Random Acts of Pizza

Mohammad J. Habib
W207

Classification problem

Predict whether a subreddit post will get a free Pizza given textual and numeric features

Coin toss results in AUC score of ~0.50

Guessing all False gives an accuracy score of ~0.75

Bag of words

- Features: Post, title, subreddit membership
- Extraction: CountVectorizer, TfidfVectorizer
- Stemming, Stop words, Frequent words, Reciprocity
- Unigrams, Bigrams, Trigrams

AUC Scores

- Naive Bayes: 0.55
- Log Reg: 0.51
- K-Neighbors: 0.51
- Tree Ensembles: 0.50
- Text Feature Unions: 0.50

Numeric features

- Number of downvotes, upvotes, comments, account age, post edited
- Meta-features: post length

AUC Scores

- Log Reg: 0.58
- Tree Ensembles: 0.58
- Gradient Boosting: 0.71

Combining text and numeric features

- Vectorize text features and Stack with numeric
- Perform PCA on text features and stack with numeric
- Binarization, various regularization approaches

AUC Scores

Text + Numeric:

- Gradient Boosting: 0.72
- Voting (LR, ET, GB): 0.51

PCA:

- Gradient Boosting: 0.51
- Gaussian Mixture: 0.50

PCA + Numeric:

- Gradient Boosting: 0.71

Discovering XGBoost

- XGBoost: Extreme Gradient Boosting
- Used for Supervised Learning
- Set of Classification and Regression Trees
  - Leaves have real prediction scores (not decision values)
  - Scores are summed to obtain prediction
- Additive training (one new tree at a time) and optimize loss function (e.g. logistic regression)
- Tune model complexity via regularization
- Score the structure of each trees
- Learn the structure

Final model

- Use xgboost's XGBClassifier
- Use Numeric features
- Randomly select Train data with equal parts True and False
- Use stacking to generate meta features (predicted probs)
    - Divide Train into two sets
    - Train an xgb on each and predict pobs
    - Combine predicted probs for Train data into a feature
    - Generate and combine predicted probs for dev/test data
    - Use data with meta features for final prediction
- Use Bagging Classifier with xgboost
- **Results on Test Data**: AUC: 0.79, Accuracy: 0.76