Phase 5 - Capstone Project

Stack Overflow - Question Modeling for Quality-of-Life

Will Dougherty - Flatiron School - Online Data Science

Overview

Stack Overflow Question Modelling

Two Components of this Project

- Question Quality
- Tag Suggestion

Goals

- Provide models that can support in-the-moment text analysis

- The models should be accurate enough to:
  - Alert users if post is predicted to be low-quality
  - Suggest top 20 tags based on text of the question

- This could improve overall post quality, helping original user and all users
- Save time and resources for Stack Overflow's moderation team
- As well, improved tagging improves search and related-post features

Question Quality

Dataset and Methodology

“60k Stack Overflow Questions with Quality Rating”

- Used to build model - classifies posts as ‘high’ or ‘low’ quality

“Python Questions from Stack Overflow”

- No quality rating, but the model is tested on this dataset to see correlations with features

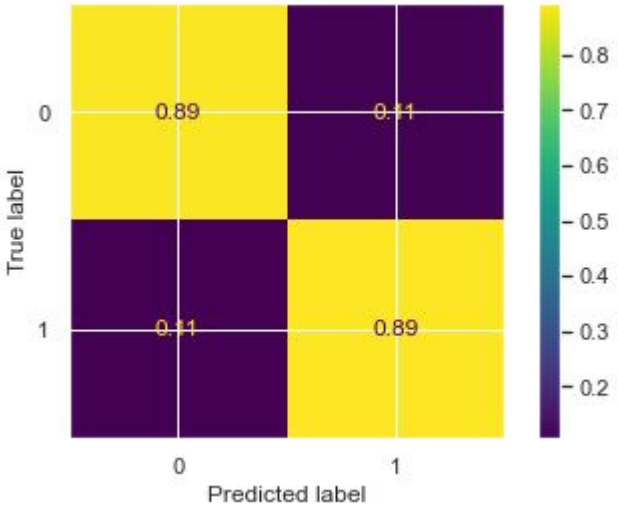
Model Details

- Using text to predict target
- Text cleaning, vectorization, target class balancing
- Logistic Regression

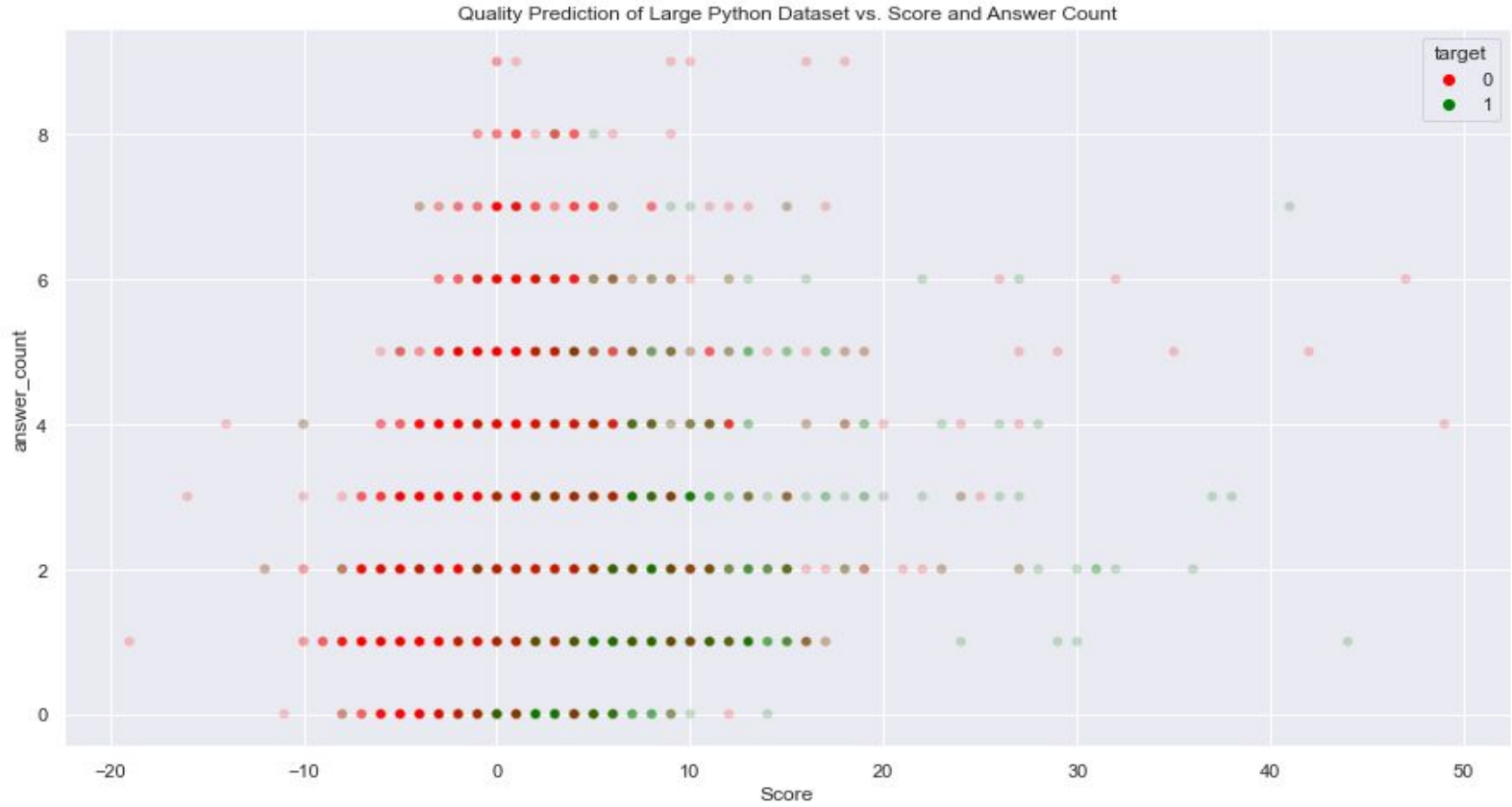
Results

The model does well in distinguishing and classifying between the two classes.

89% of both the ‘high’ and ‘low’ quality posts are correctly identified.



Application to larger dataset



Tag Suggestion

Dataset and Methodology

“StackSample: 10% of Stack Overflow Q&A”

- Very large dataset, not limited to ‘python’ tags
- Full set: 1.2 million samples, 37,000 unique tags

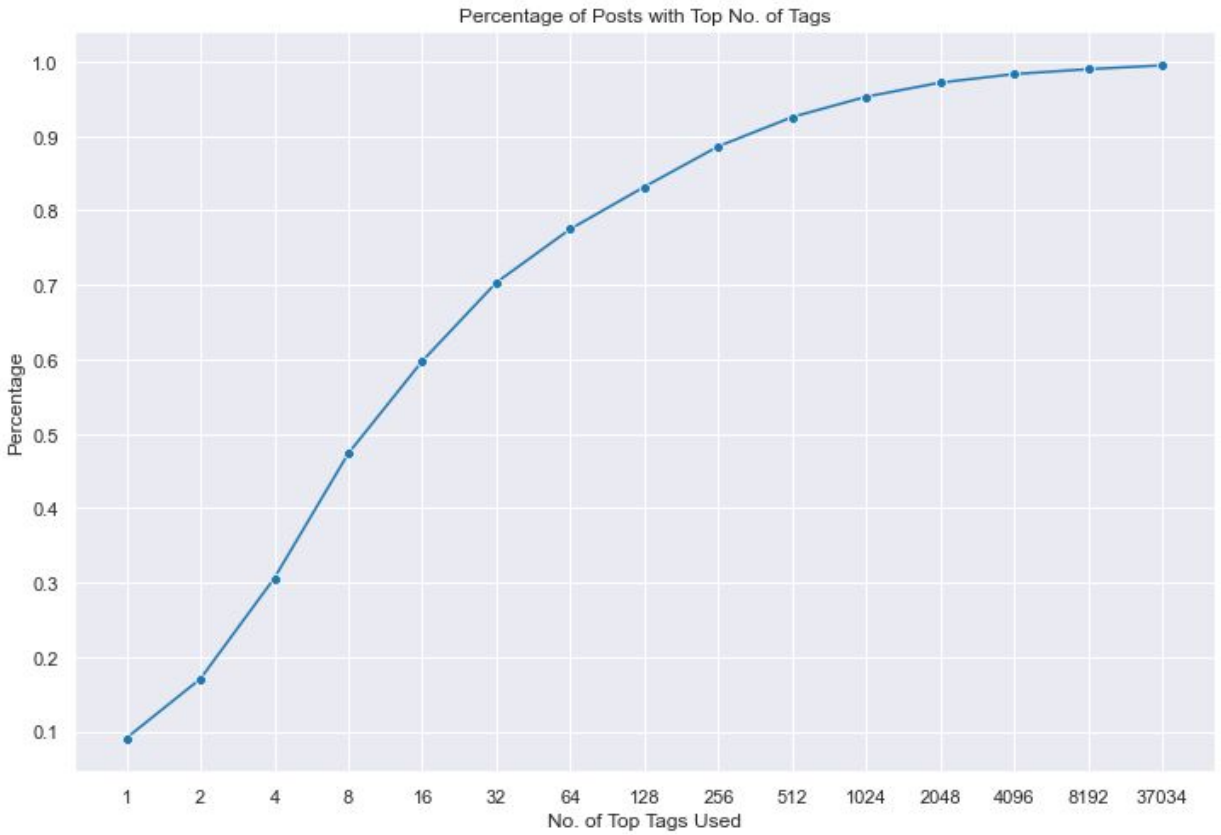
Modelling

- One-vs-Rest Classifier - with SGD-Logistic Regression
- Can predict probabilities to generate ranked list
- Model: 200,000 samples, 2,000 tags

Top Tags

Very few tags account for vast majority of posts' tags

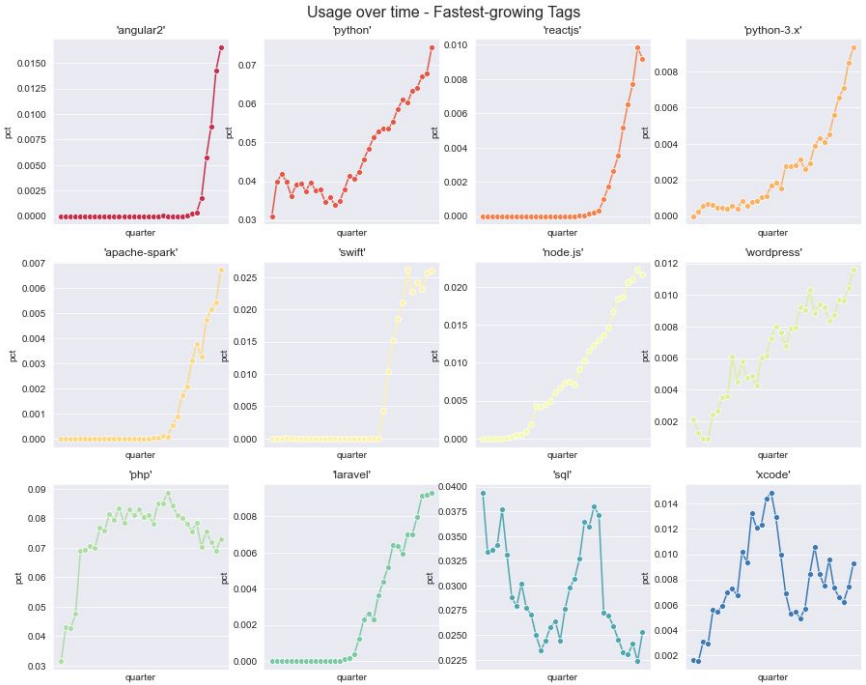
Top 2,000 tags are used in 97% of posts



Change in Tag Usage Over Time

Churn and turnover suggests a large but recent training set is best to capture up-to-date proportions of tag usage

200,000 samples, roughly the last year of data



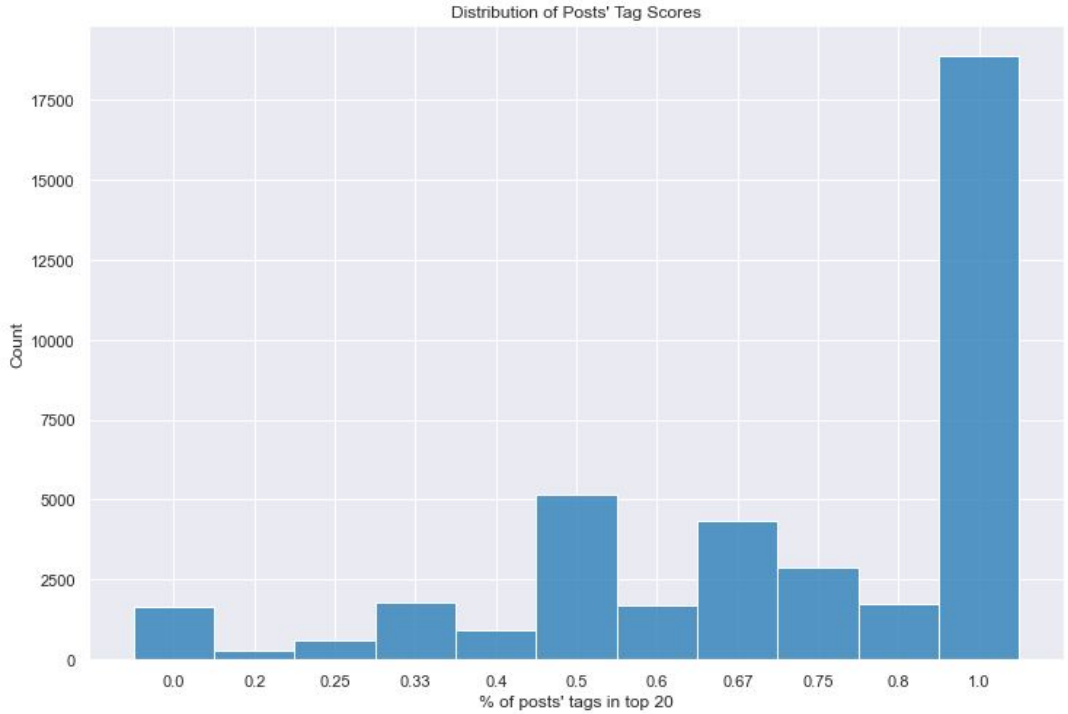
Results

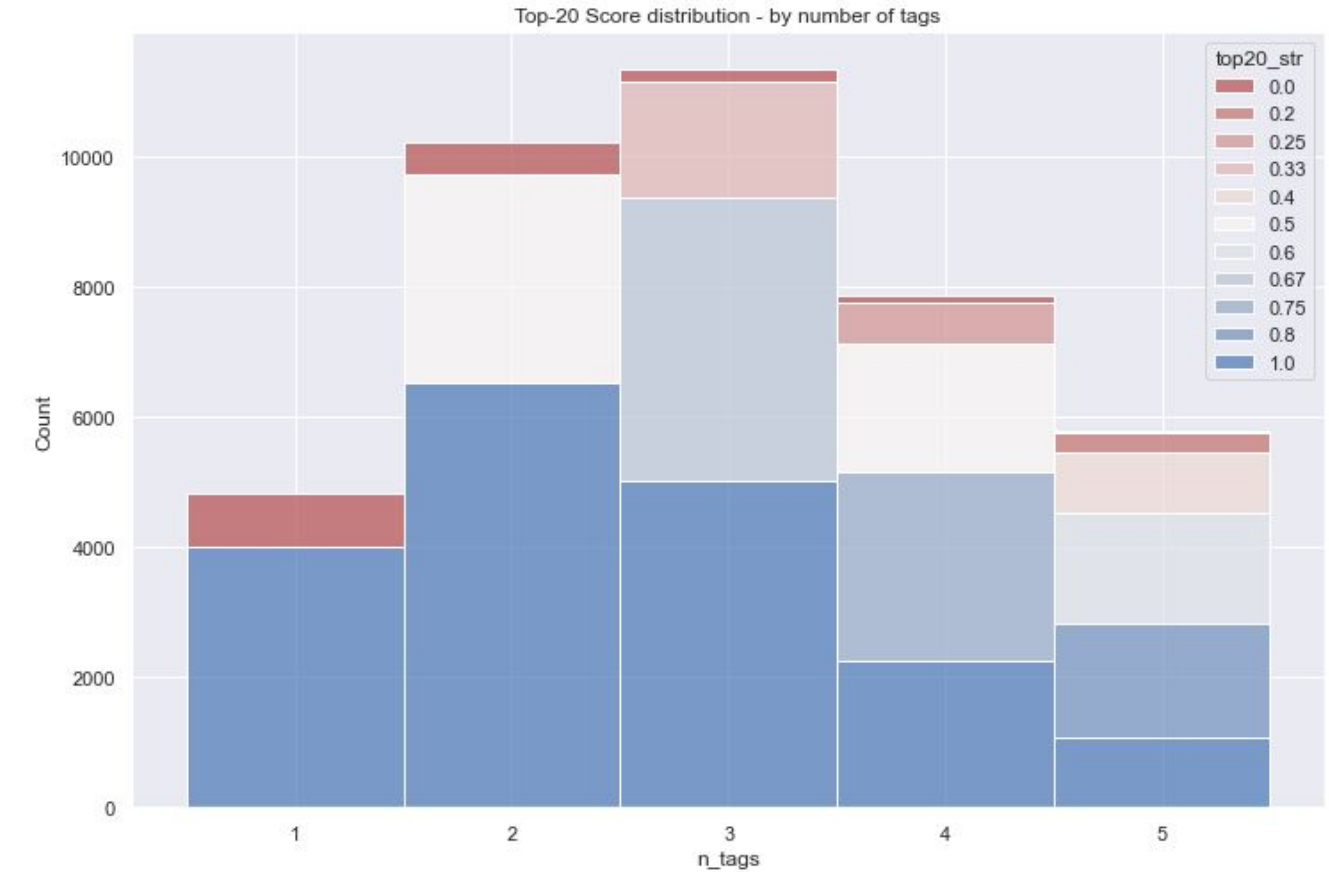
Validation metric: How well does the model do at suggesting 20 tags, and having the ‘true’ tags present in that list?

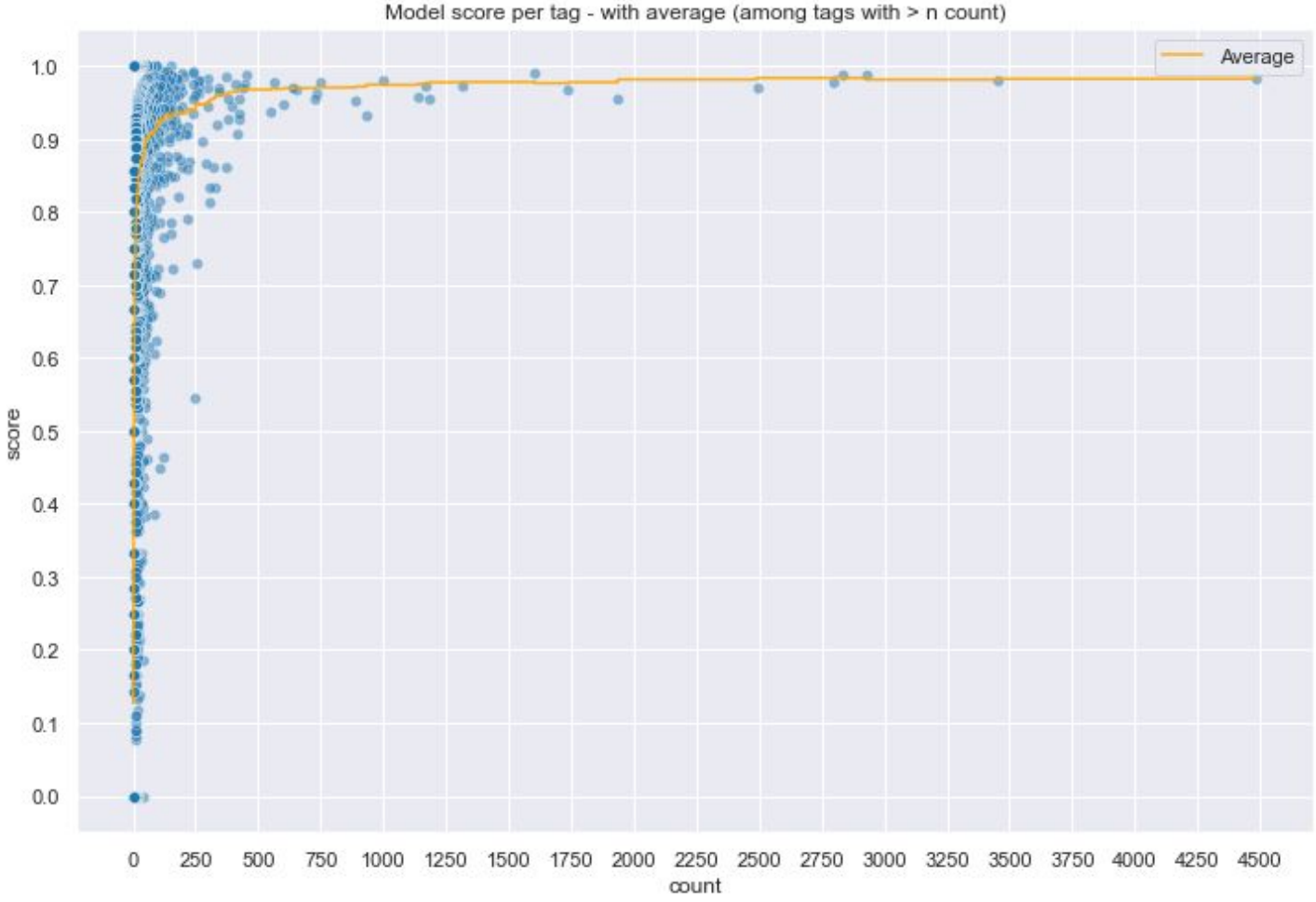
75% of ALL tags present in predicted top-20 lists

89% of model tags present

Average post score is 75%




Top-20 Score distribution - by number of tags
top20_str
0.0
0.2
0.25
0.33
0.4
0.5
0.6
0.67
0.75
0.8
1.0
Count
10000
8000
6000
4000
2000
0
1
2
3
4
5
n_tags


Model score per tag - with average (among tags with > n count)
Average
score
count
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
250
500
750
1000
1250
1500
1750
2000
2250
2500
2750
3000
3250
3500
3750
4000
4250
4500

Conclusions

Both models are successful in modelling the data, and the features seem feasible to implement

Quality

- Even with small dataset, high accuracy is achieved
- Wording of alert should balance respectfulness

Tag Suggestion

- By using a suggestion framework, as opposed to directly predicting a definitive list of tags, users can be reminded of tags and save time
- May also lead to higher # of tags per post

Limitations, Future Work

- More memory, computation resources
- Identify further modelling opportunities
    - Tailored / prescriptive quality alerts
    - Tag trend prediction / intersection with other datasets

Thank you!