Predicting Expected Email Response Times

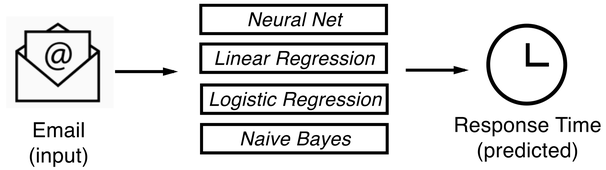
Laura Cruz-Albrecht (lcruzalb@stanford.edu) and Kevin Khieu (kkhieu@stanford.edu)
CS229: Machine Learning, Fall 2017

CS229

Predicting

Each day, countless emails are sent, yet the time in which one receives a response to an email can vary significantly. What if we could predict how long it takes for a recipient to reply to a user's email?

Goal:	**We will try to develop a model that predicts the time it takes for an email to receive a response.**



Findings: many uncapturable social factors play a role in expected email response time. However, there is a slight correlation between email attributes and response time.

Data

Dataset: Enron Email Dataset

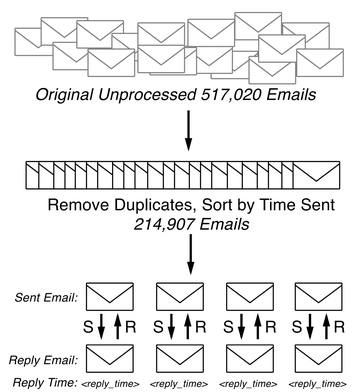
Statistics:
517,020 Total Emails
214,907 Unique Emails
19,739 Users

Schema of Data

- Email
  - Message-ID
  - FROM
  - TO
  - SUBJECT
  - X-CC
  - X-BCC
  - DATE
  - BODY
  - …

Dataset Preprocessing

Email chain extraction process



Features

Raw Input Features: *In deciding raw input features, we chose features that we hypothesized could influence users into replying more/less quickly. These features are:*

- Num. Recipients in "To" field
- Num. Words in Email
- Num. Recipients in "CC" field
- If Email is Reply
- Time of Day
- Day of Week
- Num. Words in Subject
- Num. "?" in Body/Subject
- If "?" Mark in Body/Subject
- If Keywords = ['response', 'please', 'can', 'urgent', 'important', 'need'] in Body/Subject

Derived Features: *Our dataset was proven to not be linearly separable, so we used an SVM that mapped features with a Gaussian RBF Kernel*

Models

Linear Regression
Ordinary Least Squares
Minimization via SGD

$$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x$$

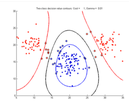
Logistic Regression
Binary + Multiclass
Maximizing Log Likelihood

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}$$

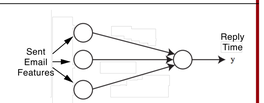
Naive Bayes
Binary + Multinomial
TF-IDF Standardization

$$\hat{y} = \underset{k \in \{1,\ldots,K\}}{\operatorname{argmax}}\ p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)$$

Support Vector Machine
Gaussian RBF Kernel +
Polynomial Kernel



Neural Network
Multilayer Perceptron
1-2 Hidden Layers (60x30)



Experiments and Results

Experiment #1: *Tested models on entire aggregated email dataset.*
Input = <Email_Features>
Output = Predicted Reply Time (nearest hour)
Training Size = 4,044
Test Size = 1,011

(1) Aggregated Dataset

	Training Accuracy	Test Accuracy
Linear Regression	0.061	0.059
Logistic Regression	0.368	0.362
Naïve Bayes	0.509	0.380
Neural Network	0.322	0.319

Takeaways:

- Heavy skew in our data: all models except Linear Regression predict the most common label
- Dataset not linearly separable

Experiment #2: *Used a "balanced" dataset where we cap the number of <email, reply time> pairs for each hour to 50*
Input = <Email_Features>
Output = Predicted Reply Time (nearest hour)
Training Size = 955
Test Size = 239

(2) Balanced Dataset

	Training Accuracy	Test Accuracy
Linear Regression	0.055	0.055
Logistic Regression	0.137	0.097
Naïve Bayes	0.509	0.008
Neural Network	0.100	0.066

Takeaways:

- Poor classification even with balanced dataset
- Potentially insufficient data for 25 labels

Experiment #3: *Used a balanced dataset where we "binarize" our labels into classes of reply times < 1/2 hr and reply times >= 1/2 hr*
Input = <Email_Features>
Output = 1 if reply time > 30 mins, 0 otherwise
Training Size = 2,896
Test Size = 724

(3) Binarized Dataset

	Training Accuracy	Test Accuracy
Linear Regression	0.536	0.526
Logistic Regression	0.543	0.531
Naïve Bayes	0.502	0.500
SVM	0.871	0.558
Neural Network	0.777	0.590

Takeaways:

- Slightly better than random performance on all models.
- SVM and NN perform best. Both are overfitting

Experiment #4: *Chose two most promising models and trained with same dataset as prior, but with improved feature set*
Input = <Email_Features>
Output = 1 if reply time > 30 mins, 0 otherwise
Training Size = 2,896
Test Size = 724

(4) Binarized Dataset + Improved Features

	Training Accuracy	Test Accuracy
SVM	0.773	0.560
Neural Network	0.828	0.570

Takeaways:

- Heavy overfitting by neural network, less overfitting by SVM

Discussion

Our investigation indicates that the problem of email response time prediction is difficult: in the multilabel space, even after balancing the labels of our skewed initial dataset, prediction accuracy across models was low. The classification problem was still challenging when reduced to one of binary prediction among balanced classes. The low accuracies across models is understandable however: **there are a lot of social factors that cannot be detected in the metadata of an email** that impact how quickly a person chooses to, or is able to, respond. Nevertheless, we did find that the **Neural Net and SVM Models did perform at above-random levels**, indicating that there is some degree of correlation between features of an email, and the likelihood of receiving a quick response within the work setting.

Future Work

Given more time, the following is a list of future directions for this project:

- **gathering more diverse data** would boost the robustness of our models
- further **feature extraction investigation** could be done to find more indicative features
- **incorporating each user's history of reply behavior**, which would require gaining user permission, would greatly aid in our work

References

Ayodele et. al. Applying Machine Learning Techniques for Email Reply Prediction. 2009. http://www.iaeng.org/publication/WCE2009/WCE2009_pp31-36.pdf

Drezde et. al. Intelligent Email: Reply and Attachment Prediction. 2008. https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/34947.pdf

Yang et. al. Characterizing and Predicting Enterprise Email Reply Behavior. 2017. https://www.microsoft.com/en-us/research/wp-content/uploads/2017/04/sigir17a.pdf

Dredze et. al. Reply Expectation Prediction for Email Management. http://www.seas.upenn.edu/~strctlrn/bib/PDF/ceas05.pdf"How quickly will your email get a response?" https://www.theatlantic.com/technology/archive/2015/10/how-quickly-will-your-email-get-a-response/409429/

"7 Proven Ways to Write Emails That Get Replies, Backed By Science". https://www.vesware.com/bloa/emails-that-aet-replies/