Mining Social Networks for Personalized Email Prioritization

Shinjae Yoo[§], Yiming Yang[§], Frank Lin[§], Il-Chul Moon[¶]

Language Technology Institute[§]
Carnegie Mellon University
5000 Forbes Ave
Pittsbrugh, PA 15213
USA
{sjyoo,yiming,frank}@cs.cmu.edu

Division of Electrical Engineering[¶]
KAIST
335 Gwahangno
Yuseong-gu, Daejeon 305-701
Republic of Korea
icmoon@smslab.kaist.ac.kr

ABSTRACT

Email is one of the most prevalent communication tools today, and solving the email overload problem is pressingly urgent. A good way to alleviate email overload is to automatically prioritize received messages according to the priorities of each user. However, research on statistical learning methods for fully personalized email prioritization (PEP) has been sparse due to privacy issues, since people are reluctant to share personal messages and importance judgments with the research community. It is therefore important to develop and evaluate PEP methods under the assumption that only limited training examples can be available, and that the system can only have the personal email data of each user during the training and testing of the model for that user. This paper presents the first study (to the best of our knowledge) under such an assumption. Specifically, we focus on analysis of personal social networks to capture user groups and to obtain rich features that represent the social roles from the viewpoint of a particular user. We also developed a novel semi-supervised (transductive) learning algorithm that propagates importance labels from training examples to test examples through message and user nodes in a personal email network. These methods together enable us to obtain an enriched vector representation of each new email message, which consists of both standard features of an email message (such as words in the title or body, sender and receiver IDs, etc.) and the induced social features from the sender and receivers of the message. Using the enriched vector representation as the input in SVM classifiers to predict the importance level for each test message, we obtained significant performance improvement over the baseline system (without induced social features) in our experiments on a multi-user data collection. We obtained significant performance improvement over the baseline system (without induced social features) in our experiments on a multi-user data collection: the relative error reduction in MAE was 31% in micro-averaging, and 14% in macro-averaging.

KDD'09, June 28– July 1, 2009, Paris, France.

Categories and Subject Descriptors

I.7.m [**Computing Methodologies**]: Document and Text Processing—*Miscellaneous*; I.5.3 [**Computing Methodologies**]: Pattern Recognition—*Clustering*; I.5.4 [**Computing Methodologies**]: Pattern Recognition—*Applications*

General Terms

Algorithms, Experimentation, Security, Human Factors, and Languages.

Keywords

Email Prioritization, Social Network, and Text Mining.

1. INTRODUCTION

Email is one of the most prevalent personal and business communication tools today; however, it is not without significant drawbacks. In contrast to telephone conversations or face-to-face meetings, communication through email is asynchronous in the sense that we receive messages (after some spam filtering) in the same way regardless of our level of interest, and a single sender can flood multiple receivers (unlike telephone or instant messaging). Users are left with the burden of having to process a large volume of email messages of differing importance. This tedious task has been shown to cause significant negative effects on both personal and organization performance [6] [20]. There is an urgent need to solve this information overload problem; i.e., we need to develop systems that automatically learn personal priorities for each user, and that identify personally interesting and important messages for user's attention.

Many statistical learning techniques have been studied in support of email-based prediction tasks, including supervised, unsupervised and semi-supervised methods for spam identification [21][22], folder recommendation [23], recipient reminding [24], action-item identification [25], social group analysis [26], etc. In spite of the wide variety of efforts and significant accomplishments, personalized email prioritization (PEP) remains an under-explored problem. Thorough investigations and conclusive solutions have been rare, mainly due to privacy issues in collecting personal data for training and testing. Unlike spam filtering where people are less concerned with sharing individually labeled spam messages, PEP requires personal judgments of the importance levels of non-spam email messages. Few are willing to share this data due to privacy

concerns. Companies who have access to customers' email messages (like Google, Yahoo! and Microsoft) cannot share such data with academic institutes for the same reason. Personal importance judgments are also missing from the Enron corpus, which has been used as a benchmark dataset in email research and evaluations. A message important for an Enron employee might not be equally important for a high-level manager. In short, there is no publicly available dataset that contains personal importance judgments by real users and on personal messages, leaving researchers no choice but to go through a process of collecting private data under strict IRB (Institutional Review Board) guidelines. Such data collection processes are costly, time consuming, tedious, and difficult to scale to a large number of users with diverse criteria in judging the importance of email messages. As a result, PEP remains an area which has not been well studied thus far.

This paper presents the first study with several statistical classification and clustering methods (including our new approach) addressing the PEP problem based on personal importance judgments by multiple users. We constructed a new dataset of anonymized email messages from each user, and used parts of the data to train personalized models and other parts to test the effectiveness of those models. Our primary research question is: "How can we effectively learn user-specific models for accurate prediction of personalized importance using only small amounts of labeled training data and limited observations on personal communications with others?" Specifically, our contributions in this paper include:

1) We created a new collection of anonymized personal email data with fine-grained importance levels. Previous work used datasets with only two priority levels, i.e., spam vs. non-spam [14], which are not sufficient for discriminating personal importance levels on non-spam email messages. On the other hand, past research with human subjects indicates that users would have difficulties in producing consistent labels if too many levels were required [13][27]. Hence, we took a middle ground with 5 levels. To our knowledge, this is the first multi-user email prioritization dataset with fine-grained importance labels.

2) We proposed a fully personalized methodology for technical development and evaluation. By fully personalized we mean that only the personal email data (textual or social network information) of each user is available for the system during the training and testing of the user-specific model. This is an important assumption for the generality of PEP methods, i.e., we cannot rely on the availability of centralized access to customer private data, neither in the development circle nor in the evaluation phase, and we cannot take the liberty to use a particular user's private data to build models for other users because the potential leak of private information across users. This assumption makes our work in this paper fundamentally different from those in spam filtering and other previous work on email-based prediction tasks.

3) We developed a supervised classification framework for modeling personal priorities over email messages, and for predicting importance levels for new messages. Using standard Support Vector Machines (SVMs) as the classifiers, the novel part of our approach is the enriched representation of each input email message, especially in the part that represent the contact persons (sender or recipients in the CC list) in the message. We explore three different types of enriched features that are automatically induced based on personal social networks as follows:

   - **Clustering contact persons based on personal social networks** We want to capture social groups among senders and recipients, which can be learned from personal email messages without importance labels (unsupervised learning). For example, email messages from two different senders who are members of the same team may carry similar importance. A personal social network is constructed for each user using his or her own data (Section 2.2). Finding closely-associated user groups from the personal perspective enables us to estimate the expected importance level per group, as a strategy for improving the robustness of importance prediction when training data are relative sparse.

   - **Measuring social importance of contacts** We want to capture leadership levels of individual contacts, and we define eight centrality measures that can be automatically computed using the graph structure of each personalized social network. Most of those metrics have been commonly used in Social Network Analyses (SNA) research for spam filtering; however, their use in personalized email prioritization has not been studied in depth. As personal social networks are different from user to user, using multi-dimensional leadership metrics to jointly characterize different users would lead to more robust predictions than using any single metric alone.

   - **Simi-supervised importance propagation** When importance labels are available for some email messages (e.g. older messages) but not available for other messages (e.g. newer ones), we can use the personal social network of each user to propagate the importance scores from messages to contacts, then from contacts to messages, and repeat the propagation until all the scores are stabilized. By doing so, we make another use of personal social networks, i.e., leveraging the transitivity of importance scores through personal social connections.

4) We present an empirical evaluation of the proposed approach in comparison with the baseline classifiers (SVMs) that do not use social-network induced features to represent senders. SVMs with the enriched sender representation obtained a significant error reduction (31% in micro-averaged MAE and 14% in macro-average MAE) over the results of the baseline method. Our experiments also show that for different users we need to rely on very different social network features for accurate PEP predictions and that our system can automatically discover and utilize those features.

2. SOCIAL CLUSTERING

2.1 Motivation

In predicting the importance of email messages, the sender information is one of the most indicative features. For example, we may have multiple user groups such as project teams or social

activity groups, and email reflects membership in such social groups naturally through co-recipient list. Often the messages sent by the members of the same group tend to share similar priority levels; thus, capturing sender groups would be informative for predicting the importance of messages.

Since we have a limited amount of training data, it is very likely that in the test data we encounter a sender who does not have any labeled instances in the training set. However, if we can identify this user as a member of a group based on unsupervised clustering, then we can infer that user's importance from that of other group members. In other words, the clustering produces equivalent classes of users based on their communication patterns in a personal social network. These clusters are used later by SVMs as input features (in addition to a standard bag-of-word representation) to each message (Sections 2.3 and 5.3). As a result, senders without labeled messages can also receive non-zero weight through these clusters, effectively addressing the data sparse problem in PEP.

2.2 Personalized Social Network

We construct a *personalized* social network for each particular user using only the email data of that user. There are two reasons for this: **Practicality**—we want our method to not rely on the unrealistic assumption that multi-user private data are always available for system development and model optimization. **Personalization**—we want the social network best representing the user's own social activity; a global social network may include noisy features and de-emphasize personalization in the inductive learning of important features through the network.

Let us use a graph $G=(V,E)$ to represent the email contact network where vertices V correspond to the email contacts (users) in the network, and edges E correspond to the messages sending events among users. The edges are un-weighted, i.e., $E_{ij}=1$ if there is (at least) a message from user i to user j, and $E_{ij}=0$ otherwise.

2.3 Newman Clustering

We choose the Newman clustering algorithm, which has been reported to successfully find social structures in large organizations [17][18]. It defines the *edge-betweenness* as a normalized number of shortest paths going through a specific link from all-pairs shortest paths. If a link has a high edge-betweenness score, it means that the link is crucial between two boundary nodes of two different highly-connected clusters. The algorithm assumes that members in a highly-connected cluster have many communication passages within the cluster, but not many links outside the cluster. Based on this assumption, it deletes links with high edge-betweenness scores, which results in disconnect components as clusters.

One way to control the granularity level of clusters is to pre-specify the number of desired clusters. This number may be based on domain knowledge about the network or automatically determined by an algorithm which certain optimization criterion or a heuristic measure. The Organization Risk Analyzer (ORA) [5] picks the number that yields the largest decrease in the sum of edge-betweenness per cluster. We use ORA in this work. Figure 1 shows embedded clusters in a network where ORA selects 27 as the number of clusters.

3. MEASURING SOCIAL IMPORTANCE

We want to measure the social importance levels of contacts, and this can be done without labeled training data. Instead, the personal contact network induced from senders and recipients link relations provides useful information about the centrality of each contact in the network. For instance, the Newman Cluster #1 in Figure 1 is highly connected with others and the person in the center of the cluster may be an important person in the network. We examine multiple graph-based metrics to characterize the social centrality of each node, which have been commonly used in social network analysis (SNA) or link structure analysis.

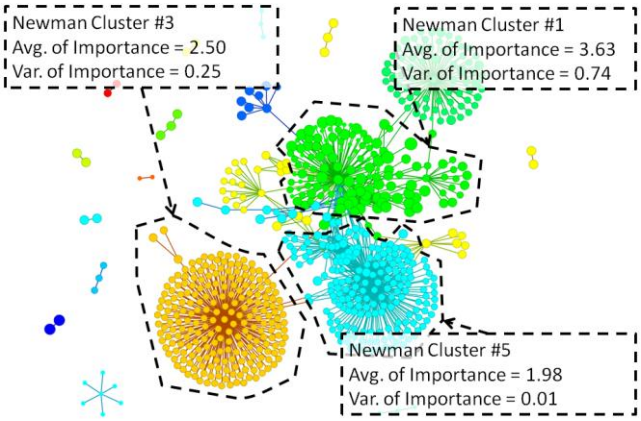


Figure 1 The clusters produced using the Newman clustering algorithm based on the email contact network of a user: nodes are the senders, and node sizes are adjusted to reflect the average importance of members in each cluster.

3.1 In-degree centrality

For node i, we define *InDegreeCent*(i) as the normalized number of unique senders who sent email to contact i:

$$InDegreeCent(i) = \frac{1}{|V|}\sum_{j=1}^{i} E_{ji}$$

where $E_{ji} \in \{0,1\}$, and $|V|$ is the total number of contacts in the personal email social network. A high in-degree may indicate that the recipient is a popular person.

3.2 Out-degree centrality

OutDegreeCent(i) is defined as the normalized number of people who receive email from contact i. Having a high out-degree may also mean certain kind of importance, e.g., as an announcement sender or a mailing-list organizer.

$$OutDegreeCent(i) = \frac{1}{|V|}\sum_{j=1}^{i} E_{ij}$$

3.3 Total-degree centrality

TotalDegreeCent(i) is defined as the normalized number of unique senders and recipients who had email communication with node i. That is, it is the simple average of the in-degree and out-degree of the node:

$$TotalDegreeCent(i) = \frac{1}{|V|}\sum_{j=1}^{|V|}\left\lceil \frac{E_{ij}+E_{ji}}{2} \right\rceil$$

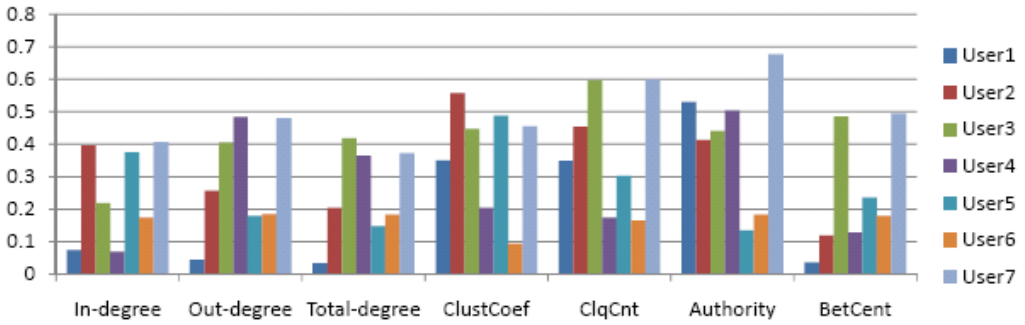


Figure 2: The Pearson Correlation Scores (vertical axis) of social metrics (horizontal axis) for different users

3.4 Clustering Coefficient

Clustering Coefficient of node *v*, denoted as *ClustCoef(v)*, measures the connectivity among the neighborhood of the node.

$$ClustCoef\ (v) = \frac{1}{Z} \sum_{i \in Nbr(v)} \sum_{j \in Nbr(v), j \neq i} E_{ij}$$

3.5 Clique Count

The clique count of a node *v* is also a neighborhood metric where the clique is a fully connected sub-graph. The clique count of a node *v*, *ClqCnt(v)*, is the number of clique sub-graphs which contain the node *v*. A large clique count means that the node *v* is connected to large and well-connected sub-graph and node *v* is located in the center of the sub-graph. Although it is not a global social metric, it measures wider network centrality than degree-based centralities or clustering coefficients.

3.6 Betweenness centrality

Betweenness centrality of a node *v*, *BetCent(v)*, is the percentage of existing shortest paths out of all possible paths that goes through the node *v*. A node with high betweenness centrality means that the corresponding person is a contact point between different social groups.

$$BetCent(v) = \frac{1}{(n-1)(n-2)} \sum_{j=1, j \neq i}^{|V|} \sum_{k=1, k \neq j, k \neq i}^{|V|} \frac{\sigma_{jk}(i)}{\sigma_{jk}}$$

where σ_{jk} is the number of shortest path between *j* and *k* and $\sigma_{jk}(i)$ is the number of shortest path between *j* and *k* that goes through *i*. This metric has been used in social network analysis [17].

3.7 HITS Authority

$HITSAuth(i)$ measures the global importance of the node *i*. The difference between $HITSAuth(i)$ and degree-based centrality is that HITS is recursively defined, taking the transitivity of popularity into account. $HITSAuth(i)$ is defined as follows: Let us use an *N*-by-*N* matrix to define the adjacency matrix whose elements are defined as $X_{ij} = E_{ij} \in \{0,1\}$ where $X_{ij} = 1$ if and only if there is a link from *i* to *j*, i.e., if and only if there is at least one message sent from person *i* to *j*. The $HITSAuth(i)$ can be calculated by finding the principle eigenvector *r* of matrix XX^T where *r* satisfies the equation $XX^T r = \lambda r$ and λ is the largest eigenvalue of XX^T.

where $Nbr(v) = \{x : E_{vx} \neq 0, E_{xv} \neq 0\}$ is the neighborhood and $Z = | Nbr(v) | \cdot (Nbr(v) - 1)$ is the normalization denominator. Boykin and Roychowdhury [2] used this metric to discriminate spam from non-spam email messages based on the neighborhood connectivity of the recipients of messages.

3.8 PPC Analysis

We computed the PCC (Pearson Correlation Coefficient) values of each social metric with respect to the human-labeled importance levels of email messages in our dataset. The PCC values are indicative about how useful each social metric feature would be for predicting the importance of messages based on the metric alone (i.e., not counting the interactions among the metrics). Figure 2 shows the absolute values of the correlation coefficient scores. The multi-metric PCC values differ from user to user, which is not surprising. For user 1, as an example, Clustering Coefficient, Clique Count and HITS Authority scores are highly informative, but In-degree, Out-degree and Total-degree are not. But for User 5, HITS Authority score is not a good predictor but in-degree is highly informative. Using multiple metrics we improve the robustness of the predictions.

4. SIMI-SUPERVISED IMPORTANCE PROPAGATION (SIP)

So far we have focused on unsupervised feature induction to enrich the representation of email contact persons. Now let us focus on another way to leverage personal email social networks, i.e., to propagate the importance values of labeled email messages (the training examples) to other messages and corresponding contact persons. We propose a new solution for Semi-supervised Importance Propagation (SIP) as the following.

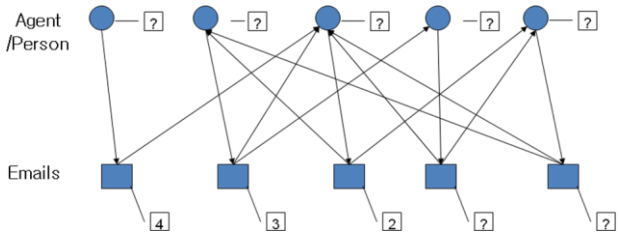


Figure 3: An example of bipartite email network: circles are contact persons and rectangles are email messages. Some email messages have human-assigned importance values but others do not. The network enables us to propagate the partially available importance values from messages to persons, and vice versa.

4.1 SIP Algorithm

As shown in Figure 3, we use a bipartite graph to represent the interactions between email contacts (circles) and email messages (boxes); we call this graph a personal email network. Let *N* be the number of email contacts and *M* be the number of messages, the two types of edges in the graph can be represented using matrix *A* (*N* by *M*) and matrix *B* (*N* by *M*), respectively where $A_{i,j} = 1$ if person *i* sends message *j*, and $A_{i,j} = 0$ otherwise; $B_{i,j} = 1$ if person *i* received message *j*, and $B_{i,j} = 0$ otherwise.

The bipartite network allows us in "inject" human-assigned importance values which are available for some messages in the graph, and propagate them through the links among messages and contact persons. To be specific, let us treat each importance label (among 1, 2, 3, 4 and 5) as a "category", and use vector $\vec{x}_k$ (*M*-by-1) to indicate the labels of messages with respect to category *k* as: $x_{k,i} = 1$ if message *i* belongs to category *k*, and $x_{k,i} = 0$ otherwise. The importance propagation from messages to persons (receivers) is calculated as $\vec{y}_k = B\vec{x}_k$, and the importance propagation from persons (senders) to messages is calculated as $\vec{x}_k A\vec{y}_k$. Updating of the importance values for contact persons at each time step (t) is calculated by:

$$\vec{y}_k^{t+1} = BA^T\vec{y}_k^t = \left(BA^T\right)^t B\vec{x}_k$$

Through this formula, we see that vector $\vec{y}_k^{t+1}$ is a linear transformation of the starting vector $\vec{x}_k$ whose elements are the partially available importance values (at a specific level) of messages, and the transformation is uniquely defined by $\left(BA^T\right)^t$ which is induced from the personal email network. It is well understood in link analysis that if matrix $C = BA^T$ is irreducible and if *t* is sufficiently large, then $\vec{y}_k$ stabilizes at the principal eigenvector of *C*. However, as *C* is induced from an arbitrary email collection, the irreducible property of the matrix is not guaranteed. Even if *C* happens to be irreducible, its principal eigenvector is still insensitive to the starting vector $\vec{x}_k$, and hence is not what we want. We first put them in probability framework and to address both issues, we make a linear interpolation:

1) We define $\vec{y}_k^1 = B\vec{x}_k$, and we normalize each of its elements using the sum of the total elements in the vector. Let us denote the normalized vector as $\vec{y}_k^1$, which contains the initial importance values of all persons in the network. Clearly the elements of the vector sum to one. We then define an importance-sensitive matrix $U_k = \vec{y}_k^1\vec{1}$, whose columns are identical and each column is equivalent to $\vec{y}_k^1$.

2) We normalize the matrix C column-wise by replacing its elements as $C'_{ij} = C_{ij} / \sum_{k=1}^{N} C_{ki}$. Let us denote the resulting matrix as *C'*. Thus, each column of *C'* sums to one, and each element of column *j* is the expected proportion that the corresponding person will receive from person *j* when the current importance value of *j* is propagated through the network.

3) We make a linear interpolation of the link-structure matrix *C'* and the importance-sensitive matrix U_k as:

$$E_k = \alpha C' + (1-\alpha)U_k$$

where α is a constant in interval [0,1]. Matrix E_k is both irreducible and importance-sensitive.

Finally, we define the SIP (Semi-supervised Importance Propagation) method iteratively as:

$$\vec{y}_k^{t+1} = E_k\vec{y}_k^t = \alpha C'\vec{y}_k^t + (1-\alpha)U_k\vec{y}_k^t = \alpha C'\vec{y}_k^t + (1-\alpha)\vec{y}_k^1$$

Notice that $U_k\vec{y}_k^t = \vec{y}_k^1\vec{1}\vec{y}_k^t = \vec{y}_k^1$. Because matrix E_k is irreducible, vector $\vec{y}_k$ stabilizes when *t* is large. This yields the fixed point equation:

$$\vec{y}_k = E_k^t\vec{y}_k$$

The solution $\vec{y}_k$ is the principal eigenvector of matrix E_k, consisting of the expected importance score of each contact person after iterative SIP. Applying this method to each importance level, we obtain vectors $\vec{y}_k$ for $k = 1, 2, \ldots, 5$. These vectors provide 5 additional features (with the corresponding weights) in the enriched representation of the contact person of each email message, in the input vector for importance prediction using a SVM.

4.2 Connections between SIP and Topic Sensitive PageRank

Our formulae for SIP are quite similar to those in PageRank [3], Topic Sensitive PageRank (TSPR) and Personalized PageRank (PPR) methods when a topic distribution is used to represent the interest of each user [10]. In fact our SIP method is intrigued by the TSPR and PPR work. The main differences in our problem and the SIP solution are:

- Our graph structure has two types of nodes (i.e., persons and messages) while the graph structures in TSPR and PPR (and in PageRank) has only one type of nodes (i.e., web pages). Consequently, we have two types of links with different semantics (i.e., "who sends what" and "what is received by whom" respectively) while there is only one type of links (directed) in conventional link analysis methods.
- We are modeling the transitivity of human-judged importance for one type of the nodes (i.e., email messages) which are partially labeled, to another type of nodes (i.e., email contact persons) which are unlabeled, and we propagate the importance values among persons

iteratively. The network structure and the corresponding matrices are different from that in TSPR, PPR and PageRank. More importantly, the semantic concept is fundamentally different. That is, SIP focuses on propagation of importance values among persons based on their email connections while TSPR, PPR and PageRank focus on probabilistic transitions in random walk over web pages.

- We use the resultant vectors in SIP to obtain enriched features for representing the sender and receivers of each email, as a part of the vector representation of each new email and the input of SVM classifiers. TSPR, PPR and PageRank are designed for ranking documents with respect to each query. Other than the above, our formulae are indeed quite similar to those in TSPR, PPR and PageRank. The convergence analyses for those methods and the formulae of the close-form solution (i.e., the principal eigenvector) of the transition matrix also apply here; we omit those details.

5. EXPERIMENTS

5.1 Data

We recruited 25 experimental subjects, mostly from the Language Technologies Institute at the Carnegie Mellon University, including eight faculty members, five staff persons and twelve graduate students. Each subject was requested to label at least 400 non-spam messages during a one-month period. The five importance levels are: absolutely non-important, relatively non-important, neutral, important, and most important. Only seven users actually submitted more than 200 messages with importance labels, which we use to construct the dataset for the experiments in this paper. Table 1 summarizes the dataset statistics.

Table 1: Summary Statistics of collected dataset (7 users)

	Total	Train	Test
User 1	1750	150	1600
User 2	376	150	226
User 3	484	150	334
User 4	569	150	419
User 5	233	150	83
User 6	279	150	129
User 7	234	150	84
Avg	561	150	411

5.2 Preprocessing

We applied a multi-pass preprocessing to email messages. First, we applied email address canonicalization. Since each person may have multiple email accounts, it is necessary to unify them before applying social network analysis. For instance, "John Smith" john.smith+@cs.xxx.edu, "John" smith@cs.xxx.edu and "John Smith" john747@gmail.com might be the email addresses of the same person. We used regular expression rules and a longest string matching algorithms to identify email addresses which may belong to the same user. We then manually checked all the groups and corrected the errors in the process. We also applied word tokenization and stemming using the Porter stemmer; we did not remove stop words from the title and body text.

5.3 Features

The basic features are the tokens in the sections of *from*, *to*, *cc*, *title*, and *body text* in email messages. Let us use a *v*-dimensional vector to represent those features for each email message where *v* is the vocabulary size. We call it the *basic feature* (BF) sub-vector.

The social-network based features are represented as follows:. We use a *m*-dimensional sub-vector to represent the *Newman cluster* (NC) features (Section 2) where *m* be the number of clusters produced by the clustering algorithm: each element of the sub-vector is 1 if the user belongs to the corresponding cluster, or 0 otherwise; each user can belong to one and only one cluster. We also use another sub-vector (7-dimensional) to represent the *social importance* (SI) features per user, whose elements are real-valued (Section 0). In addition, we use a 5-dimensional sub-vector to represent the five SIP scores per user, i.e., the mixture weights of the user at the five importance levels. The concatenation of those sub-vectors together with the basic feature (BF) vector yields a synthetic vector per email message as its full representation.

5.4 Classifiers

We use five linear SVM classifiers for the prediction of importance level per email message. Each classifier takes the vector representation of each message (as described in the above section) as its input, and produces a score with respect to a specific importance level. The importance level with the highest score is taken as the predicted importance level by our system for the input message. We used the standard SVM^{light} software package and tuned the margin parameter *C* in the range from 10^{-3} to 10^{3}. We tuned the parameter with ten-fold cross validation of training data; we repeat the random split 10 times, and report the average performance on the test sets. To obtain a performance baseline, we ran the SVM classifiers with the basic features (BF) only as the input vectors. We also ran the classifiers with additional features, namely, BF+NC for using basic features plus the Newman-cluster (NC) features, BF+SI for using basic features plus the social importance (SI) features, BF+SIP for using basis features plus SIP features, and their complete combination, namely BF+NC+SI+SIP.

5.5 Metric

We use *MAE* (Mean Absolute Error) as the main evaluation metric, which is standard in evaluating systems that produce multi-level discrete predictions. *MAE* is defined as:

$$MAE = 1/N \sum_{i=1}^{N} |y_i - \hat{y}_i| ,$$

where *N* is the number of messages in the test set, y_i is the true importance level of message *i*, and $\hat{y}_i$ is the predicted importance level for that message. Since we have five levels of importance, the *MAE* scores range from zero (the best possible) to four (the worst possible).

There are two conventional ways to compute the performance average over multiple users. One way is pooling the test instances from all users to obtain a joint test set, and computing the MAE on the pool. This way has been called *micro-averaged* MAE. The other way is to compute the MAE on the test instances of each user and then take the average of the per-user MAE values. This way has been called as macro-averaged MAE. The former gives each instance an equal weight, and is possibly dominated by the system's performance on the

Table 2: Detailed evaluation results of SVMs with each representation scheme and varying training-set sizes. Macro-averaged MAE scores are provided with p-values, indicating the statistical significances of performance improvement over that of BF (using basic features alone). Numbers in bold fond indicate the best method for each fixed training-set size. One star indicates the p-values in (0.01, 0.05]; two stars indicate the p-values equal or less than 1%.

	BF	**BF+NC**		**BF+SI**		**BF+SIP**		**BF+SI+NC**		**BF+SI+NC+SIP**	
# of tr	MAE	MAE	p-value	MAE	p-value	MAE	p-value	MAE	p-value	MAE	p-value
10	0.9666	0.9063	* 0.0382	0.8837	* 0.0106	0.8968	* 0.0311	0.9112	* 0.0211	**0.8827**	** 0.0087
20	0.9720	0.8969	0.0506	**0.8596**	* 0.0315	0.9095	* 0.0435	0.9071	0.0558	0.8659	* 0.0235
30	0.9210	0.8318	* 0.0334	**0.7994**	* 0.0182	0.8224	* 0.0149	0.8305	* 0.0324	0.8096	* 0.0210
40	0.8851	0.7995	* 0.0239	0.7911	* 0.0367	0.8029	0.0587	0.8155	* 0.0465	**0.7869**	* 0.0279
50	0.8639	0.7820	* 0.0347	**0.7613**	* 0.0219	0.7900	0.0774	0.7766	* 0.0210	0.7625	* 0.0205
60	0.8447	0.7820	0.0890	0.7514	* 0.0416	0.7603	* 0.0463	0.7607	* 0.0284	**0.7363**	* 0.0198
70	0.8294	0.7662	0.0636	0.7218	* 0.0105	0.7679	0.1237	0.7560	* 0.0354	**0.7184**	* 0.0135
80	0.8257	0.7596	* 0.0494	0.7324	* 0.0261	0.7763	0.1678	0.7433	* 0.0250	**0.7157**	* 0.0109
90	0.8294	0.7521	* 0.0352	0.7295	* 0.0174	0.7598	0.0711	0.7315	** 0.0086	**0.7142**	** 0.0087
100	0.8127	0.7411	* 0.0225	0.7236	* 0.0180	0.7634	0.1629	0.7314	* 0.0184	**0.7098**	* 0.0103
110	0.8060	0.7268	* 0.0199	0.7168	* 0.0286	0.7542	0.1318	0.7142	* 0.0159	**0.7046**	* 0.0127
120	0.8105	0.7232	* 0.0183	0.7090	* 0.0154	0.7426	0.0727	0.7135	* 0.0144	**0.6960**	** 0.0071
130	0.8028	0.7207	* 0.0287	0.6980	* 0.0156	0.7449	0.0997	0.7105	* 0.0148	**0.6904**	** 0.0058
140	0.7960	0.7100	* 0.0136	0.7112	* 0.0262	0.7389	0.1039	0.7087	* 0.0176	**0.6842**	** 0.0050
150	0.7992	0.7073	* 0.0186	0.7178	0.0594	0.7412	0.0873	0.7034	* 0.0142	**0.6963**	* 0.0128
AVG	0.8510	0.7737	0.0360	0.7538	0.0252	0.7847	0.0862	0.7676	0.0862	**0.7449**	0.0139

data of a user who has the largest test set. The latter gives each user an equal weight instead. Both methods can be informative; therefore we present the evaluation results in both metrics.

We also conducted a one-sample t-test for assessing the statistical significance of performance improvement for SVMs with using different feature types in the input vectors. For example, for comparing SVM using BF+SI and the baseline SVMs (using BF only), we calculated the difference in the absolution error of the former and the absolution error of the latter on each test instance, and used the mean of the per-instance differences to estimate the p-value under the hull hypothesis (which assumes a zero mean).

5.6 Results

Figure 4 shows the performance curves of SVM runs with different representation schemes for email messages. Detailed scores in macro-averaged MAE are given in Table 2. It can be observed that BF had the worst performance. Using the social-network based features (NC, SI and SIP) features in addition significantly reduced the importance prediction errors in most cases for the training-set sizes we tested. On average, the relative error reduction in macro-averaged MAE is 14%, i.e., from 0.8510 to 0.7449 (see the last row of Table 2 and Figure 4b). The relative error reduction in micro-averaged MAE reduction (not shown explicitly in Table 2 but observable in Figure 4a) is 31%, i.e., from 0.7759 to 0.5909. All the training-set sizes are relatively small, compared to large data collections used in benchmark evaluations for text categorization, e.g., the RCV1 news-story collection has 780,000 training examples for 103 categories. This is exactly a part of the difficulty we must deal with for personalized email prioritization. It is evident in our results that personal social networks and semi-supervised importance can be effectively leveraged for addressing such the paucity of labeled training data.

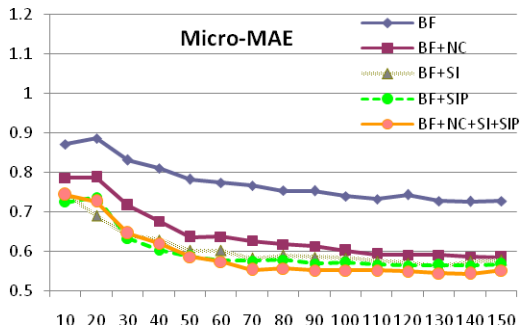
It can also be observed that using the complete combination of all the features (BF+NC+SI+SIP) is significantly better than adding each type of the social-network feature alone in most cases (graph a in Figure 4). As for using micro-average MAE as the performance measure (graph a in Figure 4), the complete combination of features had best results when the training-set sizes was not small (from the size of 50); BF+SI was the best for small training sets (of size 20). These observations suggest that social importance was better captured for most of the users when the training-set sizes were relatively small. Overall, using each type of social network feature alone may not be sufficient for characterizing the social roles and personal social networks of all the users. On the other hand, using the combining all the features enables us to model users with complementary features and hence to predict personal priorities robustly. Our detailed performance analyses (omitted here due to the space limit of the paper) on a per-user basis confirmed the above assertion.

6. RELATED WORK

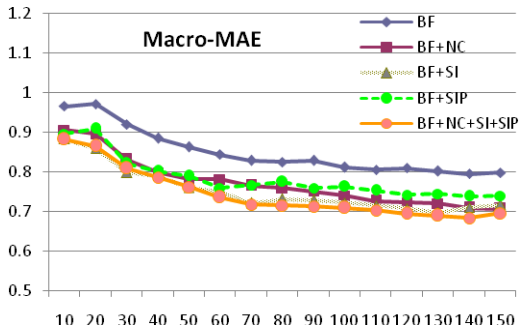
Statistical learning methods for performing email-based prediction tasks are becoming an increasingly important research area. We briefly discuss related methods with respect to their relevance to our work.

Among the early efforts in email prioritization, Horvitz et al. [11] built an email alerting system which used Support Vector Machines to classify newly arrived email messages into two categories, i.e., *high* or *low* in terms of utility. Probabilistic scores were also provided along with the system-made predictions. Personalization, however, was not considered in their method, and social network analysis was not their technical focus.

Tyler et al. [18] utilized Newman clustering algorithm to discover social structures automatically from email messages.



(a) Performance curves in micro-averaged MAE



(b) Performance curves in macro-averaged MAE

Figure 4 Performance curves in MAE (Mean Absolute Error). The horizontal axis is the training-set size used in the learning phase of SVMs. The vertical axis in graph (a) is the micro-averaged MAE, and in graph (b) is the macro-averaged MAE. A lower value in MAE means the better performance.

They found that the automatically-discovered social structures are quite similar, or consistent, with human interpretation of organizational structures. They also used email social networks to identify social leaders. However, they did not use the social network analysis (clusters or leadership scores) to prioritize email messages.

Gomes et al. [9] used email messages to automatically group users in two ways, i.e., by sender clusters and by recipient clusters, respectively. The senders were clustered based on similarity of their recipient lists, and the recipients were clustered based on similarity of their sender lists as well; email contents were not used. They examined the use of those clusters in spam detection, i.e., to separate spam messages from non-spam messages. Prioritization among non-spam messages, however, was not addressed.

Boykin and Roychowdhury [2] used clustering coefficients as enriched features to represent email messages and a Bayesian classifier to detect spam messages. Martin et al. [15] used the out-degree (the number of unique recipients) and in-degree (the number of unique senders) of each person in an email social network to detect worms which propagated through the email messages. Prioritization among non-spam messages was again not addressed by those methods.

Neustaedter et al. [16] defined metrics for measuring the social importance of individuals based on the observations in the email fields: from, to and cc, and in the recorded actions of replying and reading. They used these metrics for retrieving old email messages rather than prioritizing incoming email messages

Johansen et al. [14] proposed a social clustering approach to importance prediction of email messages. They collected email data from multiple users and induced social clusters of users. For each user, some clusters are treated as "important" and the others are not. The importance of each test instance of email message is predicted based on the cluster membership of its sender: if the sender belongs to an important cluster, then the messages is considered important; otherwise, it is predicted as not important.. The fundamental difference in their method from ours is that their clusters were induced from a community social network, not based on personal social networks. In addition, they only focused on social associations, not taking any textual features into account in the modeling and the prediction of importance.

Minkov et al. [7] used automatically-induced graphs to associate senders, email folders and messages, and a random-walk algorithm (e.g., a PageRank like method) to leverage the associations in predicting folders and recipients for email messages. . Email prioritization, personalized or otherwise, was not addressed in their approach.

In summary, there is a rich body of work in statistical learning approaches to email-based tasks. However, how to fully leverage personal email social networks in combination with email content for personalized email prioritization has not been studied in depth. Leveraging the good ideas in previous work and developing new techniques further with respect to personalized email prioritization is the unique focus and main contribution in this paper.

7. CONCLUSIONS AND FUTURE WORK

This paper presents the first study of personalized email prioritization under the assumption that only personal email data are available during the training and testing of the system. Specifically, we focus on social network analysis to capture user groups in each personal social network, and to obtain rich features for representing their user-centric social importance. We further developed a novel semi-supervised (transductive) learning algorithm that propagates importance values among nodes (messages or people) in each partially labeled and personal email network. These methods enable us to obtain an enriched vector representation of each new email message, as the basis of accurate modeling of individual users and for generating robust predictions for individual users in email prioritization. The effectiveness of the proposed approach is strongly evident in our experiments on personal email data from multiple users.

Future work would include collection of more data from a larger number of users and in a longer time period for thorough evaluation. We are also interested in a comparative study on different clustering algorithms and graph-mining techniques with respect to their effectiveness in mining social networks for personalized email prioritization.

8. ACKNOWLEDGEMENTS

This work is supported in parts by the Defense Advanced Research Project Agency (DARPA) under contract NBCHD030010, by the National Science Foundation (NSF) under grant IIS_0704689, and by Brain Korea 21 Project, the School of Information Technology, KAIST, in 2009. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

REFERENCES

[1] CEAS 2005 - Second Conference on Email and Anti-Spam, July 21-22, 2005, Stanford University, California, USA, 2005.

[2] P. O. Boykin and V. P. Roychowdhury. Leveraging social networks to fight spam. Computer, 38(4):61–68,2005.

[3] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst., 30(1-7):107–117, 1998.

[4] J. Cadiz, L. Dabbish, A. Gupta, and G. D. Venolia. Supporting email workflow. Technical Report MSR-TR-2001-88, Microsoft Research (MSR), Sept. 2001.

[5] K. M. Carley, D. Columbus, M. DeReno, J. Reminga, and I. Moon. Ora user's guide 2007. Carnegie Mellon University, SCS ISRI, Technical Report, (07-115), 2007.

[6] L. A. Dabbish and R. E. Kraut. Email overload at work: an analysis of factors associated with email strain. In P. J. Hinds and D. Martin, editors, Proceedings of the 2006 ACM Conference on Computer Supported Cooperative Work, CSCW 2006, Banff, Alberta, Canada, November 4-8, 2006, pages 431–440. ACM, 2006.

[7] R. B. Einat Minkov and W. Cohen. Activity-centred search in email. In Proceedings of the 5th Conference on Email and Anti-Spam (CEAS). CEAS, 2008.

[8] L. C. Freeman. The Development of Social Network Analysis: A Study in the Sociology of Science. Empirical Press, 2004.

[9] L. H. Gomes, F. D. O. Castro, V. A. F. Almeida, J. M. Almeida, R. B. Almeida, and L. M. A. Bettencourt. Improving spam detection based on structural similarity. In SRUTI'05: Proceedings of the Steps to Reducing Unwanted Traffic on the Internet on Steps to Reducing Unwanted Traffic on the Internet Workshop, pages 12–12, Berkeley, CA, USA, 2005. USENIX Association.

[10] T. Haveliwala, S. Kamvar, and G. Jeh. An analytical comparison of approaches to personalizing pagerank. Technical report, Stanford University, 2003.

[11] E. Horvitz, A. Jacobs, and D. Hovel.Attention-sensitive alerting. In K. B. Laskey and H. Prade, editors, UAI '99: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, Sweden, July 30-August 1,1999, pages 305–313. Morgan Kaufmann, 1999.

[12] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. J. ACM, 46(5):604–632, 1999.

[13] R. Likert. A technique for the measurement of attitudes. Archives of Psychology, 140:1–55, 1932.

[14] K. B. Lisa Johansen, Michael Rowell and P. McDaniel. Email communities of interest. In Proceedings of the 4th Conference on Email and Anti-Spam (CEAS).CEAS, 2007.

[15] S. Martin, B. Nelson, A. Sewani, K. Chen, and A. D. Joseph. Analyzing behavioral features for email classification. In CEAS [1].

[16] C. Neustaedter, A. J. B. Brush, M. A. Smith, and D. Fisher. The social network and relationship finder: Social sorting for email triage. In CEAS [1].

[17] M. E. J. Newman. Modularity and community structure in networks. 2006.

[18] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman. Email as spectroscopy: automated discovery of community structure within organizations. pages81–96, 2003.

[19] S. Wasserman and K. Faust. Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge, 1994.

[20] M. Wattenberg, Rohall, S. L., D. Gruen, and B. Kerr. E-mail research: Targeting the enterprise. Human-Computer Interaction, 20(1/2):139–162, 2005.

[21] Joshua Goodman, Gordon V. Cormack, and David Heckerman. Spam and the ongoing battle for the inbox. Commununications of the ACM, 50(2):24–33, 2007.

[22] M. Mojdeh and G. V. Cormack, Semi-supervised Spam Filtering: Does it Work?, SIGIR 2008.

[23] B. Klimt and Y. Yang. The Enron Corpus: A New Dataset for Email Classification Research. ECML 2004.

[24] R. Balasubramanyan, V. Carvalho and W. Cohen, CutOnce - Recipient Recommendation and Leak Detection in Action. In AAAI-2008, Workshop on Enhanced Messaging.

[25] P.N. Bennett and J. Carbonell (2007). Combining Probability-Based Rankers for Action-Item Detection. In Proceedings of HLT-NAACL 2007.

[26] A. McCallum, X. Wang and A. Corrada-Emmanuel. Topic and Role Discovery in Social Networks with Experiments on Enron and Academic Email. Journal of Artificial Intelligence Research (JAIR), 2007.

[27] D. Alwin, and J. Krosnick, "The reliability of survey attitude measurement: The influence of questions and respondent attributes", Sociological Methods Research, 1991, 20(139).