# You Will Get Mail!
# Predicting the Arrival of Future Email

Iftah Gamzu[1], Zohar Karnin[1], Yoelle Maarek[1], David Wajc[2]
[1]Yahoo Labs, Haifa, Israel
[2]Carnegie-Mellon University, Pittsburgh, USA
iftah.gamzu@yahoo.com, {zkarnin,yoelle}@ymail.com,david.wajc@yahoo.com

## ABSTRACT

The majority of Web email is known to be generated by machines even when one excludes spam. Many machine-generated email messages such as invoices or travel itineraries are critical to users. Recent research studies establish that causality relations between certain types of machine-generated email messages exist and can be mined. These relations exhibit a link between a given message to a past message that gave rise to its creation. For example, a shipment notification message can often be linked to a past online purchase message. Instead of studying how an incoming message can be linked to the past, we propose here to focus on predicting future email arrival as implied by causality relations. Such a prediction method has several potential applications, ranging from improved ad targeting in up sell scenarios to reducing false positives in spam detection.

We introduce a novel approach for predicting which types of machine-generated email messages, represented by so-called "email templates", a user should receive in future time windows. Our prediction approach relies on (1) statistically inferring causality relations between email templates, (2) building a generative model that explains the inbox of each user using those causality relations, and (3) combining those results to predict which email templates are likely to appear in future time frames. We present preliminary experimental results and some data insights obtained by analyzing several million inboxes of Yahoo Mail users, who voluntarily opted-in for such research.

## Categories and Subject Descriptors

H.4.3 [**Information Systems Applications**]: Communications Applications—*Electronic Email*

## 1. INTRODUCTION

Today's Web consumer email is dominated by non-spam machine-generated email messages [2, 8]. More than 90% of traffic originates from mass senders like social networks and ecommerce sites. These email messages range from slightly annoying wide-spread newsletters and promotions to critical information such as e-tickets and booking confirmations. Machine-generated email messages represent a great ground for research and data mining as they keep reoccurring over time, and over thousands if not millions of users. Indeed, by design, a single script generates numerous variations of messages for similar types of activities. Mailing lists represent the simplest form, where all messages are almost identical, while more sophisticated messages, such as travel itineraries and purchase orders, use a common boilerplate that is instantiated according to the user, transaction details, and more.

Some of the authors of this paper have previously introduced a method for identifying these similar messages by leveraging the notion of "email template" [2]. This method follows the intuition that if several messages share a common sender and a variation of subject lines, they should have been generated by a same script, and conceptually belong to a same equivalence class that can be mapped into a unique email template. In this previous work, we identified about 12,000 templates on a one month period of email traffic and then learned *causal relations* between pairs of templates. A typical example of causality relation is a purchase confirmation email that is followed by a shipping notification message for the acquired product. In this case, the purchase confirmation is said to cause the shipping notification. We used such causality relations to link a machine-generated message to a previously received message that might have caused it.

In this work, we tackle a more ambitious task. We are trying to link a message to a future email message that has yet to be delivered, by extrapolating from causality relations. More specifically, we introduce a novel approach for predicting which email templates a user should receive in selected future time frames. Our prediction approach relies on (1) statistically inferring causality relations between email templates by analyzing large amounts of email messages across a large number of users, (2) building a generative model that explains the inbox of each user using those causality relations, and (3) combining those results to predict which email templates are likely to appear in future time frames. We describe experimental results obtained by analyzing several million inboxes of Yahoo Mail users, who voluntarily opted-in for such research.

This line of research has several potential applications. One example is a new kind of advertising mechanisms that present ads that are not directly related to a current email message, but rather related to future predicted email. For instance, if it is observed that users flying to a specific Greek island often make an online booking of a cruise from a leading provider soon after, then it seems intuitive to show them competing cruise advertisements as soon as they receive their travel itinerary. Another use case for such a prediction method is a mail client feature that identifies that important messages, like shipment notifications, were not received within some expected delay. In this case, the mail system could suggest users to check their spam folders for the message, and if they still cannot

*WWW 2015 Companion,* May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2741694.

find it, to inquire with the relevant ecommerce site about the shipment. Note that the work presented here represents only a first step in investigating the predictability of future email arrival. We reserve the demonstration of its value in some of the above use cases to future work.

## 2. RELATED WORK

Prediction of future events has been given a lot of attention recently. In several papers, the predicted events are news items, political events, or crises [18, 6, 12, 16, 17]. The data that is used for those predictions commonly originates from various websites and social networks. In Radinsky et al. [16], future news events are predicted by extracting causal relations from past data, and applying them on recent events. Other relevant research has explored movie ticket sales prediction using signals derived from social media platforms such as Twitter [13, 4, 9]. Several other papers deal with the problem of identifying bursts or trends of content type as soon as they appear [10, 11, 3, 7]. An example of a burst can be a sudden interest in a specific performer due to a significant event in her life. This phenomenon may be reflected by many search queries. This setting has a slightly different flavor from our setting in the sense that we do not exploit a specific extreme event such as a burst, but rather predict future events based on past occurrences and causality relations. In [1, 15], prediction of future events was made via modeling user behavior. In the context of email, Dabbish et al. [5] suggest a method for predicting whether an email message will be replied to or not. To the best of our knowledge, this work is the first to explore the specific task of predicting the arrival of future email.

## 3. AN EMAIL PREDICTION MODEL

### 3.1 Email templates and causal threads

The notions of email templates and causal threads were introduced by a subset of the authors of this paper in [2]. Email templates provide an abstraction of the groups of machine-generated messages that were generated by a same script. They are identified by finding the commonality among numerous similar messages from the same mass sender. For privacy preserving reasons and performance considerations, the commonality is derived only from the header of such messages, and more specifically from the sender and subject line fields. A template is then represented as a pair (sender, subject mask), where the subject mask is a regular expression that covers many variations of a same subject. For example, the template (facebookmail.com, "$\star$ has commented on your status") will represent a message sent by facebookmail.com to a given user, with the subject line "Bonnie Parker has commented on your status", as well as another message sent to another user with the subject line "Clyde Barrow has commented on your status", under the rationale that both should have been generated by the same script. Other examples of frequent templates include (ups.com " ups ship notification tracking number $\star$"), or (americanairlines.com,"$\star$ your flight from $\star$ to $\star$"). Email templates were used in the same work [2], in order identify *causal threads*. Unlike usual conversation threads that group together messages between a same group of senders who reply to each other after an initial message, causal threads link machine-generated messages that were indirectly caused by an action reflected in an initial message, such as a purchase order or a travel reservation. Take as an example the sequence of email arrival: an email from ebay.com congratulating a user for her successful bid on a collector's item and asking him to pay for this purchase, followed by a receipt notification from paypal.com for payment to eBay inc, and then by two additional ups.com notification messages, one about the shipment, and one about the subsequent delivery of the item. By observing such patterns of messages through their templates over a large number of users, causal thread rules between templates can be automatically derived. The template-based rule representing the above sequence of events is given below.

ebay.com:"congratulations! your bid for $\star$ won!"
→ paypal.com:"receipt for your payment to $\star$"
→ ups.com:"ups.com shipment notification"
→ ups.com:"ups package arrived"

Such rules are then used to link an incoming machine-generated message to a sequence of previously delivered messages that might have caused it. We note that the method for predicting messages represented by their templates is independent from the template discovery process, and should work with any similar abstractions of mail messages.

### 3.2 A generative model for an inbox

We assume a generative model for an inbox, where every message appears due to one of the following reasons: (1) a *temporal event* occurring regularly on a weekly, monthly or other basis (e.g., a monthly statement from the bank or a weekly newsletter), (2) a *previous email message* that caused it (e.g., in the above, the previous email for the ups.com shipment notification is the receipt message sent by paypal.com) and, (3) an *unobserved action* that occurred externally and is not documented in the inbox (e.g., a purchase in an online store is an unobserved action for a receipt message that is sent later on).

Note that our model assumes that every message can give rise to at most one message, which holds true in most cases. It also assumes the existence of three (almost) independent processes, each generating a different type of messages. The first process selects messages resulting from unobserved actions (type 3 above), while the second generates messages that result from temporal events (type 1). The messages generated by these two processes define a seed set for a third process, which generates messages that are caused by previous messages (type 2). The two latter processes are similar in the sense that they account for messages generated by observable events. We later refer to such relations that correspond to observable events as causal relations.

### 3.3 A three-step prediction approach

An email message $E$ is abstracted as a pair $(T, t)$ where $T$ is its template and $t$ is its delivery time. Given an inbox with messages and temporal events $\mathcal{I} = \{E_1, \ldots, E_n\}$, a current time $t_{\text{cur}}$, and a time window size $\Delta$, our goal is to return a list of templates that may occur in the time window $[t_{\text{cur}}, t_{\text{cur}} + \Delta]$. Each template is associated with a score that indicates how likely this template to appear in the mentioned time window. Our prediction approach works as follows. As a preliminary step, we discover causal relations, which are pairs $(S, T)$, where $S$ is either a template or a temporal event, and $T$ is a template, such that given an appearance of $S$ it is likely that $T$ follows it, or given an appearance of $T$ it is likely that $S$ appears beforehand. This step is done once on all our email data, and not repeated for each inbox. We analyze each of the inboxes using our generative model. We utilize the statistically inferred causal relations to discover the probability of each email message in the inbox to cause any other message within the inbox. This step is crucial for our prediction step since any email message that already gave rise to a message cannot cause another message by our assumption. The core prediction step consists of generating a small set of candidate templates whose chances of appearing is non-negligible. We associate with each candidate a score representing the probability of its appearance in the given time window. These three steps of (1) discovering events and causal relations, (2) analyzing inboxes and (3) the actual template prediction, are detailed below.

## 3.4 Observed events and causal relations

We consider two kinds of causal relations: (1) between two templates, and (2) between a temporal event and a template. Recall that our data includes inboxes in which all messages arrived in the time window $[t_{\text{bgn}}, t_{\text{cur}}]$. For a template or a temporal event $S$ and a template $T$, we define $\text{Cnt}_1(S,T)$ as the overall number of times the pair $(S,T)$ was observed, where $S$ appeared at least two weeks prior to $t_{\text{cur}}$. Note that we increment this counter only for pairs $(S,T)$ in which $S$ appeared before $T$ and no other $S$ or $T$ appeared between them. Similarly, $\text{Cnt}_2(S,T)$ is the analog that counts the number of times that the pair $(S,T)$ was observed, but $T$ appeared at least two weeks after $t_{\text{bgn}}$. Note that we use these (two weeks) truncated counts to avoid inaccuracies that relate to the exact times that our time window begins and ends. We also define $\text{Cnt}_1(S)$ as the total number of times $S$ was observed at least two weeks prior to $t_{\text{cur}}$, and $\text{Cnt}_2(T)$ as the number of times $T$ was observed at least two weeks after $t_{\text{bgn}}$. We first focus on the case that $S$ is a template. We say that $S$ *causes* $T$ if

$$\frac{\text{Cnt}_2(S,T)}{\text{Cnt}_2(T)} > \theta_1, \text{and } \frac{\text{Cnt}_2(S,T)}{\text{Cnt}_2(T)} \cdot \frac{\text{Cnt}_2(S)}{\text{Cnt}_2(T,S)} > \theta_2 \,. \quad (1)$$

In a similar way, we say that $T$ *is caused by* $S$ if

$$\frac{\text{Cnt}_1(S,T)}{\text{Cnt}_1(S)} > \theta_1, \text{and } \frac{\text{Cnt}_1(S,T)}{\text{Cnt}_1(S)} \cdot \frac{\text{Cnt}_1(T)}{\text{Cnt}_1(T,S)} > \theta_2 \,. \quad (2)$$

where $\theta_1$ and $\theta_2$ are parameters (eventually given corresponding values of 0.3 and 2 after a grid search). In both equations, the first inequality asserts a correlation between $S$ and $T$. Informally, in equation 1 we get that $\Pr[S \mid T]$ is large, and in equation 2, we get that $\Pr[T \mid S]$ is large. The latter inequality in both equations filters spurious relations, where both $S$ and $T$ are caused by unobserved events[1]. We emphasize that although those causal relations may seem similar or even identical, they are inherently different. For example, suppose $S$ is a purchase notification template of a very small vendor and $T$ is a shipment notification template of a prime corporation. It is quite natural that $S$ causes $T$. However, the appearance of $T$ does not indicate that it was caused by $S$ since only a small fraction of the shipments done by the corporation are for this small vendor. As another example, consider the case that $S$ is a product promotion template and $T$ is a purchase notification template. This time it seems natural to say that, given $T$, it was caused by $S$. However, the opposite does not seem true, that is, the appearance of $S$ does not indicate that it causes $T$ since promotions are more likely not to lead to purchases.

We now turn to analyzing whether a temporal event $S$ causes a template $T$. In this case, it seems less natural to focus on the corresponding counts since temporal events are consistently interleaved with templates, and hence, some relations may look spurious. Therefore, we rather validate that the time difference between $S$ and $T$ is consistent. This is achieved by validating that the variance of the time differences is small. More formally, for a template or a temporal event $S$ and a template $T$, we define $\delta(S,T)$ to be the random variable that captures the time difference for the pair $(S,T)$ in which $S$ appeared before $T$ and no $S$ or $T$ appeared between them. We statistically infer this random variable from the email data. We also define $\delta(S)$ to be the (same) time difference between two consecutive temporal events of $S$. We say that the temporal event $S$ *causes* $T$ if

$$\text{stdev}(\delta(S,T)) < \theta_3 \cdot \delta(S) \,, \quad (3)$$

where $\theta_3$ is a parameter (eventually given the value of 0.1 by a grid search).

Let $S$ be a template or a temporal event. We define $\text{Follow}(S)$ to be the set of templates that may be caused by $S$, that is, the union of all templates $T$ such that $S$ causes $T$. Note that given the appearance of a template $S$ at most one $T \in \text{Follow}(S)$ may appear due to $S$ by our assumption. However, in case that $S$ is a temporal event, each $T \in \text{Follow}(S)$ can appear independently. We also define $\text{Precede}(T)$ to be the collection of templates that may give rise to the appearance of template $T$, namely, templates $S$ such that $T$ is caused by $S$.

### *3.4.1 Analyzing an inbox*

The goal of this step is to identify causal relations between messages in the inbox. This step is essential since any message that already gave rise to a message within the inbox cannot cause another message. Consequently, we should ignore those messages when we predict the arrival of future email messages. We explain below how to estimate for each message, the probability that it causes some other message in the inbox as well as the probability that it is caused by some message.

We first focus on an important feature in our approach, namely *time differences*. Recall that $\delta(S,T)$ is a random variable that captures the time difference for the pair $(S,T)$, where $S$ is a template or a temporal event and $T$ is a template. We assume that the distribution $\mathcal{D}_\delta(S,T)$ describes the random variable $\delta(S,T)$. As $\mathcal{D}_\delta(S,T)$ is unknown, we approximate it by considering all the time differences for the pair $(S,T)$ in our email data, and remembering their 99 percentiles, while assuming a uniform distribution within each bucket. More formally, let $t_1, \ldots, t_{99}$ be the 99 percentiles of the observed time differences between $S$ and $T$. We assume that $F_{S,T}$, the probability density function of $\mathcal{D}_\delta(S,T)$, has a probability of $1/(100(t_{i+1} - t_i))$ for any value in $[t_i, t_{i+1}]$. Here, $t_0 = 0$ and $t_{100}$ is the maximum time difference.

Let us focus on an inbox, and let $E = (T,t)$ be an email message in the inbox. Suppose the inbox consists of the messages and temporal events $E_1 = (S_1, t_1), \ldots, E_k = (S_k, t_k)$ such that,

1. $t_i \le t$ for every $i \in [k]$, and
2. $T \in \text{Follow}(S_i)$ or $S_i \in \text{Precede}(T)$ for every $i \in [k]$.

We posit that the probability that the message $E$ was not caused by any of $E_1, \ldots, E_k$ is

$$p_T = \frac{\text{Cnt}_2(\neg(S_1, \ldots, S_k), T)}{\text{Cnt}_2(T)} \,,$$

where the expression $\text{Cnt}_2(\neg(S_1, \ldots, S_k), T)$ stands for the number of times template $T$ is observed in our email data, while there are no appearances of $S_1, \ldots, S_k$ during some (parametrized) time period preceding it (eventually two weeks were selected by a grid search). We emphasize that although $p_T$ is not an accurate estimation of the probability that $E$ was not caused by any of $E_1, \ldots, E_k$, it is a simple expression that provides a good approximation for a carefully selected time period. In the case that $E$ was caused by one of $E_1, \ldots, E_k$, which happens with probability $1 - p_T$, we assume that the probability that $E_i$ gave rise to $E$ is proportional to

[1] Note that this approach does not remove *all* spurious relations, yet it does eliminate the vast majority of them, which is sufficient for practical purposes.

$$\frac{\mathrm{Cnt}_1(S_i,T)}{\mathrm{Cnt}_1(S_i)} \cdot F_{S_i,T}(t-t_i) ,$$

where $F_{S_i,T}$ is the (approximate) probability density function of $\mathcal{D}_\delta(S_i,T)$ which is generated using the process described before.

The above discussion accounts for the distribution of messages or events being the parent (or cause) of a specific email message. We are interested in the probabilities of all relations within an inbox. To estimate these probabilities, we pick those relations at random according to the calculated distribution. Specifically, given an inbox $E_1,\ldots,E_n$, we go over the messages in the inbox from the oldest one to the most recent. For each message under consideration, we randomly assign a parent (or possibly reflect the case in which a message does not have a parent) according to the above-mentioned probability distribution. Based on our random choices, we adjust the probabilities, and continue to the next message. In particular, if we decide that message $E_i$ is the parent of $E_j$, we alter the probability that $E_i$ is the parent of messages $E_k$ such that $k > j > i$ to zero, and normalize the relevant probabilities. We repeat the above-mentioned random process that identifies parental relations multiple times to get more accurate estimations $p_{\mathrm{parent}}(E_i)$ (respectively, $p_{\mathrm{child}}(E_i)$) of the probability that each message $E_i$ caused another message (respectively, was caused by a message or event) in the inbox.

### 3.4.2 *Predicting future templates*

Given an inbox that consists of the messages and temporal events $E_1 = (S_1,t_1),\ldots,E_n = (S_n,t_n)$, we want to predict the templates that will arrive within the next $\Delta$ time units. We first compute a list of candidate templates that have a non-negligible chance of appearing. This collection is composed of all the templates in the inbox and all the templates in $\cup_{i=1}^{n}\mathrm{Follow}(S_i)$. Let $\mathcal{T}$ be this set of templates. We compute a score for each $T \in \mathcal{T}$ that indicates how likely it is to appear within the given time window. The list of templates that we finally predict consists of all templates whose score exceeds some threshold, which can be adjusted to increase recall or precision.

In order to assign a probability score to a given template $T \in \mathcal{T}$, we first calculate $\alpha_0$, which is defined as the probability that $T$ appears due to an unobserved event in the next $\Delta$ time units. We model this generation process as a series of Bernoulli trials that decides, in each time unit, whether $T$ arrives. For this purpose, we consider the collection $\mathcal{C}$ of all messages $E_i$ whose template is $T$. We let

$$\mathrm{Cnt}(\mathcal{C}) = \sum_{E\in\mathcal{C}} (1 - p_{\mathrm{child}}(E)) ,$$

be the (fractional) expected number of messages in the inbox that appeared due to an unobserved event. We use $q_T$ to mark the probability that $T$ spontaneously appears in a single time unit. Specifically, $q_T = \min\{1, \mathrm{Cnt}(\mathcal{C})/(t_{\mathrm{cur}} - t_{\mathrm{bgn}})\}$. Consequently, the probability that $T$ appears in a time window of length $\Delta$ is

$$\alpha_0 = 1 - (1 - q_T)^{\Delta} .$$

In order to account for the creation of $T$ due to causality relations, we consider only such templates and temporal events $S_i$ such that $T \in \mathrm{Follow}(S_i)$. Let us assume without loss of generality that every $S_i$ satisfies this property. In case $S_i$ is a template, we define

$$\beta_i = \frac{\mathrm{Cnt}_1(S_i,T)}{\mathrm{Cnt}_1(S_i)} \cdot \int_{t=t_{\mathrm{cur}}-t_i}^{t_{\mathrm{cur}}-t_i+\Delta} F_{S_i,T}(t)dt .$$

This quantity is the estimate of the probability that $S_i$ causes $T$. Dealing with temporal events is a bit more tricky since temporal events occur periodically, and thus, there are such future events that occur after $t_{\mathrm{cur}}$ but before $t_{\mathrm{cur}} + \Delta$. Clearly, we like to take them into account. In case that $S_i$ is a temporal event that happens every $t_{\mathrm{spn}}$ time units, we define

$$\beta_i = \frac{\mathrm{Cnt}_1(S_i,T)}{\mathrm{Cnt}_1(S_i)} \cdot \int_{t=\max\{t_{\mathrm{cur}}-t_i,0\}}^{\min\{t_{\mathrm{cur}}-t_i+\Delta,t_{\mathrm{spn}}\}} F_{S_i,T}(t)dt .$$

Recall that an email message can cause at most one other message. Since some of the messages in the inbox may have already caused other messages in the inbox, we set

$$\alpha_i = \beta_i \cdot (1 - p_{\mathrm{parent}}(E_i)) ,$$

for all messages $E_i$. This term is the estimate of the probability that $E_i$ caused $T$ (given the underlying inbox). Note that if $E_i$ is a temporal event then we set $\alpha_i = \beta_i$. Since we assume independence between the different generation processes, we estimate the probability that $T$ is generated in the mentioned time window by

$$1 - \prod_{i=0}^{n} (1 - \alpha_i) .$$

# 4. DATA INSIGHTS

An important signal in our approach is the time differences between pairs $(S,T)$, where $S$ is a template or a temporal event and $T$ is a template. This information is utilized in both the inbox analysis and the prediction components. Given a pair $(S,T)$, it may seem natural to approximate the distribution $\mathcal{D}_\delta(S,T)$ by some standard distribution, such as Gaussian or Zipf, and mine its underlying parameters. This would allow us to represent the probability density function concisely and efficiently. Unfortunately, such an approach fails miserably. In almost all cases, the distribution $\mathcal{D}_\delta(S,T)$ has a complex structure. Furthermore, different $(S,T)$ pairs often exhibit widely-differing time differences behavior, which is very inductive to the relation between them (see the figures and discussion below). Luckily, as we are only interested in pairs $(S,T)$ that co-occur often, we obtain a large number of samples from the mentioned distribution. As a consequence, we decided to approximate the probability density function of $\mathcal{D}_\delta(S,T)$ by recording percentiles 1 through 99 of the observed time differences, while assuming a uniform distribution within each bucket defined by two consecutive percentiles. We note that such an approximation not only provides a good estimation of the unknown distribution, but also scales well.

Figure 1 includes a few examples that demonstrate how template pairs may have different distributions. For each of the graphs in the figure, the template pair is indicated in the caption above the graph, the $x$-axis represents the time differences between instances of that pair (with a step size of 1 hour), and the $y$-axis represents the number of pair instances that were observed. For example, the upper graph demonstrates the relation between the template "your ebay item sold .$\star$" from ebay.com and the template "we're transferring money to your bank" from paypal.com. This graph confirms for instance that most commonly, the transfer of money immediately follows the sale. Also, one can see that there is a decrease in the number of times money was transferred as the time passes (with the caveat that this overall trend exhibits diurnal fluctuations).

One can derive additional insights from these graphs. The second to bottom graph suggests that walmart.com most commonly sends a survey two weeks after a purchase has been done, while the bottom graph suggests that processing photo printing orders at walgreens.com usually takes less than 3 hours, although there are some orders that are ready after 10 hours (maybe due to nighttime). Another example, presented in Figure 2, shows that when users indicate that they have forgotten their password on ebay.com, it typically takes less than 10 minutes until they change it. There are

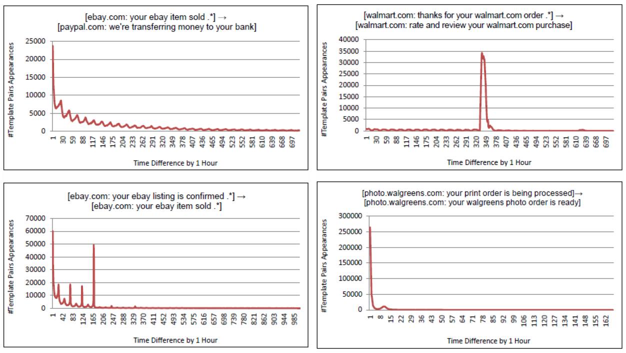

**Figure 1: Time difference distribution of template pairs.**

many other examples of interesting relations between templates, that we cannot include here for lack of space. These examples further support our modeling decision in representing the probability density function.

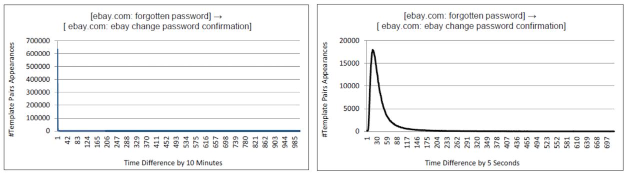

**Figure 2: Time difference distribution of a template pair.**

# 5. EXPERIMENTS AND RESULTS

We next describe the experiments that we conducted and their results. Note that all the email data used for those experiments was collected from users who voluntarily opted-in for such research.

## 5.1 Evaluation metrics

For the purpose of verifying the effectiveness of our prediction model, we use the traditional measures of *recall* and *precision*. We analyze each user's inbox over a given past time period, and then, predict the arrival of certain templates, representing machine-generated classes of email messages, in a future time period. In our experiments, we fixed the "past" to two months of incoming mail, while the "future" time period $\tau$ was set to cover a two weeks period. Given the inbox of a certain user, our model generates a list of predicted templates $P = (P_1, P_2, \ldots, P_n)$ that are sorted by decreasing order of confidence scores, that is, $P_1$ has the highest confidence score, and $P_n$ the lowest one. Note that each template in this list is unique. We mark by $F(\tau) = (F_1, F_2, \ldots, F_m)$ the collection of templates of messages that actually arrived during $\tau$. Note that each template in this collection is also unique; even if there are two or more messages that share the same template, $F$ contains only one instance of this template. Now, we can view the list $P$ as a set of retrieved results whose quality we want to evaluate against a ground truth of relevant documents, namely, $F$. With this definition in mind, recall is defined as $|P \cap F|/|F|$, and precision as $|P \cap F|/|P|$. More specifically, we computed $recall@k$ and $precision@k$, that is, the recall and precision scored achieved when predicting $k$ templates "from the future". Following the notation above, where $F$ is the true collection of future templates and $P$ is the ordered list of predicted templates, we denote by $P(k)$ the ordered list that consists of the first $k$ templates in $P$ (or fewer if there are less than $k$ templates in $P$). We now have

$$recall@k \triangleq \frac{|P(k) \cap F|}{|F|}, \text{and } precision@k \triangleq \frac{|P(k) \cap F|}{|P(k)|}.$$

## 5.2 Predicting the past

It has often been observed in the literature that naïvely predicting the exact behavior seen in the past achieves surprisingly good precision results on average. A typical example of this behavior is exhibited by the famous weather forecast *persistence method*, which predicts that tomorrow's weather will be identical to today's weather [14]. A simple prediction model in this vein for our task, which we refer to as *PastPredictor*, predicts that the templates to appear are those in the user's inbox with a confidence score proportional to the corresponding number of appearances in the inbox. Indeed, we also found that "predicting the past" results in reasonable precision and recall in our case; specifically, we achieved $recall@10 > 0.63$ and $precision@10 > 0.18$. However, in many cases of interest, it is important to keep some level of freshness and diversity. For instance, in advertisement scenarios, one wants to arouse attention and avoid ad blindness. Diversity is a recognized important factor in recommender systems. For example, in the domain of question recommendation in community-question answering, it has even been demonstrated that it is preferable to reduce relevance in favor of diversity and freshness [19]. Following the same intuition, we argue that predicting diverse templates that have not been observed in the past is important for the applicability of the model.

A possible attempt to add diversity to the prediction is to use a mix of past and popularity prediction. We call the obtained prediction model the *PastAndPopularityPredictor*. We use this model as a baseline for comparison with our prediction model, which we refer to as the *ThreadingBasedPredictor*.

## 5.3 Experimental setup and results

Our experiments leveraged over 10,000 templates, a sample data of over 40 million Yahoo mail users, and a grand total of over 1.5 billion email messages. We primarily measured $recall@k$ and $precision@k$ for the two predictors mentioned above: the *PastAndPopularityPredictor* and our *ThreadingBasedPredictor* .

In Figures 3 and 4, one can see that our model most commonly outperforms the *PastAndPopularityPredictor* in both precision and recall. We wish to emphasize that both predictors improve upon the recall rate of *PastPredictor*, which is limited to predicting the past templates of the user's inbox. Specifically, our model, *ThreadingBasedPredictor* achieves $recall@30$ of nearly $0.8$, improving upon a score of under $0.67$ obtained by the *PastPredictor*. In contrast, the competing predictor, *PastAndPopularityPredictor* achieves a score of less than $0.72$. This overall trend in recall scores between the 3 predictors is maintained for all values of $k$. We note that around the values of $k = 5$, the baseline has a slight advantage over our method with respect to the recall, yet this advantage is insignificant compared to its counterpart in the regime of larger $k$. Interestingly, this mentioned improvement in recall happens simultaneously to an improvement in precision.

# 6. CONCLUSIONS

We introduced a novel problem of predicting the future arrival of machine-generated emails. We presented a generative model for an inbox, along with an offline method to infer its parameters. Our approach builds upon mining users inboxes and identifying causal relations between email messages. To demonstrate the effectiveness of our approach, we collected email data from Yahoo mail

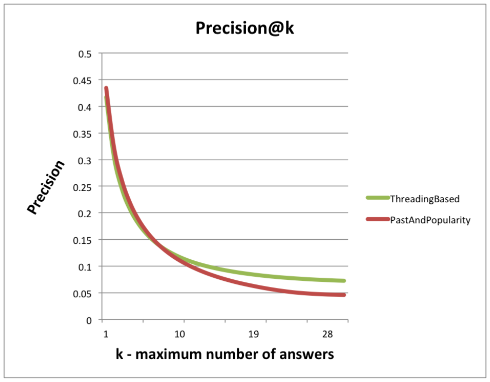

**Figure 3: Precision comparison between predictors.**

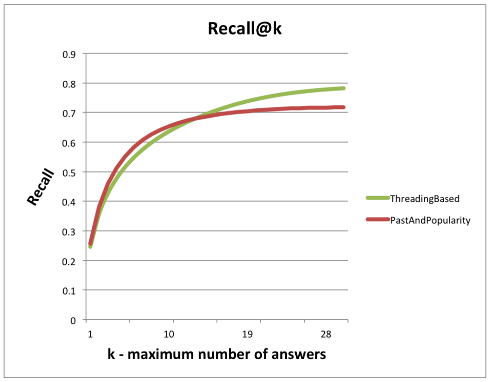

**Figure 4: Recall comparison between predictors.**

users who voluntarily opted-in to such research. After partitioning the data into train and test time periods, we showed that our method improves upon a strong baseline that mixes together popular messages and past messages of users. While these results are only preliminary, we believe that they open new grounds for research on mail data. In future work, we plan to continue improving our approach, refine our templates representation and enrich it with additional attributes extracted from email bodies. We also intend to verify the value of predicting email in a variety of application domains such as up-sell ads and anti-spam mechanisms.

We also like to emphasize that our prediction approach is general in the sense that it can be applied to other temporal problems that admit an underlying event generation process with causal relations. We classified machine-generated email messages using templates, and mined for causal relations between those events. One can similarly define classification in other domains (like categorization of tweets or social network posts), and use our approach to mine and predict causal relations over this classification domain. In particular, we believe that one interesting component of our temporal approach is the way we model different frequency patterns between events using time-based quantile histograms.

## Acknowledgments

We would like to thank David Carmel for sharing with us his knowledge on evaluation metrics. This work would not have been possible without the support of our colleague Guy Halawi, who masters the machine-generated mail processing system on which we relied. We are also grateful to our partners at Yahoo mail in Sunnyvale, CA; they taught us the wonders of large-scale consumer mail.

## 7. REFERENCES

[1] Eytan Adar, Daniel S Weld, Brian N Bershad, and Steven S Gribble. Why we search: visualizing and predicting user behavior. In *WWW*, pages 161–170, 2007.

[2] Nir Ailon, Zohar S. Karnin, Edo Liberty, and Yoelle Maarek. Threading machine generated email. In *WSDM*, 2013.

[3] Giuseppe Amodeo, Roi Blanco, and Ulf Brefeld. Hybrid models for future event prediction. In *CIKM*, pages 1981–1984, 2011.

[4] Sitaram Asur and Bernardo A Huberman. Predicting the future with social media. In *WI-IAT*, pages 492–499, 2010.

[5] L. Dabbish, R. Kraut, S. Fussell, and S. Kiesler. Understanding email use: predicting action on a message. In *SIGCHI*, pages 691–700, 2005.

[6] Jeremy Ginsberg, Matthew H Mohebbi, Rajan S Patel, Lynnette Brammer, Mark S Smolinski, and Larry Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457(7232):1012–1014, 2008.

[7] Nadav Golbandi, Liran Katzir, Yehuda Koren, and Ronny Lempel. Expediting search trend detection via prediction of query counts. In *WSDM*, pages 295–304, 2013.

[8] 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*, 2014.

[9] Mahesh Joshi, Dipanjan Das, Kevin Gimpel, and Noah A Smith. Movie reviews and revenues: An experiment in text regression. In *Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 293–296, 2010.

[10] Jon Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.

[11] Jon Kleinberg. Temporal dynamics of on-line information streams. *Data stream management: Processing high-speed data streams*, 2006.

[12] Kalev Leetaru. Culturomics 2.0: Forecasting large-scale human behavior using global news media tone in time and space. *First Monday*, 16(9), 2011.

[13] Gilad Mishne and Natalie S Glance. Predicting movie sales from blogger sentiment. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 155–158, 2006.

[14] Weather World 2010 project. Persistence method: today equals tomorrow, 2010.

[15] Kira Radinsky, Sagie Davidovich, and Shaul Markovitch. Predicting the news of tomorrow using patterns in web search queries. In *WI-IAT*, volume 1, pages 363–367, 2008.

[16] Kira Radinsky, Sagie Davidovich, and Shaul Markovitch. Learning causality for news events prediction. In *WWW*, pages 909–918, 2012.

[17] Kira Radinsky and Eric Horvitz. Mining the web to predict future events. In *WSDM*, 2013.

[18] Diana Eva-Ann Richards and Diana Richards Doyle. *Political complexity: Nonlinear models of politics*. University of Michigan Press, 2000.

[19] Idan Szpektor, Yoelle Maarek, and Dan Pelleg. When relevance is not enough: promoting diversity and freshness in personalized question recommendation. In *WWW*, pages 1249–1260, 2013.