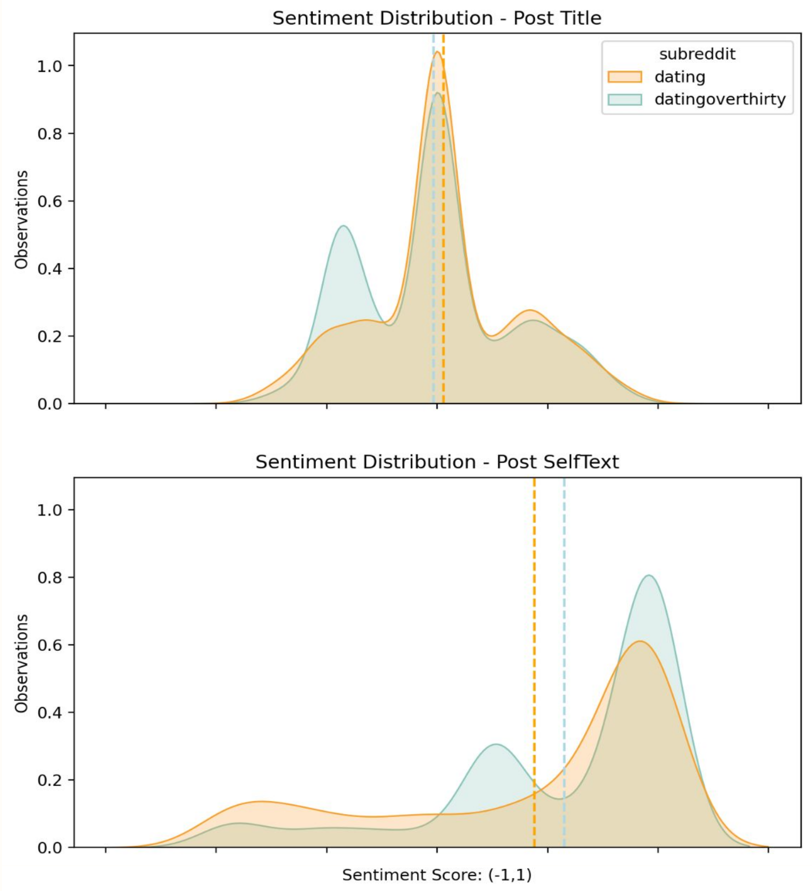
Exploratory Data Analysis

Title and Selftext Lengths - Context Provided



Exploratory Data Analysis

Sentiment Analysis



Sentiment Distribution - Post Title

Sentiment Distribution - Post SelfText

Classification Model Speed Dating

M1 - Multi-Estimator w/TFIDF Vectorizer

79.9%

- RandomizedSearchCV w/ TFIDF params
- Logistic Regression / Multinomial Naive Bayes / Kernelized SVM

M2 - Multi-Estimator w/CountVectorizer

80.3%

- RandomizedSearchCV w/ CountVectorizer params
- Logistic Regression / Multinomial Naive Bayes / Kernelized SVM

M3 - Bootstrap Aggregated Trees w/TFIDF Vectorizer

76.5%

- RandomizedSearchCV
- Tree Depths between 5 and 15
- 200 Estimators

Classification Model Speed Dating

M4 - RandomForest Classifier w/TFIDF

81.5%

- RandomizedSearchCV
- Tree depths between 5 and 30
- 200 trees

M5 - AdaBoost Boosted Decision Trees w/TFIDF

77.9%

- RandomizedSearchCV
- Learning Rates between 0.1 and 10
- Estimators between 5 and 30

M6 - Kernelized SVM w/TFIDF

84.6%

- RandomizedSearchCV
- Polynomial and RBF Kernels

82.2% **M7 - Hard Voting Classifier**

Recommendations

After evaluating seven models, spanning pre-processing techniques, vectorizers and estimators, an **84.5%** accuracy level is achievable when categorizing reddit conversations into dating subreddits.

Next Steps:

- Continue evaluating Support Vector Machines and TfI-Idf Vectorization
- Re-fit the model leveraging an expanded dataset once Reddit reopens
- Consider moving forward with a subreddit recommendation service - we don't know the answers, but we know who can (with an 84.5% accuracy!)

Thanks!

Recommendations

After evaluating seven models, spanning pre-processing techniques, vectorizers and estimators, an **84.5%** accuracy level is achievable when categorizing reddit conversations into dating subreddits.

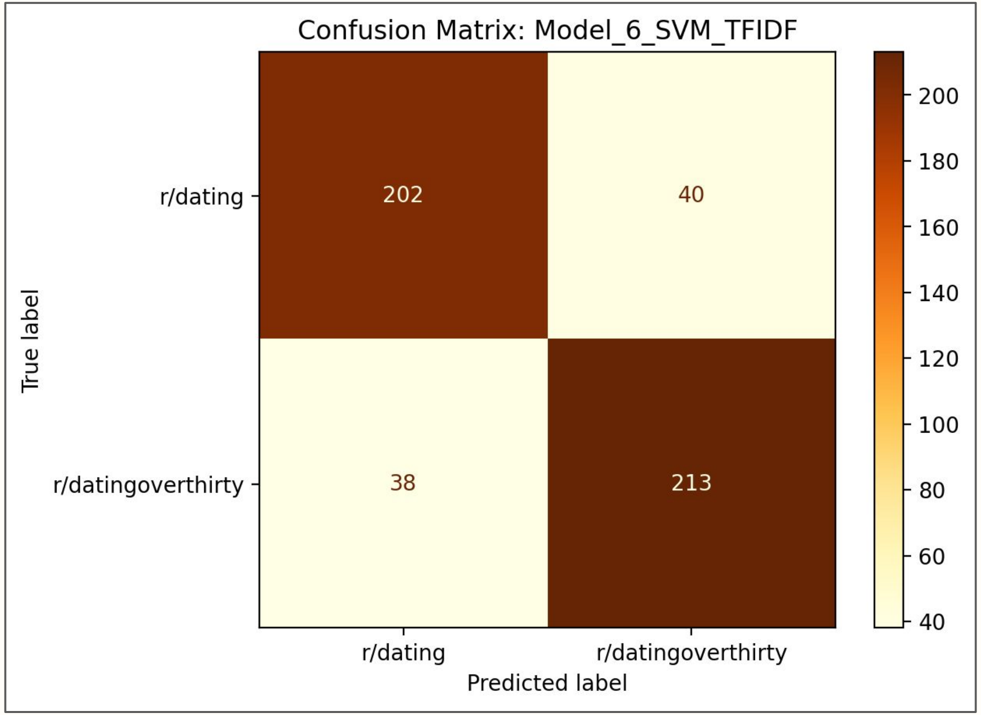
Next Steps:

- Continue evaluating Support Vector Machines and TfI-Idf Vectorization
- Re-fit the model leveraging an expanded dataset once Reddit reopens
- Consider moving forward with a subreddit recommendation service - we don't know the answers, but we know who does (with an 84.5% accuracy!)

CREDITS: This presentation template was created by **Slidesgo**, including icon by **Flaticon**, and infographics & images from **Freepik** and illustrations by **Storyset**

Appendix

Model 6 Performance



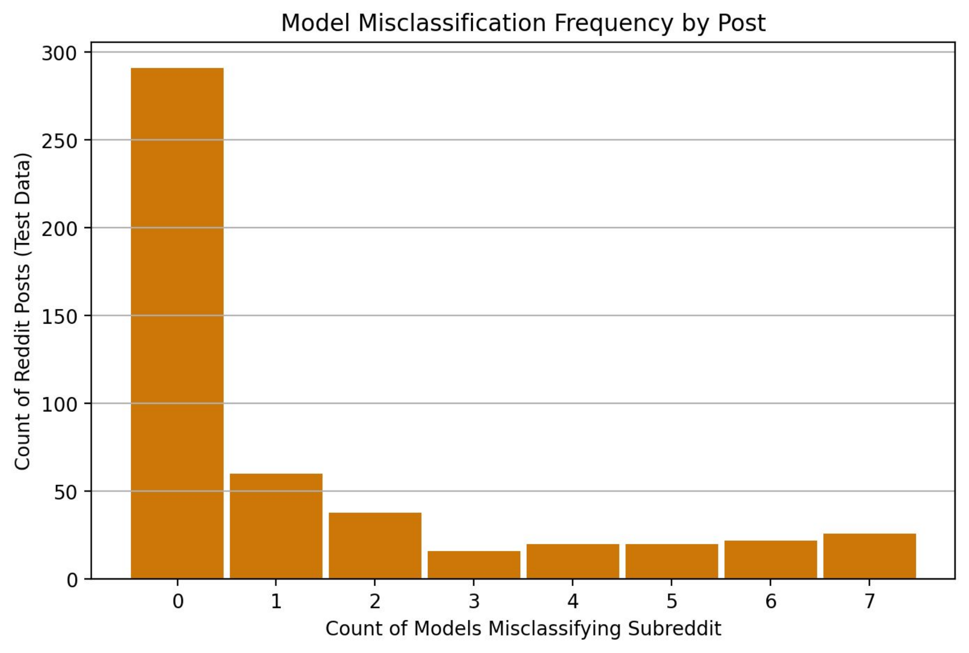
Confusion Matrix:

- **Top-Left**: Correctly predicted **dating**
- **Top-Right**: Incorrectly predicted **datingoverthirty**
- **Bottom-Left**: Incorrectly predicted **dating**
- **Bottom-Right**: Correctly predicted **datingoverthirty**

Even split indicates low bias between categories.

Model Miscategorizations

Why did performance degrade when results were ensembled?

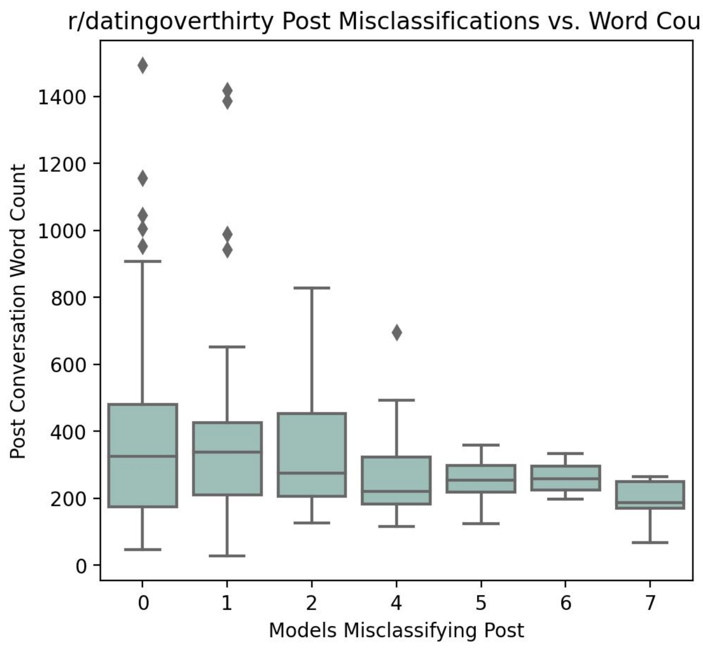


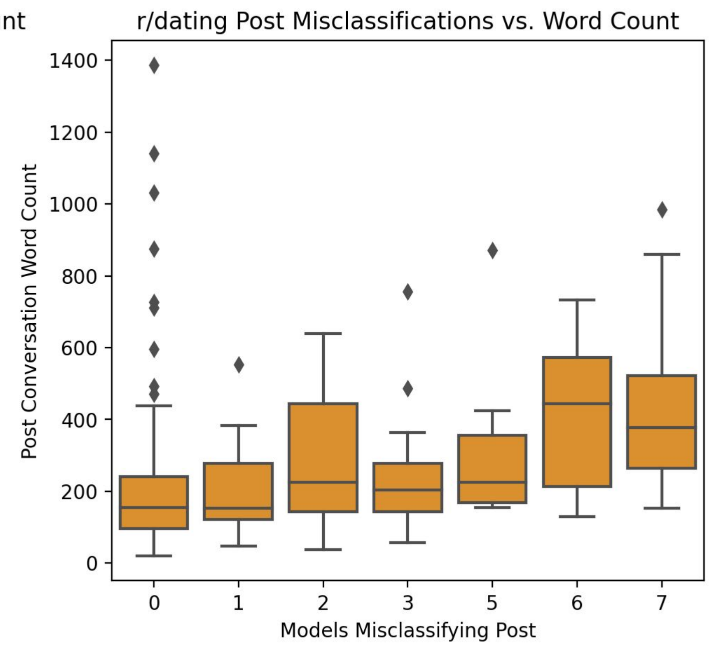
Models frequently miscategorized the same posts - ensembling predictions is likely to result in reduced predictive performance.

If we saw a strong right skew in this distribution (0-2, with a tail through 7), the models would be shown to disagree on mis-classifications more often than not - resulting in predictive improvements.

Model Miscategorizations

Why did the models predict the same incorrect category?





Conversation Length - As word count increases, models to miscategorize posts as r/datingoverthirty. The reverse also holds.