Characterizing and Predicting Enterprise Email Reply Behavior

Liu Yang[1] Susan T. Dumais[2] Paul N. Bennett[2] Ahmed Hassan Awadallah[2]
[1] Center for Intelligent Information Retrieval, University of Massachusetts Amherst, Amherst, MA, USA
[2] Microsoft Research, Redmond, WA, USA
lyang@cs.umass.edu,{sdumais,pauben,hassanam}@microsoft.com

ABSTRACT

Email is still among the most popular online activities. People spend a significant amount of time sending, reading and responding to email in order to communicate with others, manage tasks and archive personal information. Most previous research on email is based on either relatively small data samples from user surveys and interviews, or on consumer email accounts such as those from Yahoo! Mail or Gmail. Much less has been published on how people interact with enterprise email even though it contains less automatically generated commercial email and involves more organizational behavior than is evident in personal accounts. In this paper, we extend previous work on predicting email reply behavior by looking at enterprise settings and considering more than dyadic communications. We characterize the influence of various factors such as email content and metadata, historical interaction features and temporal features on email reply behavior. We also develop models to predict whether a recipient will reply to an email and how long it will take to do so. Experiments with the publicly-available Avocado email collection show that our methods outperform all baselines with large gains. We also analyze the importance of different features on reply behavior predictions. Our findings provide new insights about how people interact with enterprise email and have implications for the design of the next generation of email clients.

KEYWORDS

Email reply behavior; information overload; user behavior modeling

1 INTRODUCTION

Email remains one of the most popular online activities. Major email services such as Gmail, Outlook, and Yahoo! Mail have millions of monthly active users, many of whom perform frequent interactions like reading, replying to, or organizing emails. A recent survey shows that reading and answering emails takes up to 28% of enterprise workers' time, which is more than searching and gathering information (19%), communication and collaboration internally (14%), and second only to role specific tasks (39%) [6]. Understanding and characterizing email reply behaviors can improve communication and productivity by providing insights for the design of the next generation of email tools.

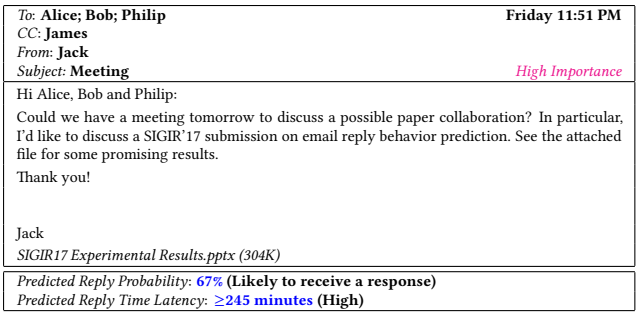


Figure 1: A motivational email example with predicted reply probability and reply time latency.

By modeling user reply behaviors like reply rate and reply time, we can integrate machine intelligence into email systems to provide value for both email recipients and senders. For email recipients, reply predictions could help filter emails that need replies or fast replies, which can help reduce email overload [9]. For email senders, the reply behaviors could be predicted in advance during email composition. More generally, better reply strategies could lead to improved communication efficiency. Figure 1 shows a motivating email example with predicted reply probability and reply time latency shown in the bottom panel. Specific features could also be highlighted. For example, identifying a request in the email (*Could we have a meeting...*) could improve automated triage for the recipient by highlighting that a reply is needed; or alerting the sender that a reply is likely to take longer if it is sent late at night or over the weekend could improve communication efficiency.

Previous work investigated strategies that people use to organize, reply to, or delete email messages [9, 10, 22, 29]. However, those studies are based on relatively small surveys or interviews. Some recent research proposes frameworks for studying user actions on emails with large scale data [11, 20]. Both of these studies are based on consumer emails from Yahoo! Mail. Enterprise email has received little attention compared to consumer email even though several studies have shown that enterprise email usage is not the same as consumer email usage. For example, [25] reports that enterprise users send and receive twice as much emails as consumer users and [15] shows that consumer email is now dominated by machine-generated messages sent from business and social networking sites.

Perhaps the closest prior research to our work is the study on email reply behavior in Yahoo! Mail by Kooti et al. [20]. However, they focus on personal email and only consider a subset of email exchanges, specifically those from dyadic (one-to-one) email conversations for pairs of users who have exchanged more than 5 prior

Work primarily done during Liu Yang's internship at Microsoft Research.

SIGIR'17, August 7–11, 2017, Shinjuku, Tokyo, Japan

DOI: http://dx.doi.org/10.1145/3077136.3080782

emails in consumer email. Focusing only on dyadic email conversations is limiting, especially in the context of enterprise emails. In the enterprise email data that we study, 52.99% of emails are non-dyadic emails, that is, they are sent to more than one recipient other than the sender. Thus it is important and more realistic to study the more general setting of modeling email reply behavior including both dyadic emails and emails sent to a group of people, without any threshold on previous interactions.

In this paper, we address this gap by characterizing and predicting reply behaviors in enterprise emails, where we consider both dyadic conversations and group discussions. We use the publicly-available Avocado research email collection,[1] which consists of emails and attachments taken from a defunct information technology company referred to as "Avocado". There are 938, 035 emails from 279 accounts in this email collection.

We analyze and characterize various factors affecting email replies including: temporal features (e.g. time of day and day of week), historical interaction features (e.g., previous interactions between sender and recipients), properties of the content features (e.g. length of subject and email body), predictions based on textual content features (e.g., sentiment, contains a request), address features (e.g., recipients), and metadata features (e.g., attachments). We find several interesting patterns connecting these factors to reply behavior. For example, emails with requests or commitments get more but slower replies while longer emails tend to get fewer and slower replies. Based on this analysis, we used a variety of features to build models to predict whether an email will receive a reply and the corresponding reply time. We show that our proposed model outperforms all baselines with large gains. We also perform feature importance analysis to understand the role different features play in predicting user email reply behavior.

Our contributions can be summarized as follows:

(1) We introduce and formalize the task of reply behavior prediction in enterprise emails involving both one-to-one (dyadic) and one-to-many communication. Unlike previous work either on small user surveys and interviews [9, 10, 22, 29] or only for dyadic email conversations in consumer emails [20], our work is the first to model reply behavior in a more general setting including emails sent to groups of people as well as individuals for enterprise email.
(2) We analyze and characterize various factors affecting email replies. Compared to previous work, we study many novel factors including email textual content, request / commitment in emails, address features like internal/external emails, and number of email recipients.
(3) We extract 10 different classes of features and build models to predict email reply behaviors. We perform thorough experimental analysis with the publicly-available Avocado email collection and show that the proposed methods outperform all baselines with large gains. We also analyze the importance of each class of features in predicting email reply behavior.

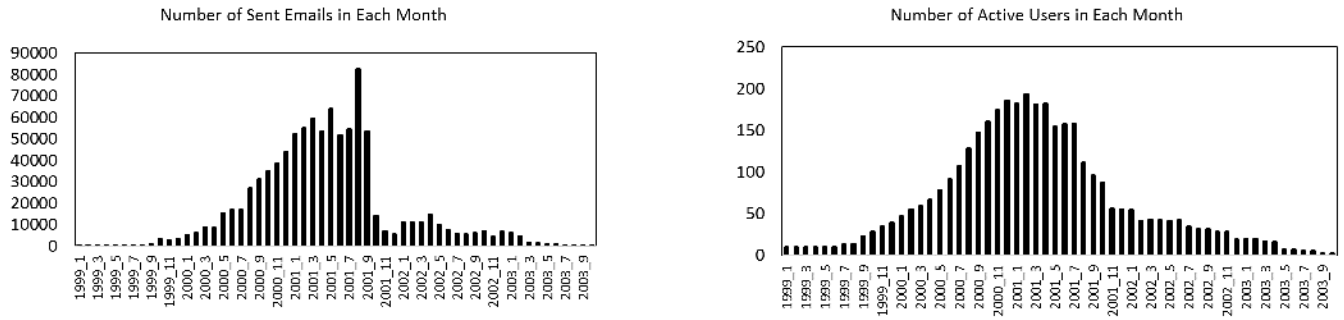
[1] https://catalog.ldc.upenn.edu/LDC2015T03

2 RELATED WORK

Our work is related to several research areas, including modeling actions on email, email overload, email acts and intent analysis, email classification and mining.

Modeling Actions on Email. Our work is related to previous research on user behavior and email action modeling [10, 11, 20, 23, 24, 28]. Dabbish et al. [10] examined people's ratings of message importance and the actions they took on specific email messages with a survey of 121 people at a university. While this work provides insights on email usage and user actions on email messages, how well the results generalize to other user groups is not clear. DiCastro et al. [11] studied four common user actions on email (read, reply, delete, delete-withoutread) using an opt-in sample of more than 100k users of the Yahoo! Mail service. They proposed and evaluated a machine learning framework for predicting these four actions. Kooti et al. [20] also used Yahoo! Mail data to quantitatively characterize the reply behavior for pairs of users. They investigated the effects of increasing email overload on user behaviors and performed experiments on predicting reply time, reply length and whether the reply ends a conversion. Our work is inspired by the latter two studies but it differs in several important ways. These two studies looked at behavior in Yahoo! Mail, a consumer email collection, whereas we studied interaction in a enterprise setting using the Avocado collection. What's more, the study by Kooti et al. [20] only considers dyadic emails from a subset of people who had at least five previous email interactions. Our work considers a more general setting where we consider both dyadic emails and emails sent to a group of users. We allow cases where there is no previous interactions between the sender and receivers, which makes our prediction task a more challenging (and realistic) one. Our experimental data is also publicly available in the recently-released Avocado collection from LDC, whereas prior research used proprietary internal data. Last but not least, we analyzed several novel features including properties of the content like email subject and body length, predictions of whether the email contained a request or commitment, and address features like internal vs. external emails and showed that they are useful for building models to predict user email reply behavior.

Email Overload. Several research efforts have examined the *email overload* problem and proposed solutions to reduce it [2, 9, 12, 22, 29]. In their pioneering work, Whittaker and Sidner [29] explored how people manage their email and found that email was not only used for communication, but also for task management and maintaining a personal archive. A decade later, Fisher et al. [12] revisited the email overload problem by examining a sample of mailboxes from a high-tech company. They showed that some aspects of email dramatically changed, such as the size of archive and number of folders, but others, like the average inbox size remained more or less the same. Several researchers have proposed solutions to mitigate the email overload problem [1, 2, 13, 18]. Aberdeen et al. [1] proposed a per-user machine learning model to predict email "importance" and rank email by how likely the user is to act on that mail; this forms the the Priority Inbox feature of Gmail. Our work shares similar motivations for handling the email overload problem by modeling and predicting user reply behaviors on emails. We focused on email reply behaviors, specifically identifying emails



(a) The number of sent emails in each month. **(b) The number of active users in each month.**

Figure 2: Temporal analysis of the number of sent emails and active users in each month of the Avocado email collection.

which receive replies and the reply latency. Their work looked at consumer emails, Gmail, whereas we focus on the enterprise email collection, Avocado.

Email Acts and Intent Analysis. Previous research studied email acts and email intent analysis [3, 7, 21, 26, 27]. Cohen et al. [7] proposed machine learning methods to classify emails according to an ontology of verbs and nouns, which describe the "email speech act" intended by the email sender. Follow-up work by Carvalho and Cohen [4] described a new text classification algorithm based on a dependency-network-based collective classification method and showed significant improvements over a bag-of-words baseline classifier. In a recent study using the Avocado collection, Sappelli et al. [26] studied email intent and tasks, and proposed a taxonomy of tasks in emails. They also studied predicting the number of tasks in emails. Our work extends previous research on email acts and intent by extracting requests and commitments from emails and using them as features for predicting user reply behavior.

Email Classification and Mining. We formalized the user email reply behavior prediction task as a classification task. There are many previous papers on email classification and mining [5, 14–16, 19]. Klimt and Yang [19] introduced the Enron corpus as a data set and used it to explore automated classification of email messages into folders. Graus et al. [14] studied the task of recipient recommendation. They also used enterprise email collections, but their collection was proprietary and their focus was on recipient recommendation, rather than reply behavior modeling.

To summarize, the research described in this paper extends previous research on email interaction in several ways. Using a recently released public enterprise email collection, we formalize the task of prediction whether an email will be responded to and how long it will take to do so. In contrast to earlier work on reply prediction, we study enterprise (vs. consumer) email behavior, consider both one-to-one (dyadic) and one-to-many emails, and develop several new types of features to characterize the email content and intent and study their importance in reply prediction.

3 DATA SET & TASK DEFINITION

Our data set is the Avocado research email collection from the Linguistic Data Consortium. This collection contains corporate emails from a defunct information technology company referred to as "Avocado". The collection contains the full content of emails, various meta information as well as attachments, folders, calendar entries, and contact details from Outlook mailboxes for 279 company employees.

The full collection contains 938, 035 emails and 325, 506 attachments. Since our goal is to study email reply behavior, we need to preprocess the data to figure out the reply relationships between emails. For emails that are replies, the email meta information includes a "reply_to" relationship field, containing the ID of the email that this email is replying to. We generate the email thread structure by parsing the "reply_to" relationship fields in email meta data. Specifically, we first use the "reply_to" relationship fields to collect sets of messages that are in the same thread. Then emails in the same thread are ordered by the sent time to generate their thread positions. We removed all the duplicated quoted "original messages" in the email text files to identify the unique body for each message. From the "reply_to" relationship fields and associated email sent time, we generate the ground truth for reply behavior including whether the email received a reply, and reply latency.

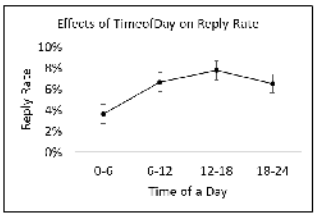
We first perform a temporal analysis of sent emails and active users in each month. Figure 2a and Figure 2b show the number of sent emails and the number of active people in this collection respectively. Emails are aggregated by the sent time into different months. Each person is represented by an interval based on the sent time of his/her first and last email. Then the number of active people in each month is the number of overlapping intervals during the month. We see that the number of active people increases to a peak in February 2001 and then decreases to under 50 after February 2002, as the company started laying off employees. The number of emails sent has a similar pattern over months. To ensure that our analysis and experimental results are built upon emails when the company was under normal operations, we select the emails from June 1st 2000 to June 1st 2001.

We further perform a few additional data cleaning and filtering steps. We focus on the reply behaviors toward the first message in each thread. Because replies towards the first messages and follow-up replies may have different properties and we want to focus on one phenomenon in this study. We filter out email threads where the first reply comes from the sender himself or herself and emails where the only recipient is the sender. After these filtering

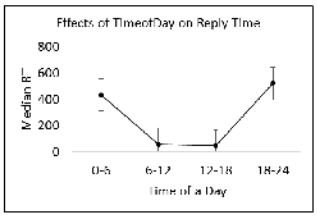
steps, we have 351,532 emails left, which become the basis of our data analysis.[2]

The email behavior we study includes the reply action and reply time. We formally define the task as following. Given an email message m that user u_i sent to a user set $\{u_j\}$ at time t, we study: (1) **Reply Action**: whether any one of users in $\{u_j\}$ will reply to message m; (2) **Reply Time**: how quickly users in $\{u_j\}$ reply, which is the difference between the first response time and the start time t. If there are multiple replies from users in $\{u_j\}$, we consider only the time latency corresponding to the first reply.

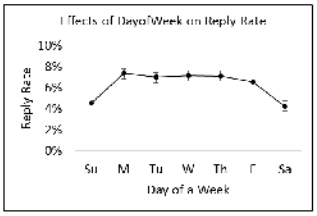
Thus our setting includes both "one-to-one" and "one-to-many" emails, which is more general than [20] which only included dyadic emails between pairs of users with at least five previous emails.



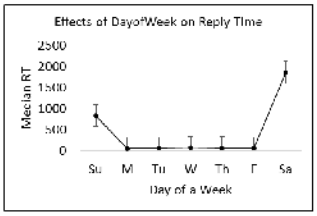
(a) Reply rate vs. time of day.



(b) Reply time vs. time of day.



(c) Reply rate vs. day of week.



(d) Reply time vs. day of week.

Figure 3: The effects of temporal features of the original sent emails on user email reply behavior. The median reply time denoted by Median RT is in minutes.

4 CHARACTERIZING REPLY BEHAVIOR

We characterize multiple factors affecting email reply action and reply time. For reply action, we compute the reply rate which is the percentage of emails that received replies. [3] In Section 5 we show that these factors enable us to learn models to predict user reply behavior.

4.1 Temporal Factors

We first study the impact of temporal factors on user replies. Figure 3a and Figure 3b show the reply rate and median reply time to emails sent at different times of the day. We partition the time of a day into Night (0 to 6), Morning (6 to 12), Afternoon (12 to 18) and Evening (18 to 24). Then we aggregate emails in each time range and compute the reply rate and median reply time. The unit of reply time is in minutes. We see that emails sent in morning and afternoon receive more and faster replies. For reply rate, emails sent in the afternoon have the highest reply rate (7.77%) and emails sent at night have the lowest reply rate (3.63%). For reply time, the median reply time for morning and afternoon is less than 1 hour, whereas reply time for mails sent in evening and night is more than 7 hours. This makes sense since users are more active during the day than night time.

Figure 3c and Figure 3d show the reply rate and reply time to emails sent on different days of a week. We see that emails sent on weekdays receive more and faster replies. The reply rate for emails sent on weekdays is around 7%, but that rate drops to 4% on weekends. For the median reply time, we see that emails sent on weekdays receive replies in less than 1 hour. However, emails sent on the weekend have 13 (for Sunday) or 30 (for Saturday) times longer reply latency. This is consistent with the fact that most people don't check or reply to emails as regularly on the weekend. Most emails sent over the weekend are replied to after Sunday.

4.2 Email Subject and Body

Next we study the effects of properties of the email content on user reply behavior. We start with the length of email subjects and email bodies. We remove punctuations and maintain stop words when counting the number of words. Figure 4a and Figure 4b show the effects of email subject length on user reply behavior. The reply rate decreases from 11.47% to 1.10% as the length of email subject increases from 1 to 29, which is an interesting phenomenon. Because we have access to the full content of emails in Avocado, we were able to examine emails that have long and short subjects to identify. We found that many long subjects include announcements in the subjects but no text in the body or some machine generated bug reports. Such email subjects are similar to "*[Bug 10001] Changed - Loop length not assigned correctly for xml file when extracted to a file and tested, works OK when tested from database.*". [4] Users don't need to reply to such emails, leading to lower reply rate.

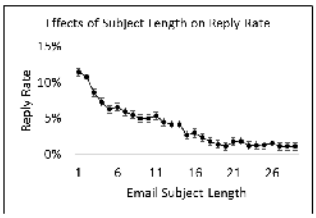
In examining short subjects, we computed the reply rates for all unique subjects. Checking those that occur 20 times or more, we summarize those that have the highest reply rate. A large number of these are simply company names – these often indicate current contracts, customers, or sales leads that are being pursued. Other phrases such as "*expense report*", "*sales training*", "*meeting*", "*lunch*", and "*alerts*" indicate common activities that require approval, coordination, or action. Finally, a set of other phrase types such as "*hello*", "*hi*", "*hey*", etc. indicate recipient-sender familiarity (note that spam e-mail filters were in place before collection). We also find that reply time is influenced by subject length. The main trend is that the reply time decreases as the subject length increases.

We also analyze the impact of the email body length on reply behavior. Figure 4c and Figure 4d show the effects of email body length on reply behavior. We see that the reply rate initially increases from 4.65% to 9.64% words as the email length increases from 1 to 40 words and then decreases to 2.82% as the email body length increases to 500 words. Thus people are not likely to reply to emails with too few or too many words. The median reply time increases as the length of the email body increases. This may be

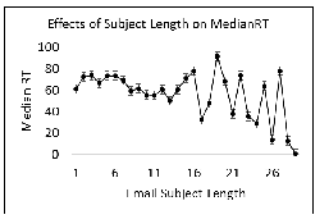
[2] The email IDs of the filtered subset can be downloaded from https://sites.google.com/site/lyangwww/code-data.

[3] For partitions of the data we compute the reply rate for each partition separately. For example, the reply rate to emails sent at night is the percentage of emails sent at night that receive a reply regardless of whether the reply is sent at night or day.

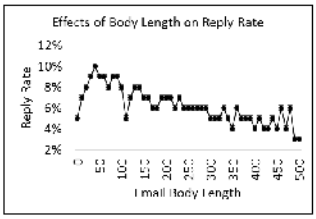
[4] We paraphrase this sentence due to data sensitivity.



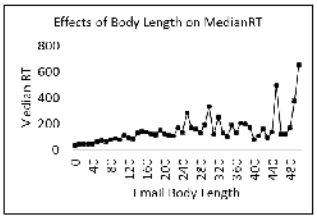
(a) Reply rate vs. subject length.



(b) Reply time vs. subject length.



(c) Reply rate vs. body length.



(d) Reply time vs. body length.

Figure 4: The effects of email subject length and email body length on reply rate and median reply time. The median reply time denoted by Median RT is in minutes.

Table 1: The effects of requests and commitments in emails on user reply behavior. The median reply time denoted by Median RT is in minutes.

Factor	Requests		Commitments	
Email Feature	HasReq	NoReq	HasCom	NoCom
Reply Rate	14.81%	7.45%	8.32%	8.78%
Median RT	81.13	54.51	86.71	60.60

because people need to spend more time reading and digesting the content of longer emails, thus increasing reply time.

4.3 Requests and Commitments in Emails

Previous work [3, 4, 7, 21] has studied "speech acts" in emails. Carvalho and Cohen [4] developed algorithms to classify email messages as to whether or not they contain certain "email acts", such as a request or a commitment. We classify emails into those with requests (e.g. "Send me the report.") or commitments (e.g. "I'll complete and send you the report.") and those without requests or commitments using an internally developed classifier inspired by previous work in this area [8].

Table 1 shows the effects of whether an email contains a request or commitment on reply rate and reply time. Emails that contain requests are almost twice as likely to receive replies as those that do not, 14.81% vs. 7.45% respectively. This is reasonable since intuitively people are more likely to reply to emails if the sender makes a request. In contrast, there is very little differences between the reply rate toward emails with and without commitments.

The median reply time toward emails with requests or commitments is longer, which may seem a bit counter-intuitive. However, this may be because such emails are associated with tasks, and therefore, people may need to do some work like searching information, reading or writing before they can reply to such a mail.

Table 2: The effects of email addresses and attachments on user reply behavior. The median reply time denoted by Median RT is in minutes.

Factor	Email Addresses		Attachments	
Email Feature	Internal	External	HasAttach	NoAttach
Reply Rate	7.76%	2.26%	8.08%	6.38%
Median RT	65.52	134.67	80.37	61.23

4.4 Internal / External Email Addresses

Next we investigate the impact of internal or external emails on reply rate and reply time. To do this, we adopt a heuristic method to classify the email addresses. For each receiver address, we check whether there is a "@" in it. If a receiver address contains "@" and does not contain "@avocadoit.com", this address is assigned an "external" label. If there is no "@" or only "@avocadoit.com" in a receiver address, we classify it as an internal address. Email addresses with "@avocadoit.com" are definitely internal addresses. However, a limitation of this method is that if email addresses with external domains are in the sender's contacts, they will be treated as internal addresses. In this sense, the internal email addresses in our analysis are those of Avocado employees or people who have frequent communications with Avocado employees and are stored in the contact books. Emails sent to at least one external address are labeled as external, otherwise they are treated as internal emails.

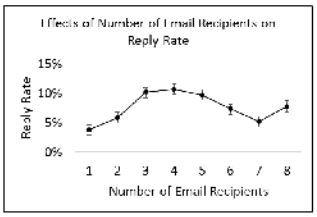
Table 2 shows the comparison of the reply rate/ reply time for internal emails and external emails. People are 3.4 times more likely to reply to internal emails (7.76%) than external emails (2.26%). In addition median reply time to internal emails is 2.1 times faster than to external emails, 66 vs. 135 minutes respectively. In our context, internal emails come from the colleagues or people in the sender's contacts, and are thus more likely to be quickly replied to.

4.5 Email Attachments

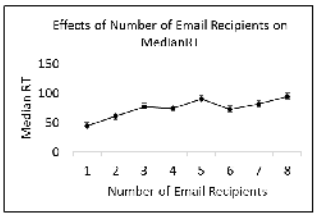
We further analyze the effects of email attachments on user reply behaviors. Table 2 shows that reply rates are higher for emails with attachments (8.08%) than emails without attachments (6.38%). This may be because the types of emails that contain attachments are more likely to require replies. The median reply time for emails with attachments is 23.81% longer than that for emails with no attachments. This result is consistent with the fact that people need to open the attachments and spend more time reading or replying to emails with attachments.

4.6 Number of Email Recipients

Unlike [20] who consider only dyadic emails, we also include emails sent to multiple recipients in our analysis. Figure 5 shows the comparisons of reply rate and reply time for emails with different number of recipients. Most emails are sent to 1 to 8 addresses. Emails sent to 3 to 5 addresses have the highest reply rates (approximately 10%). As the number of email recipients continues to increase, the reply rate begins to decrease. Thus more email recipients does not always mean a higher reply rate. There are at least two reasons for this: 1) some emails sent to many addresses are general announcements or reports which do not require replies; 2) when an email is



(a) Reply rate vs. the number of email recipients.



(b) Reply time vs. the number of email recipients.

Figure 5: The effects of the number of email recipients on reply rate and median reply time. The median reply time denoted by Median RT is in minutes.

sent to many recipients, it is more likely that no one will reply to it since people may think the other co-recipients will reply to it.

Median reply time increases from 44 minutes to 95 minutes as the number of email recipients increases from 1 to 8. Thus emails sent to more recipients get slower replies. When an email has many recipients, people may wait a while to see if others reply before they choose to do so, or it may be that emails that are sent to many people require some work to be accomplished before the reply.

5 PREDICTING REPLY BEHAVIOR

The results presented in Section 4 show that email reply behaviors are influenced by various factors such as time, email content, email addressees, email recipients and attachments. In this section, we use these insights to guide the development of features to train a supervised learning model to predict email reply behavior including reply time and reply action.

5.1 Data Overview and Methodology

Given the Avocado email collection described in Section 3, we split the data into training / testing partitions using the sent time of emails. Specifically, we use emails in 9 months from June 1st 2000 to February 28th 2001 for training and use emails in 3 months from March 1st 2001 to June 1st 2001 for testing. Because email is a temporally ordered collection, we used a split by time (rather than randomly) to ensure that we do not use future information to predict past reply behavior.

We formalize the reply action prediction task as a binary classification problem and the reply time prediction task as a multi-class classification problem. We follow the notations and task definitions presented in Section 3. Given an email message m that user u_i sent to a user set $\{u_j\}$ at time t, reply action prediction is to predict whether any user in $\{u_j\}$ will reply to message m. Thus the classified instances are emails with binary labels. For reply time prediction, we do not necessarily need to get the exact reply time latency. We follow a similar setting as previous related work [20], where we consider three classes of reply times: immediate replies that happen within 25 minutes (32.80% of the training data), quick replies that happen between 25 minutes and 245 minutes (32.84% of the training data), and slow replies that take longer than 245 minutes (34.36% of the training data). The ground truth labels can be directly extracted from the training/testing data based on the actual reply time and actions on emails. The statistics of the experimental data and the distribution of labels for reply time/action prediction task are shown in Table 3 and Table 4.

Table 3: Reply time prediction class distribution.

Class	Training Data		Testing Data	
1: $\leq$ 25min	5,143	32.80%	3,299	38.53%
2: 25-245min	5,150	32.84%	2,820	32.93%
3: $\geq$ 245min	5,389	34.36%	2,444	28.54%

Table 4: Reply action prediction class distribution.

Class	Training Data		Testing Data	
0: No Reply	224,605	93.47%	102,682	92.30%
1: Has Reply	15,682	6.53%	8,563	7.70%

5.2 Learning Models and Evaluation Metrics

We experiment with a variety of machine learning algorithms including Logistic Regression, Neural Networks, RandomForest, AdaBoost [31] and Bagging algorithms. Since the focus of our work is on feature analysis, we only report the experimental results with a basic Logistic Regression (LR) model and the best model, AdaBoost. We used the sklearn[5] package for the implementation of LR and AdaBoost. We did multiple runs of hyper-parameter tuning by grid search to find the best setting for each model with cross validation on the training data.[6]

Since reply action prediction is highly imbalanced (see Table 4) and ranking quality is of importance for tasks like triage, we report Area Under the ROC Curve (AUC), which is a ranking metric insensitive to class imbalance, as the metric for reply action prediction. For reply time prediction, we report precision, recall, F-1 and accuracy (i.e., the percentage of correctly classified emails). The precision, recall, and F-1 are computed as weighted macro averages over all three classes.

5.3 Features

Our analysis in Section 4 identifies factors that impact email reply behaviors. Building on these observations we develop 10 classes of features that can be used to build models for predicting both reply time and reply action. The summary and description of the extracted features is provided in Table 5. In total, we extract 65 features and Bag-of-Words features in 10 feature groups. We normalize all features to be in the range of $[0, 1]$ using min-max feature normalization.

Address Features (Address): These features include features derived from the email addresses such as whether the email is internal or external and the number of recipients.

[5] http://scikit-learn.org

[6] For reply time prediction with LR, we set $C = 1$, the maximum number of iterations as 500, tolerance for stopping criteria as 0.0001, penalization as l1 norm. For reply action prediction with LR, we set $C = 1$, the maximum number of iterations as 100, tolerance for stopping criteria as 0.0001, penalization as l1 norm. For reply time prediction with AdaBoost, we set the learning rate as 0.1, the maximum number of estimators as 800, boosting algorithm as $SAMME.R$. For the reply action prediction with AdaBoost, we set the learning rate as 1, the maximum number of estimators as 50, boosting algorithm as $SAMME.R$.

Table 5: The features extracted for predicting user email reply behaviors. The 10 feature groups Address, BOW, CPred, CProp, HistIndiv, HistPair, Meta, MetaAdded, Temporal, User stand for "Address Features", "Bag-of-Words", "Content Predictions", "Content Properties", "Historical Interaction-Individual", "Historical Interaction-Pairwise", "Metadata Properties", "Metadata Properties-Sender Added", "Temporal Features" and "User Features" respectively. Note that for the computation of features in User, HistIndiv and HistPair, we respect the temporal aspects of the data and only use the historical information before the sent time of the email instance.

Feature	Group	Description
IsInternalExternal	Address	1 binary feature indicating whether the email is internal or external
NumOfRecipients	Address	The number of recipients of the email
BagOfWords	BOW	The bag-of-words features indicating the TF-IDF weights of terms in the email body text
SentimentWords	CPred	2 integer features indicating the number of positive/negative sentiment words in the email body text
CommitmentScore	CPred	The commitment score of the email from an internal binary classifier
RequestScore	CPred	The request score of the email from an internal binary classifier
EmailSubLen	CProp	The length of the subject of the email
EmailBodyLen	CProp	The length of the body of the email
HistReplyRateGlobalUI	HistIndiv	The historical reply rate of the sender u_i towards all the other users
HistReplyNumGlobalUI	HistIndiv	The historical reply count of the sender u_i towards all the other users
HistRecEmailNumSTGlobalUI	HistIndiv	The historical number of received emails of the sender u_i as in sent to address from all the other users
HistRecEmailNumCCGlobalUI	HistIndiv	The historical number of received emails of the sender u_i as in CC address from all the other users
HistSentEmailNumGlobalUI	HistIndiv	The historical number of sent emails of sender u_i to all the other users
HistReplyTimeMeanGlobalUI	HistIndiv	The historical mean reply time of sender u_i to all the other users
HistReplyTimeMedianGlobalUI	HistIndiv	The historical median reply time of sender u_i to all the other users
HistGlobalUJ	HistIndiv	21 features indicating similar mean/min/max historical behavior statistics of recipients $\{u_j\}$ towards the other users
HistReplyNumLocal	HistPair	3 features indicating the historical mean, min, max reply count of the recipient $\{u_j\}$ to sender u_i
HistReplyTimeMeanLocal	HistPair	3 features indicating the historical mean, min, max of the mean reply time of the recipient $\{u_j\}$ towards sender u_i
HistReplyTimeMedianLocal	HistPair	3 features indicating the historical mean, min, max of the median reply time of the recipient $\{u_j\}$ towards sender u_i
HasAttachment	Meta	1 binary feature indicating whether the email has attachments
NumOfAttachment	Meta	1 integer feature indicating the number of attachment of the email
IsImportant	MetaAdded	1 binary feature indicating the importance of the email, which is a tag specified by u_i
IsPriority	MetaAdded	1 binary feature indicating the priority of the email, which is a tag specified by u_i
IsSensitivity	MetaAdded	1 binary feature indicating the sensitivity of the email, which is a tag specified by u_i
TimeOfDay	Temporal	4 binary features indicating the time of the day (0-6, 6-12, 12-18, 18-24)
DayOfWeek	Temporal	7 binary features indicating the day of week (Sun, Mon, ... , Sat)
WeekDayEnd	Temporal	2 binary feature indicating whether the day is a weekday or a weekend
UserDepartment	User	1 feature indicating the department of the email sender u_i.
UserJobTitle	User	1 feature indicating the job title of the email sender u_i.

Bag-of-Words (BOW): These features include the TF-IDF weights of non-stop terms in the email body text. The vocabulary size of our experimental data set is 554061.

Content Predictions (CPred): These features include some predictions like positive / negative sentiment words, commitment / request scores from email textual content. We count the number of positive sentiment words and negative sentiment words in email body text using a sentiment lexicon from a previous research [17]. We also include the commitment and request score of emails from an internal classier to infer the likelihood of whether an email contains a commitment or a request.

Content Properties (CProp): These features are content properties including the length of email subjects and email body text.

Historical Interaction-Individual (HistIndiv): These features characterize the historical email interactions related to each sender u_i and recipient in $\{u_j\}$ aggregated across all interactions. This feature group has two subgroups: global interaction features for the sender u_i and global interaction features for recipients $\{u_j\}$. The global interaction features for the sender u_i contain a set of features to capture the historical interactions between u_i with all the other users such as reply rate, reply count, number of received emails, number of sent emails, mean/median reply time, etc. For the global interaction features for recipients $\{u_j\}$, since there could be multiple recipients, we compute the mean/min/max of those statistics to capture the historical interactions between $\{u_j\}$ and all other users.

Historical Interaction-Pairwise (HistPair): These features characterize the local (pairwise) interactions between the sender u_i and the recipients $\{u_j\}$, which are statistics like number of replied emails, mean/median reply time from the historical interactions between the sender u_i and the recipients $\{u_j\}$. Note that for the computation of "HistIndiv" and "HistPair" features, we compute per day updated user profiles and only use the information before the email instance.

Metadata Properties (Meta): This feature group contains features derived from email attachments including whether the email has attachments and number of email attachments.

Metadata Properties-Sender Added (MetaAdded): These features include tags specified by the sender u_i to indicate the importance, priority or sensitivity of the sent email. In our data, less than 3% of emails have such tags. But they can still provide some clues to infer user reply behavior once they are set by the sender.

Temporal Features (Temporal): These features are generated based on the sent time of emails to capture the temporal factors on user email reply behaviors.

User Features (User): These features include the department and job title of the person.

Table 6: Summary of the prediction results for user email reply time and reply action. The Precision, Recall and F-1 scores are weighted averages by supports over all classes. The best performance is highlighted in boldface. Both LR and AdaBoost show significant improvements over all baseline methods with p < 0.01 measured by micro sign test [30].

	Action	Time			
Method	AUC	Prec	Rec	F1	Accuracy
Random	.5024	.3262	.3253	.3244	.3257
Majority Vote	.5000	.0815	.2854	.1267	.2854
Previous Reply	.5858	.3717	.3742	.3613	.3633
LR	.7036	.3952	.4098	.3791	.4098
AdaBoost	**.7208**	**.4561**	**.4591**	**.4476**	**.4591**

5.4 Baselines

We compare our method against three baselines as follows:

Random. This method randomly generates a predicted class from the label distribution in the training data.

Majority Vote. This method always predicts the largest class, which is class 0 (no reply) for reply action prediction and class 3 (> 245min) for reply time prediction.

Previous Reply. This method generates predictions according to the previous reply behaviors of the recipients $\{u_j\}$ towards the sender u_i before the sent time t of email m. For reply action, it predicts 0 (no reply) if there is no previous reply behavior from $\{u_j\}$ to u_i. For reply time, it predicts the majority class if there is no previous reply behavior from$\{u_j\}$ to u_i. If there are previous reply behaviors, it computes the median time of previous reply time as the predicted result. Note that this baseline is similar to the "Last Reply" baseline used in [20]. [7]

5.5 Experimental Results and Analysis

We now analyze the results of our proposed method compared with various baseline methods. Table 6 shows the prediction results for reply action and reply time. Figure 6 shows the ROC curves for all methods for reply action prediction. The baseline "Majority Vote", while accurate since 92.30% of emails are negative, achieves zero true positive rate (recall) and predicts no positive instances. Likewise "Random" falls nearly on the $x = y$ dashed line in red which indicates expected random performance (empirical variance leads to a slight bit of luck). As shown in Table 6, both LR and AdaBoost outperform all three baselines by AUC. The best model AdaBoost achieves large improvements of 44.16% comparing with "Majority Vote" and 23.05% comparing with "Previous Reply". AdaBoost achieves slightly better performance than LR. Examining the ROC curves, the most competitive baseline is "Previous Reply", which is still under the ROC curves of LR and AdaBoost. The AUC scores show that our methods outperform all baseline methods with a large margin.

For reply time prediction, both LR and AdaBoost models with the proposed features outperform all baseline methods with large gains. The differences are statistically significantly with $p < 0.01$ measured by a micro sign test [30]. The best method based on AdaBoost achieves large improvements of 23.89%, 26.36% for F-1 and accuracy comparing with "Previous Reply" and 253.18%, 60.85% for F-1, accuracy comparing with "Majority Vote". Comparing the two learning models, AdaBoost has better performance than LR in terms of both F-1 and accuracy. This shows the advantages of AdaBoost that can feed the relative "hardness" of each training sample into the tree growing algorithm such that later trees tend to focus on harder-to-classify instances.

5.6 Feature Importance Analysis

Feature Group Analysis. We further perform analyses to understand the relative importance of different feature groups in predicting reply time. We consider two approaches: (1) Remove one feature group. We observe the change in performance when we remove any one of the 10 feature groups. (2) Only one feature group. We observe the change in performance when we classify emails only using one feature group. Table 7 and Table 8 show the results of these analyses for reply action prediction and reply time prediction using the AdaBoost model, which is the best method in our previous experiments.

Table 7 shows the performance when we use only one feature group. The classes are ordered by AUC scores on action prediction. The most important features are highlighted using a triangle. For reply action prediction, "HistIndiv" and "HistPair" show the best performance compared to other feature groups. Using only "HistIndiv" features results in 0.6924 for the AUC score, which is close to the performance with all feature groups. These results suggest that historical interactions are important features for reply action prediction. "CPred" features (i.e., algorithmic predictions of request, commitments and sentiment) are also important although somewhat less so than the historical interaction features. However, for reply time prediction we see a different story. "Temporal" features are the most important features for predicting reply time as highlighted in Table 7. Using only "Temporal" features results in good latency prediction accuracy of 0.4261, which is only slightly worse than the result from combining all feature groups. "HistIndiv" features which result in accuracy above 0.40 are also helpful in predicting latency. For reply actions, historical interaction features are the most important in indicating whether people will reply to the email eventually no matter the time latency. Given they will reply to an email, people seem to strongly prefer to reply during office hours on workdays, which explains why "Temporal" factors are so important for reply time prediction.

Another way of looking at the importance of features is to remove one class and look at the decrease in performance. Table 8) shows the results of removing one feature group. Performance decrease the most when we remove "HistIndiv" features for reply action prediction and "Temporal" features for reply time prediction. These results are consistent with the results when we only use one feature group. We also found some features are not very useful for reply behaviour prediction. For instance, when we remove "Meta" features which are derived from email attachments, both F-1 and accuracy increase slightly for reply time prediction. This suggests that there is still space for feature selection to further improve the

[7] For the proposed method in [20], we can not reproduce their method since they don't disclose the details of the 83 features in their model and they also don't release the code and data due the proprietary nature of their work.

Table 7: Comparison of performance on predicting reply time and reply action when we only use one feature group. The learning model used is AdaBoost. The best performance is highlighted in boldface. ▲ indicates strong performance when only use one feature group. The feature settings are sorted by the AUC scores.

	Action	Time			
Feature Set	AUC	Prec	Rec	F-1	Accuracy
HistIndiv	.6924▲	.3891	.4104▲	.3642	.4104▲
HistPair	.6382▲	.3721	.3890	.3463	.3890
CPred	.5954	.3748	.3784	.3352	.3784
User	.5944	.3729	.3847	.3575	.3847
Address	.5912	.3790	.3641	.3038	.3641
Temporal	.5401	.4436▲	.4261▲	.4264▲	.4261▲
CProp	.5346	.3415	.3785	.3060	.3785
MetaAdded	.5291	.2524	.3877	.2672	.3877
Meta	.5247	.2398	.3670	.2866	.3670
BOW	.5106	.3744	.3976	.3391	.3976
AllFeat	**.7208**	**.4561**	**.4591**	**.4476**	**.4591**

Table 8: Comparison of performance on predicting reply time and reply action when we remove one feature group. The learning model used is AdaBoost. ▼ indicates large drops in performance when remove one feature group. The feature settings are sorted by the AUC scores.

	Action	Time			
Feature Set	AUC	Prec	Rec	F-1	Accuracy
−HistIndiv	.6620▼	.4453	.4481	.4409▼	.4481
−CProp	.7112	.4472	.4543	.4413	.4543
−Address	.7187	.4550	.4599	.4476	.4599
−MetaAdded	.7191	.4549	.4579	.4449	.4579
−HistPair	.7198	.4544	.4572	.4432	.4572
−Meta	.7216	.4573	.4604	.4515	.4604
−Temporal	.7218	.3841▼	.4056▼	.3800▼	.4056▼
−CPred	.7229	.4540	.4611	.4457	.4611
−BOW	.7237	.4539	.4573	.4503	.4573
−User	.7256	.4473	.4482	.4431	.4482
AllFeat	.7208	.4561	.4591	.4476	.4591

performance of reply behavior prediction. We leave the study of feature selection to future work.

Importance of Individual Features. AdaBoost [31] provides a mechanism for reporting the relative importance of each feature. By analyzing the relative importance, we gain insights into the importance of individual features for different email reply prediction tasks. Table 9 shows the most important features for predicting reply time and reply action with the relative feature importance learned by AdaBoost. The most important features for reply action prediction are historical interaction features including "HistReplyCountRecipientMax", "HistSentEmailCountSender", "HistReceiveEmailSTRecipientMin", content properties like the length of email subjects and email bodies, and address features like "NumOfReceivers", "IsInternalExternal" etc. On the other hand, temporal features like "TimeOfDay1Morning", "IsWeekDay", "DayOfWeek1Sunday", "TimeOfDay4Night" are among the most important features for reply time prediction. These interesting differences are also consistent with the results in the feature group analysis. Some features including historical interaction features and content properties like the length of email bodies are important for both reply action prediction and reply time prediction.

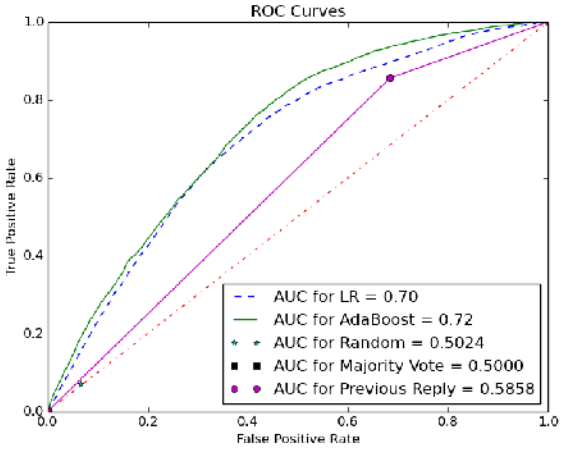


Figure 6: The ROC curves of different methods for the reply action prediction task.

6 CONCLUSIONS AND FUTURE WORK

In this paper, we introduce and formalize the task of reply behavior prediction in a corporate email setting, using the publicly available Avocado collection. We characterize various factors affecting email reply behavior, showing that temporal features (time of day and day of week), content properties (such as the length of email subjects and email bodies) and prior interactions between the sender and recipients are related to reply behavior. We use these insights to extract 10 classes of features groups and build models to predict whether an email will be responded to and how long it will take to do so. We show that the proposed methods outperform all baselines with large gains. We further show that temporal, textual content properties, and historical interaction features are especially important in predicting reply behavior.

Our research represents an initial effort to understand email actions in a corporate setting. We examined email reply behavior in detail in one technology company, but is unclear how representative the company is. It is important to see how our findings generalize to different industry sectors and different demographic backgrounds of employees. Future work will consider more available email collections and more features that could be signals for user reply behavior prediction. Of special interest is the use of richer content features such as lexical, syntactic and semantic features. We shared the IDs of the emails that we consider in our research so that others can extend our research on the Avocado collection.

Table 9: The most important features for predicting reply time and reply action with relative feature importances in AdaBoost.

	Reply Action Prediction			Reply Time Prediction		
Rank	Feature Name	Group	Importance	Feature Name	Group	Importance
1	EmailSubjectLen	Cprop	1.0000	TimeOfDay1Morning	Temporal	1.0000
2	HistReplyCountRecipientMax	HistIndiv	0.5714	IsWeekDay	Temporal	0.8083
3	HistSentEmailCountSender	HistIndiv	0.4286	DayOfWeek1Sunday	Temporal	0.4946
4	HistReceiveEmailSTRecipientMin	HistIndiv	0.4286	EmailBodyLen	Cprop	0.3260
5	NumOfReceivers	Address	0.4286	TimeOfDay4Night	Temporal	0.3052
6	EmailBodyLen	Cprop	0.4286	HistRTMedianRecipientAvg	HistIndiv	0.1690
7	HistLocalMeanRTMin	HistPair	0.2857	HistRTMedianRecipientMin	HistIndiv	0.1563
8	IsInternalExternal	Address	0.2857	HistLocalReplyCountMax	HistPair	0.1125
9	UserJobTitleSender	User	0.2857	HistReceiveEmailSTSender	HistIndiv	0.1101
10	HistLocalReplyCountMin	HistPair	0.2857	IsPriority	MetaAdded	0.0951
11	NumoOfAttachment	Meta	0.2857	IsWeekEnd	Temporal	0.0909
12	HistReceiveEmailCCSender	HistIndiv	0.1429	HistLocalMedianRTMin	HistPair	0.0907
13	HistRTMeanSender	HistIndiv	0.1429	HistReceiveEmailCCSender	HistIndiv	0.0858
14	HistReplyCountRecipientAvg	HistIndiv	0.1429	IsSensitivity	MetaAdded	0.0759
15	HistLocalReplyCountMax	HistPair	0.1429	HistRTMeanRecipientAvg	HistIndiv	0.0674

7 ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #IIS-1419693. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] Douglas Aberdeen, Ondrej Pacovsky, and Andrew Slater. The Learning behind Gmail Priority Inbox. In *NIPS 2010 Workshop on Learning on Cores, Clusters and Clouds*.

[2] Victoria Bellotti, Nicolas Ducheneaut, Mark Howard, and Ian Smith. Taking Email to Task: The Design and Evaluation of a Task Management Centered Email Tool. In *CHI '03*.

[3] Paul N. Bennett and Jaime Carbonell. Detecting Action-items in E-mail. In *SIGIR '05*.

[4] Vitor R. Carvalho and William W. Cohen. On the Collective Classification of Email "Speech Acts". In *SIGIR '05*.

[5] Marta E. Cecchinato, Abigail Sellen, Milad Shokouhi, and Gavin Smyth. Finding Email in a Multi-Account, Multi-Device World. In *CHI'16*.

[6] Michael Chui, James Manyika, Jacques Bughin, Richard Dobbs, Charles Roxburgh, Hugo Sarrazin, Geoffrey Sands, and Magdalena Westergren. 2012. The social economy: Unlocking value and productivity through social technologies. (2012). A report by McKinsey Global Institute.

[7] William W. Cohen, Vitor R. Carvalho, and Tom M. Mitchell. Learning to Classify Email into Speech Acts. In *EMNLP '04*.

[8] Simon Corston-Oliver, Eric Ringger, Michael Gamon, and Richard Campbell. Integration of Email and Task Lists. In *First Conference on Email and Anti-Spam*.

[9] Laura A. Dabbish and Robert E. Kraut. Email Overload at Work: An Analysis of Factors Associated with Email Strain. In *CSCW '06*.

[10] Laura A. Dabbish, Robert E. Kraut, Susan Fussell, and Sara Kiesler. Understanding Email Use: Predicting Action on a Message. In *CHI '05*.

[11] Dotan Di Castro, Zohar Karnin, Liane Lewin-Eytan, and Yoelle Maarek. You've Got Mail, and Here is What You Could Do With It!: Analyzing and Predicting Actions on Email Messages. In *WSDM '16*.

[12] Danyel Fisher, A. J. Brush, Eric Gleave, and Marc A. Smith. Revisiting Whittaker & Sidner's "Email Overload" Ten Years Later. In *CSCW '06*.

[13] Michael Freed, Jaime G. Carbonell, Geoffrey J. Gordon, Jordan Hayes, Brad A. Myers, Daniel P. Siewiorek, Stephen F. Smith, Aaron Steinfeld, and Anthony Tomasic. RADAR: A Personal Assistant that Learns to Reduce Email Overload. In *AAAI '08*.

[14] David Graus, David van Dijk, Manos Tsagkias, Wouter Weerkamp, and Maarten de Rijke. Recipient Recommendation in Enterprises Using Communication Graphs and Email Content. In *SIGIR '14*.

[15] Mihajlo Grbovic, Guy Halawi, Zohar Karnin, and Yoelle Maarek. How Many Folders Do You Really Need?: Classifying Email into a Handful of Categories. In *CIKM '14*.

[16] Ido Guy, Michal Jacovi, Noga Meshulam, Inbal Ronen, and Elad Shahar. Public vs. Private: Comparing Public Social Network Information with Email. In *CSCW '08*.

[17] Minqing Hu and Bing Liu. Mining and Summarizing Customer Reviews. In *KDD '04*.

[18] Anjuli Kannan, Karol Kurach, Sujith Ravi, Tobias Kaufmann, Andrew Tomkins, Balint Miklos, Greg Corrado, Laszlo Lukacs, Marina Ganea, Peter Young, and Vivek Ramavajjala. Smart Reply: Automated Response Suggestion for Email. In *KDD '16*.

[19] Bryan Klimt and Yiming Yang. The Enron Corpus: A New Dataset for Email Classification Research. In *ECML'04*.

[20] Farshad Kooti, Luca Maria Aiello, Mihajlo Grbovic, Kristina Lerman, and Amin Mantrach. Evolution of Conversations in the Age of Email Overload. In *WWW '15*.

[21] Andrew Lampert, Robert Dale, and Cecile Paris. Detecting Emails Containing Requests for Action. In *HLT '10*.

[22] Carman Neustaedter, A. J. Bernheim Brush, and Marc A. Smith. Beyond "From" and "Received": Exploring the Dynamics of Email Triage. In *CHI EA '05*.

[23] Byung-Won On, Ee-Peng Lim, Jing Jiang, Amruta Purandare, and Loo-Nin Teow. Mining Interaction Behaviors for Email Reply Order Prediction. In *ASONAM '10*.

[24] Ashequl Qadir, Michael Gamon, Patrick Pantel, and Ahmed Hassan Awadallah. Activity Modeling in Email. In *NAACL-HLT '16*.

[25] S. Radicati. 2014. Email statistics report, 2014-2018. (2014).

[26] Maya Sappelli, Gabriella Pasi, Suzan Verberne, Maaike de Boer, and Wessel Kraaij. 2016. Assessing E-mail intent and tasks in E-mail messages. *Inf. Sci.* 358-359 (2016), 1–17.

[27] Michael Gamon Richard Campbell Simon H. Corston-Oliver, Eric Ringger. Task-focused Summarization of Email. In *ACL'04*.

[28] Joshua R. Tyler and John C. Tang. When Can I Expect an Email Response? A Study of Rhythms in Email Usage. In *ECSCW'03*.

[29] Steve Whittaker and Candace Sidner. Email Overload: Exploring Personal Information Management of Email. In *CHI '96*.

[30] Yiming Yang and Xin Liu. A Re-examination of Text Categorization Methods. In *SIGIR '99*.

[31] Ji Zhu, Hui Zou, Saharon Rosset, and Trevor Hastie. 2009. Multi-class AdaBoost. *Statistics and Its Interface* (2009).