Email Prioritization and Response Recommendation

Shreyas Muralidhara(schikkb)
Sharath Narayana(snaraya9)

Introduction

- Emails have become the primary mode of communication in schools, colleges, professional industries and other such organizations.

- With the humongous influx of messages everyday it's important to prioritize the emails which should be responded first.

- This feature with an automatic response recommendation that would help in cleaning mail faster.

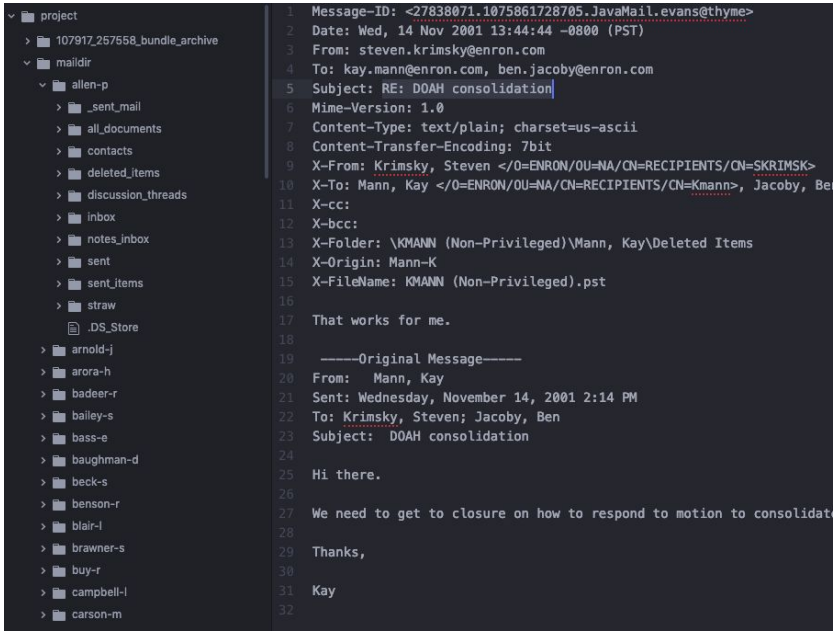
Implementation

We decided to approach the problem in 5 steps.

- Data Pre-processing
- Annotation
- Data Augmentation
- Email Ranking
- Recommendation System

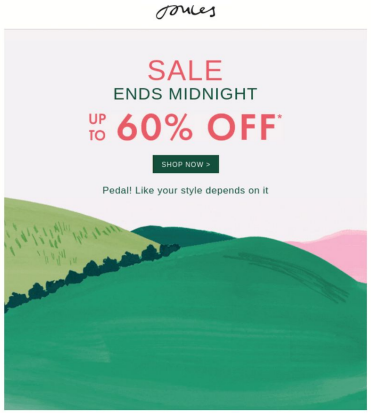
Data Pre-Processing

- We have selected Enron Email Dataset for our project.
- The text files where traversed one by one and each of the text file was scraped for data.
- Regular expressions was used to match the patterns.
- Date had multiple formats which had to be handled separately.
- Missing data points were sometimes tried to match with previous found one and few were ignored.



Annotation

- For recommendation system, we needed annotated email feature list.
- The class labels includes **delete, reply and thread**.
- Delete means that the message is either no longer relevant or it contains details which are out of scope for the user.
- Reply means that the message has an action item which requires the user to respond the email.
- Thread email is one which goes to the inbox and no specific action is required from the user.



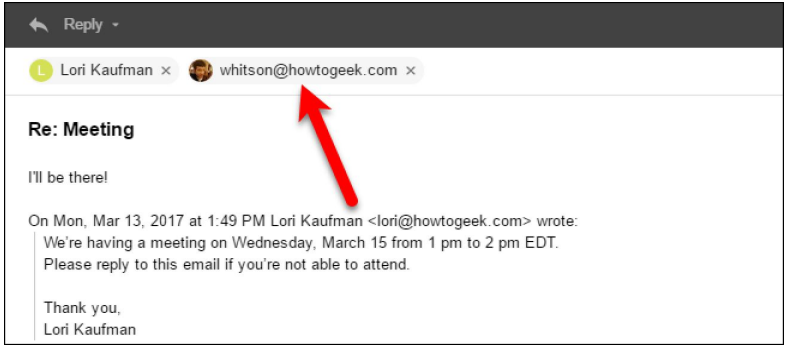


Image References
https://www.howtogeek.com/298780/how-to-change-the-reply-to-address-for-email-messages-in-outlook/
https://actumarketing.com/%F0%9F%94%B5%E2%9A%AA%F0%9F%94%B4-how-to-write-promotional-emails-with-examples-landing-page/

Data Augmentation

- To improve information from the dataset we had to further augment the data.
- We performed lemmatization, stemming and POS tagging.
- Subject and Content was further cleaned to remove stopwords, punctuations and special characters.
- NLTK library was used to this augmentation.

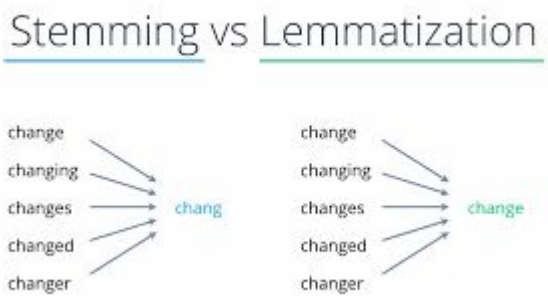
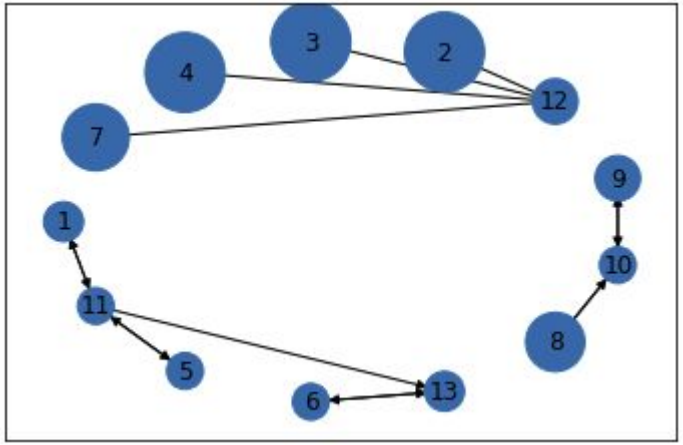


Image References
https://medium.com/swlh/introduction-to-stemming-vs-lemmatization-nlp-8c69eb43ecfe

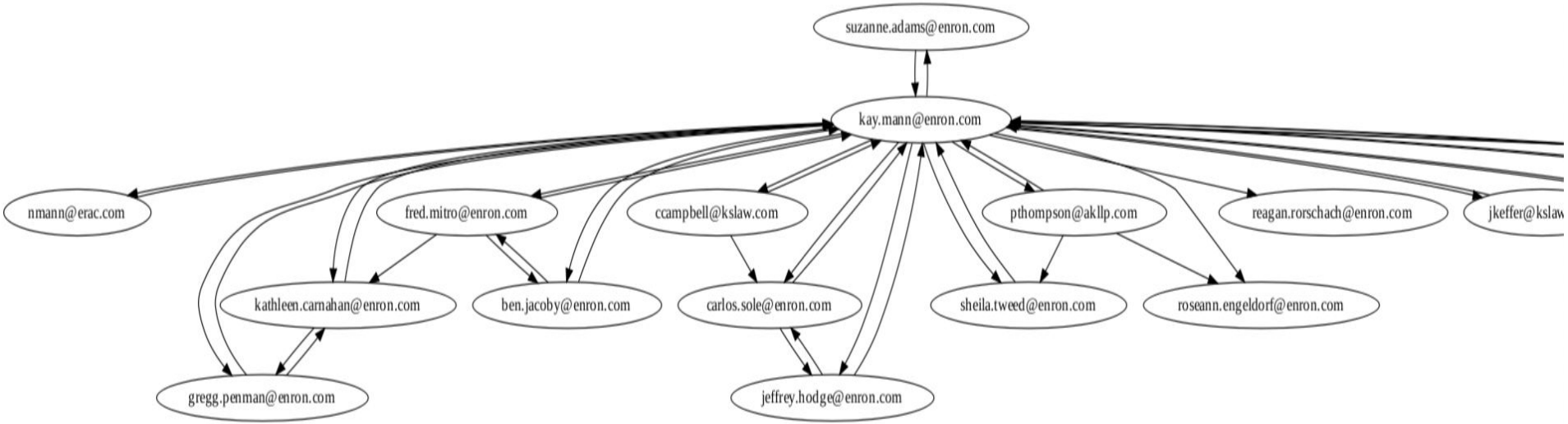
Email Ranking

- PageRank is an algorithm used to calculate the weight for web pages.
- Variant of this algorithm where each node would be the user email
- User interactions, time between the interactions, the content of the email and repetitions between the users was used to calculate the weight.
- The weight matrix is multiplied with the node adjacency matrix to calculate the email weight.
- Framework was built to add(decrease/increase) more weights to the email

$$S(V_i) = (i - d) * d * \sum_{j \in ln(v_i)} S(V_j)$$



Email DiGraph



Response Recommendation

- Recommendation system suggests the type of response for a given email.
- Two models for identifying the response type was implemented.
- Tf-Idf considers the features From, To, Email date, Subject, email content for generating count vector matrix.
- Doc2Vec embedding model generates the individual tagged documents for the features From, To, Email date, Subject and Content
- The weights associated with each of the features was updated to capture the Subject and Corpus in higher vector space, From and To feature vectors remaining the same, and the Date vector space reduced to lowest priority
- Models Trained
    - TFIDF (Naive Bayes, KNN, Random Forest)
    - Doc2Vev (SVM with radial basis kernel)

Results

suzanne.adams@enron.com : 2.049069306501175e-05
nmann@erac.com : 7.210027058583903e-06
kathleen.carnahan@enron.com : 1.978249540900212e-06
carlos.sole@enron.com : -6.663852482585805e-07
ben.jacoby@enron.com : -2.7360994310785023e-06
sheila.tweed@enron.com : -6.703051614816691e-06
ccampbell@kslaw.com : -8.48530549446718e-06
pthompson@akllp.com : -9.865114949680462e-06
reagan.rorschach@enron.com : -1.0037591131582122e-05
roseann.engeldorf@enron.com : -9.980099070948235e-06
jkeffer@kslaw.com : -1.1014956162358196e-05
gregg.penman@enron.com : -1.130241646552763e-05
heather.kroll@enron.com : -1.1992321193134274e-05
kay.mann@enron.com : 1.0001783638915656
nwodka@bracepatt.com : -1.3774575072784762e-05
kathleen.clark@enron.com : -1.4234511557855856e-05
kent.shoemaker@ae.ge.com : -1.4234511557855856e-05
fred.mitro@enron.com : -1.4062035375954196e-05
jeffrey.hodge@enron.com : -1.469444804292695e-05

TFIDF

```
Classification Model 2 - Random forest Classifier Tf-Idf TEST metrics:
Accuracy - 0.4313
f1 score - 0.4316
Classification Report:
              precision    recall  f1-score   support

      delete       1.00      0.91      0.95        45
       reply       0.39      0.39      0.39      1109
      thread       0.45      0.45      0.45      1239

    accuracy                           0.43      2393
   macro avg       0.61      0.58      0.60      2393
weighted avg       0.43      0.43      0.43      2393
```

```
Classification Model 2 - Naive Bayes Tf-Idf TEST metrics:
Accuracy - 0.5357
f1 score - 0.5148
Classification Report:
              precision    recall  f1-score   support

      delete       0.00      0.00      0.00        45
       reply       0.52      0.35      0.42      1109
      thread       0.54      0.72      0.62      1239

    accuracy                           0.54      2393
   macro avg       0.35      0.36      0.35      2393
weighted avg       0.52      0.54      0.51      2393
```

Doc2Vec

```
Classification Model 1 - SVM - Doc2Vec TEST metrics:
Accuracy - 0.6987
f1 score - 0.6776
Classification Report:
              precision    recall  f1-score   support

      delete       1.00      0.04      0.09        45
       reply       0.62      0.97      0.76      1109
      thread       0.90      0.48      0.63      1239

    accuracy                           0.70      2393
   macro avg       0.84      0.50      0.49      2393
weighted avg       0.77      0.70      0.68      2393
```

Regularization Factor = 1

```
Classification Model 1 - SVM - Doc2Vec TEST metrics:
Accuracy - 0.6753
f1 score - 0.6424
Classification Report:
              precision    recall  f1-score   support

      delete       0.00      0.00      0.00        45
       reply       0.60      1.00      0.75      1109
      thread       0.95      0.41      0.57      1239

    accuracy                           0.68      2393
   macro avg       0.51      0.47      0.44      2393
weighted avg       0.77      0.68      0.64      2393
```

Regularization Factor = 0.1

Conclusions and Future Work

- We experimented email ranking which is one of its kind and created a framework to extend the ranking algorithms weights.
- The recommendation system was trained on various models and we got the following accuracy. We can see that SVM with RBF kernel of 1 gave the highest accuracy.
- Future scope of this project is that it can be experimented on various ranking algorithms.
- Neural network models can be trained for the recommendation systems to get better accuracy.

References

- Towards Explainable NLP: A Generative Explanation Framework for Text Classification, Hui Liu, Qingyu Yin, William Yang Wang, 2018, arXiv:1811.00196, https://arxiv.org/abs/1811.00196
- Yoo, S. (2010). Machine learning methods for personalized email prioritization (Doctoral dissertation, Carnegie Mellon University).
- Conway, D. and White, J.M., 2011, January. Machine Learning for Email: Spam Filtering and Priority Inbox isbn-9781449320706.
- Ha, Minh & Tran, Quang & Luyen, Thu. (2012). Personalized Email Recommender System Based on User Actions. 280-289. 10.1007/978-3-642-34859-4_28.
  Page Rank Algorithm Reference https://github.com/bhaveshgawri/PageRank
- Improving Email Response in an Email Management System Using Natural Language Processing Based Probabilistic Methods - https://www.researchgate.net/profile/Abdulkareem_Al-Alwani/publication/283028323
- Hasegawa, Takaaki & Ohara, Hisashi. (2000). Automatic Priority Assignment to E-mail Messages Based on Information Extraction and User's Action History. 573-582. 10.1007/3-540-45049-1_69.

Thank You

Q and A?