# Recipient Recommendation in Enterprises using Communication Graphs and Email Content

David Graus
University of Amsterdam
d.p.graus@uva.nl

David van Dijk
University of Amsterdam
d.v.vandijk@uva.nl

Manos Tsagkias
University of Amsterdam
e.tsagkias@uva.nl

Wouter Weerkamp
904Labs
wouter@904labs.com

Maarten de Rijke
University of Amsterdam
derijke@uva.nl

## ABSTRACT

We address the task of recipient recommendation for emailing in enterprises. We propose an intuitive and elegant way of modeling the task of recipient recommendation, which uses both the communication graph (i.e., who are most closely connected to the sender) and the content of the email. Additionally, the model can incorporate evidence as prior probabilities. Experiments on two enterprise email collections show that our model achieves very high scores, and that it outperforms two variants that use either the communication graph or the content in isolation.

## Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.3 Information Search and Retrieval

## Keywords

Recipient recommendation, email, generative models

## 1. INTRODUCTION

Despite the huge increase in the use of social media platforms, email remains one of the most popular ways of (online) communication. Messages that are sent around via email range from very informal, unimportant chatter to formal, official communication. In both cases it is important that the recipients of the emails are the correct ones: we want to avoid scenarios in which business contacts or clients receive email jokes or other chatter, and at the same time, sensitive, official communication intended for clients should not be sent to friends or family.

To prevent errors in assigning recipients to emails we can use recipient recommendation methods. These methods aim at providing the sender of an email with the appropriate recipients of the email that is currently being written. Previous attempts at recipient recommendation use the communication graph, constructed from previously sent emails [9], or the content of the current email [2]. Additionally, previous work typically focuses on predicting recipients when one or more seed recipients are given, also known as CC-prediction [4, 8]. Finally, previous work typically addresses the task of recipient prediction by restricting to a sender's ego network for prediction. In this paper, we focus on an enterprise setting, allowing us to leverage the full content and structure of the communication network, as opposed to taking a strictly local (ego network) approach. Combining signals from email content and the communication graph has been studied, e.g., in e-discovery, where combining keyword search with communication pattern analysis of e-mail corpora reduces the amount of information reviewers need to process [3]. Recipient recommendation similarly allows us to gain a better understanding of communication patterns in enterprises, potentially revealing underlying structures of enterprises. We propose a novel hybrid generative model, which incorporates both the communication graph and email content for recipient prediction. Our model predicts recipients without assuming seed recipients and can quickly deal with updates in both the communication graph and the profiles of recipients due to new emails being sent around.

Our main research question is whether combining the communication graph and email content improves recipient recommendation over using either of the two separately. On top of that we investigate how we can optimally estimate the various components of our model. We train our model on the Enron email collection and test the model on the Avocado email collection. Our main finding is that a combination of the communication graph and email content outperforms the individual components. We obtain optimal performance when we incorporate the number of emails a recipient has received so far and the number of emails a given sender sent to a recipient at that point in time in our model. Other options, like using PageRank as a recipient prior, do not improve performance.

## 2. COMMUNICATION GRAPH

We construct a communication graph from all emails sent by users in our email collections. We consider the email traffic as a directed graph $G$, consisting of a set of vertices and arcs (directed edges) $\langle V, A \rangle$. Each vertex $v \in V$ represents a distinct email address in the corpus (i.e., a sender $S$ or recipient $R$ in terms of our modeling in Section 3), and arcs $a \in A$ that connect them represent the communication between the two corresponding addresses (i.e., emails exchanged). The arcs are weighted by the number of emails sent from one user to the other.

The communication graph allows us to model the network and interactions by considering several graph-based metrics. One example is to measure a user's relative importance in the communication graph through her *PageRank*-score. The PageRank algorithm measures a users's relative importance through its connected arcs and their corresponding weights. In this model, an arc is a "vote


*SIGIR '14,* July 06–11, 2014, Gold Coast, QLD, Australia.

ACM 978-1-4503-2257-7/14/07 ... $15.00.
http://dx.doi.org/10.1145/2600428.2609514

of support," and thus users with a larger number of interactions receive a higher score [7]. We update our communication graph after each email that is being sent. We describe the utility of our communication graph for recipient recommendation in Section 3.

# 3. MODELING

We propose a generative model that is aimed at calculating the probabilities of recipients given the sender and the content of the email. Instead of recommending one recipient, we cast the task as a ranking problem in which we try to rank the appropriate recipients as high as possible.

More formally, let $R$ be a candidate recipient, $S$ the sender of an email, and $E$ the email itself. Our final ranking will be based on the probability of observing $R$ given $S$ and $E$: $P(R|S,E)$. We use Bayes' Theorem to rewrite this probability:

$$P(R|S,E) = \frac{P(R) \cdot P(S|R) \cdot P(E|R,S)}{P(S) \cdot P(E|S)}. \quad (1)$$

We can explain Eq. 1 as follows: the "relevance" of a recipient is determined by (i) its prior probability (how likely is this person to receive email in general), (ii) the likelihood of this email to be generated from communication between the recipient and the sender, and (iii) the probability of observing the sender with this particular recipient. To obtain the final probability, we normalize using the prior probability of the sender, and the likelihood of observing this email given its sender.

For ranking purposes we can ignore $P(S)$ and $P(E|S)$, which will be the same for all recipients. Our final ranking function is displayed in Eq. 2.

$$P(R|S,E) \propto P(E|R,S) \cdot P(S|R) \cdot P(R). \quad (2)$$

In the next three sections we explain how we estimate the three components of the model: the email likelihood ($P(E|R,S)$), the sender likelihood ($P(S|R)$), and the recipient prior ($P(R)$).

## 3.1 Email likelihood

We have several options when it comes to estimating $P(E|R,S)$. We could, for example, incorporate individual emails as latent variables. However, in this paper we opt to directly estimate the email likelihood using the terms in the email (viz. Eq. 3).

$$P(E|R,S) = \prod_{w \in E} \left[\lambda P(w|R,S) + \gamma P(w|R) + \beta P(w)\right]. \quad (3)$$

In this estimation, $P(w|R,S)$ indicates the probability of observing a term $w$ in all emails exchanged between $S$ and $R$. To prevent zero probabilities, we smooth this probability with the term probability in all emails sent and received by $R$ and the term probability over the whole collection. We introduce three parameters, $\lambda$, $\beta$, and $\gamma$, with $\lambda + \gamma + \beta = 1$, to combine the three term probabilities.

Each of the three term probabilities is calculated using the maximum likelihood estimate, i.e., $P(w|\cdot) = \frac{n(w,\cdot)}{|\cdot|}$, the frequency of term $w$ in the set of documents divided by the length of this set in number of terms.

## 3.2 Sender likelihood

We move to the estimation of $P(S|R)$, the likelihood of observing the sender for a given recipient. Here, we use the communication graph constructed from email exchanges. The closer a recipient is in this graph and the stronger his connection to the sender $S$, the more likely it is that the two "belong together." We estimate this connection strength in two ways: by considering (i) the *frequency* (*freq*), or the number of emails $S$ sent to $R$ at that point in time, and (ii) *co-occurrence* (*co*), or the number of times $S$ and $R$ co-occur as addressees in an email. More specifically, the frequency-based probability is defined as:

$$P_{freq}(S|R) = \frac{n(e, S \rightarrow R)}{\sum_{S' \in \mathcal{S}} n(e, S' \rightarrow R)}, \quad (4)$$

where $n(e, x \rightarrow R)$ indicates the number of emails sent from $x$ to $R$ and $\mathcal{S}$ is the set of all senders in the graph at the current point in time. The co-occurrence-based probability is defined as:

$$P_{co}(S|R) = \frac{n(e, \rightarrow R, S)}{n(e, \rightarrow R) + n(e, \rightarrow S)}, \quad (5)$$

where $n(e, \rightarrow R, S)$ corresponds to the number of emails sent to *both* $R$ and $S$, and $n(e, \rightarrow X)$ the incoming email, or number of emails sent to $X$.

## 3.3 Recipient likelihood

Finally, we introduce a recipient prior, that is, the email independent probability that a recipient will be observed. This probability is unrelated to both the email at hand ($E$) and the sender of that email ($S$) and can be estimated without knowing these two variables. Again, we can choose from a variety of ways to estimate this prior probability, but we stick to two obvious choices. First, we use the *number of emails received* by $R$, normalized by the total number of emails sent at that point in time (*rec*). This estimation indicates how likely it is that any given email would be sent to this recipient. Second, we calculate recipient $R$'s *PageRank* score (*pr*) as an indication of how important $R$ is in the communication graph.

# 4. EXPERIMENTAL SETUP

We put our model to the test using a realistic experimental setup. Our experiments aim at demonstrating the recommendation effectivenesss of the individual components of our model, i.e., the communication graph (CG) component and the email content component (EC), and their combination (CG+EC) as in Eq. 3. We optimize our models on the Enron email collection [5], and test it on the Avocado collection, which consists of email boxes of employees of an IT firm that developed products for the mobile Internet market. The two collections are described in Table 1.

For both collections we follow the same method to select the set of users for evaluation. We first split users into three groups based on their email activity: high activity, medium activity, and low activity. This way we can study the correlation between a user's level of activity and the model's performance. As email networks typically show a long-tailed distribution [6], with a small number of users responsible for a large volume of the sent mails, and a large number of users responsible for a small volume, we define a user's activity by taking the log of the number of sent emails. We prune users that have less then 100 sent mails, and compute the resulting distribution's mean ($\mu$) and standard deviation ($\sigma$) and split the distribution into three bins: (i) *medium* active users (MED) are those between $\mu - \frac{1}{2}\sigma$ and $\mu + \frac{1}{2}\sigma$, (ii) *highly* active users (HIGH) are the ones over $\mu + \frac{1}{2}\sigma$, and (iii) users with *low* activity (LOW) are those below $\mu - \frac{1}{2}\sigma$. From each bin we randomly sample 50 users, which results in our final evaluation set of 150 users.

## 4.1 Evaluation

Before we start recommending email recipients we allow our model to gather evidence from all email communication up to that point. More specifically, we use an initial *construction period* to generate the users' language models and the communication graph. We start to recommend recipients in the subsequent *testing period*.

**Table 1: Summary of Enron and Avocado email collections. We list the time span in months (Period), total number of Emails, total number of employee addresses (Addr.), the average number of emails sent ($\overline{S/p}$) and received ($\overline{R/p}$) per address.**

| | Period | Emails | Addr. | $\overline{S/p}$ | $\overline{R/p}$ |
|---|---|---|---|---|---|
| Enron | 45 | 252,424 | 6,145 | 34 | 294 |
| Avocado | 58 | 607,011 | 2,068 | 174 | 321 |

**Table 2: System performance (MAP) on the Enron dataset over different methods for estimating $P(R)$ and $P(S|R)$ (§3).**

| | $P_{pr}(R)$, $P_{co}(S\|R)$ | $P_{pr}(R)$, $P_{freq}(S\|R)$ | $P_{rec}(R)$, $P_{co}(S\|R)$ | $P_{rec}(R)$, $P_{freq}(S\|R)$ |
|---|---|---|---|---|
| LOW | 0.2207 | 0.1488 | 0.2317 | **0.4365** |
| MED | 0.1961 | 0.2334 | 0.2116 | **0.3857** |
| HIGH | 0.1016 | 0.1213 | 0.1169 | **0.2060** |
| ALL | 0.1755 | 0.1676 | 0.1893 | **0.3480** |

During both periods our model is updated for each sent email. We split each user's period of activity (starting from the user's first sent email, up to the last sent mail) into the *construction period*, covering $\frac{2}{3}$ of the emails, and the *testing period*, which is $\frac{1}{3}$ of the emails.

For each sender in our user evaluation set, we select 10 emails, evenly distributed over the testing period as evaluation points. For each of these emails we rank the top 10 recipients and compare to the actual recipients of the email. We report on mean average precision (MAP), as it allows us to identify improvements in the ranking of recipients. We indicate the best performce in bold face and test statistical significance using a two-tailed paired $t$-test. Significant differences are marked ▲/▼ for $\alpha = 0.01$ and △/▽ for $\alpha = 0.05$.

## 4.2 Parameter tuning

We use the Enron collection to tune the parameters $\lambda$, $\gamma$, and $\beta$ in Eq. 3, and to decide which methods for estimating the sender ($P(S|R)$) and recipient ($P(R)$) likelihood work best. The results of the parameter tuning are displayed in Tables 2 and 3. The final settings we use for testing our model on the Avocado collection are the following: $\lambda = 0.6$, $\gamma = 0.2$, $\beta = 0.2$, we estimate $P(S|R)$ using the number of emails $S$ sent to $R$, $P_{freq}(S|R)$, and we estimate $P(R)$ using number of emails received by $R$, $P_{rec}(R)$.

# 5. RESULTS AND ANALYSIS

We compare the found optimal settings for the CG and EC components to their combination using our training collection (Enron). Table 4 shows that our hybrid model significantly outperforms either of the single models across all groups of users in the Enron collection, even if the performance increase is modest. The highest performance is achieved in the LOW and MED user groups: lower user activity correlates positively with performance.

We present the results of our final experiments on the Avocado set in Table 4. Compared to the results on the Enron set, which we used for tuning our parameters, our model's performance is higher throughout on the Avocado set, both across the different models, and within each subgroup of users. An indication for this difference in absolute performance scores comes from the collection statistics in Table 1. Here we see that the Avocado set contains fewer unique addressees, spans a longer period of time, and contains a larger number of emails per person. As a possible factor contributing to our models' higher performance on the Avocado collection, we point to the fact that our model has more data available to leverage for ranking a smaller number of candidates. While different datasets may need different models, the consistently high scores show that the components work in isolation and in combination over different datasets.

**Table 3: System performance (MAP) on the Enron dataset over a parameter sweep for the parameters $\lambda$, $\gamma$, and $\beta = 1-(\lambda+\gamma)$ in Eq. 3 with a step size of 0.2.**

| $\lambda\downarrow/\gamma\rightarrow$ | 0.2 | 0.4 | 0.6 |
|---|---|---|---|
| 0.2 | 0.4670 | 0.4699 | 0.4752 |
| 0.4 | 0.5070 | 0.5095 | |
| 0.6 | **0.5258** | | |

**Table 4: System performance (MAP) on Enron and Avocado.**

| | Enron | | | Avocado | | |
|---|---|---|---|---|---|---|
| | CG | EC | CG+EC | CG | EC | CG+EC |
| LOW | 0.4365$^{\blacktriangledown}$ | 0.5757$^{\blacktriangledown}$ | **0.5833** | 0.6502$^{\blacktriangledown}$ | 0.6946$^{\blacktriangledown}$ | **0.7077** |
| MED | 0.3857$^{\blacktriangledown}$ | 0.5161$^{\blacktriangledown}$ | **0.5325** | 0.6052$^{\triangledown}$ | 0.6328$^{\blacktriangledown}$ | **0.6542** |
| HIGH | 0.2060$^{\blacktriangledown}$ | 0.4779$^{\triangledown}$ | **0.4853** | **0.6652**$^{\triangle}$ | 0.5739$^{\blacktriangledown}$ | 0.6136 |
| ALL | 0.3480$^{\blacktriangledown}$ | 0.5258$^{\blacktriangledown}$ | **0.5362** | 0.6402 | 0.6352$^{\blacktriangledown}$ | **0.6597** |

## 5.1 Between groups comparison

Similar to what we saw in the experiment on the Enron collection (Table 4), higher user activity generally seems to result in lower performance on the Avocado dataset (Table 4). The CG model is an exception and outperforms our combined model for highly active users. The combined model achieves significant performance improvements over the content model in each subset of users.

To better understand these patterns, we turn to the characteristics of the different user subgroups. We plot the users' numbers of emails (both sent and received), indicating their activity, and juxtapose it to the size of their *egonet*, which corresponds to the set of directly connected neighbors in the communication graph [1]. This *egonet* represents the users they interact(ed) with, and is indicative of a user's reach or embedding inside the communication graph.

The resulting plot is shown in Figure 1. There is a clear clustering: highly active users who send and receive a large number of emails, also have a larger number of people they interact with. While more textual content allows the generative model to create richer recipient profiles, in turn enabling more informed recipient ranking, there is a catch to a larger egonet too. The sender-recipient communication smoothing in our generative model results in a larger number of high-scoring candidates for highly active users. This makes it more difficult for the ranker to discern the true recipient(s) from the larger pool.

## 5.2 Performance over time

Motivated by the fact that our model is updated at each sent email, we study its performance over time. To get an indication of whether our model's performance improves or deteriorates over time, we apply linear regression on the data points of each model, and fit a trend line. Both the EC and combined model's trend lines have positive slopes at $1.36 \cdot 10^{-4}$ and $1.27 \cdot 10^{-4}$ respectively, whilst the CG model has a negative slope ($-1.38 \cdot 10^{-4}$). This indicates that our language modeling approach benefits from the larger amount of textual content it receives for each recipient over time, allowing the generation of richer recipient profiles for better email likelihood estimations. The CG approach on the other hand, deteriorates over time, suffering from the increased size and complexity of the communication graph. We note that, as time pro-

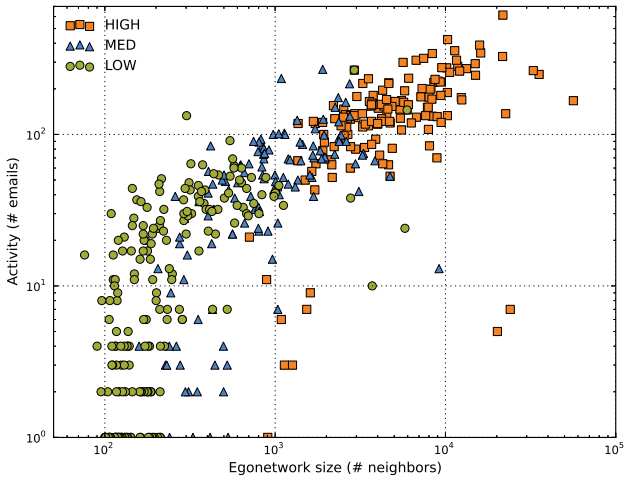


**Figure 1: The three user groups in Avocado, showing each user's activity against the size of its egonetwork.**

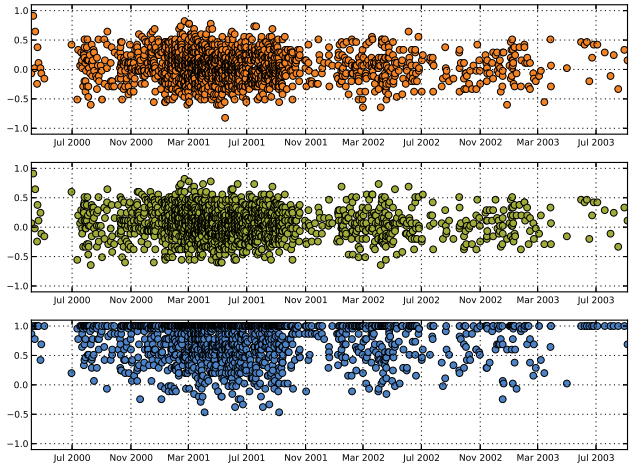


**Figure 2: Kendall's $\tau$ over time, between CG and EC (orange), CG and CG+EC (green), and EC and CG+EC (blue).**

gresses, the communication graph model "settles in," and becomes less likely to pick up on changing balances or shifting communication patterns in the communication graph. For future work we argue for a time-aware model, that is able to adapt to shifts in the communication graph over time, by taking recency into account.

## 5.3 Rankers' correlation

To analyze the performance of our combined model in comparison to the isolated ones, we look at their rankings and compute the Kendall tau rank correlation coefficient ($\tau$) between pairs of models. The top plot in Figure 2 shows how agreement between CG and EC is relatively low, centering around the 0 mark with an average of 0.0471. This pattern largely coincides with that of the second plot, which depicts agreement between CG and our combined model. The average correlation coefficient is only slightly higher at 0.0562. Finally, the agreement is highest between the EC model and our combined model, at on average 0.7425. The high agreement offers an explanation for the comparatively low performance of our combined model in the HIGH subset of the Avocado set. The EC model's comparatively low performance (as seen in Table 4), indicates that the combined model is negatively affected by following EC's incorrect rankings.

# 6. DISCUSSION AND CONCLUSIONS

In this paper we presented a novel hybrid model for email recipient prediction that leverages both the information from an email network's communication graph and the textual content of the messages. Our model starts from scratch, in that it does not assume or need seed recipients, and it is updated for each email sent. The proposed model achieves high performance on two email collections.

We have shown that the number of received emails is an effective method for estimating the prior probability of observing a recipient, and the number of emails sent between two users is an effective way of estimating the "connectedness" between these two users, and proves to be a helpful signal in ranking recipients.

We identified characteristic weaknesses of our individual models' robustness to specific circumstances. As witnessed by the decrease in performance for highly active users, our email content model seems unfit to deal with users that exchange a large number of emails with a large number of people. The deteriorating performance over time shows that our communication graph models' performance suffers with expanding graphs as they develop.

To address both these issues, for future work we propose to incorporate time into our model. We can, for example, use decay functions to weigh the edges between users and promote more recent communication. Similarly, we can use time-dependent language models that favor recent documents to incorporate time.

**Acknowledgments.** This research was partially supported by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements nr 288024 (LiMoSINe) and nr 312827 (VOX-Pol), the Netherlands Organisation for Scientific Research (NWO) under project nrs 727.011.005, 612.001.116, HOR-11-10, 640.006.013, the Center for Creation, Content and Technology (CCCT), the QuaMerdes project funded by the CLARIN-nl program, the TROVe project funded by the CLARIAH program, the Dutch national program COMMIT, the ESF Research Network Program ELIAS, the Elite Network Shifts project funded by the Royal Dutch Academy of Sciences (KNAW), the Netherlands eScience Center under project number 027.012.105 the Yahoo! Faculty Research and Engagement Program, the Microsoft Research PhD program, and the HPC Fund.

# REFERENCES

[1] L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *PAKDD'10*, 2010.

[2] V. R. Carvalho and W. W. Cohen. Ranking users for intelligent message addressing. In *ECIR'08*, 2008.

[3] H. Henseler. Network-based filtering for large email collections in e-discovery. *Artif. Intell. Law*, 2010.

[4] Q. Hu, S. Bao, J. Xu, W. Zhou, M. Li, and H. Huang. Towards building effective email recipient recommendation service. In *SOLI '12*, 2012.

[5] B. Klimt and Y. Yang. The enron corpus: A new dataset for email classification research. *ECML '04*, 2004.

[6] A. McCallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *J. Artif. Int. Res.*, 2007.

[7] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

[8] C. Pal and A. McCallum. Cc prediction with graphical models. In *CEAS*, 2006.

[9] M. Roth, A. Ben-David, D. Deutscher, G. Flysher, I. Horn, A. Leichtberg, N. Leiser, Y. Matias, and R. Merom. Suggesting friends using the implicit social graph. In *KDD '10*, 2010.