MaiReS: Mail Recommender System for knowledge sharing

Tomoki Taniguchi, Yoshihiro Ueda, Motoki Taniguchi, Yasuhide Miura and Tomoko Ohkuma
Fuji Xerox Co., Ltd.

Abstract

Mail is a main communication channel in most offices. Despite the growing popularity of chat and social network services, mail has maintained its position as a very important communication tool for office workers. This fact implies that mail includes knowledge that office workers need. Sharing and reusing such mail messages help office workers in their daily work. We propose the MaiReS mail recommender system, which supports knowledge-sharing via mail. Here, knowledge is related to daily operations for some procedures such as obtaining a voucher or being paid back for a business trip and paperwork procedures such as making contracts. MaiReS is intended to provide mail messages that include knowledge that is useful to office workers for their procedures. Through preliminary analysis, we validated the MaiReS concept by analyzing mail messages collected using a prototype system. We further conducted a user study to confirm the MaiReS effectiveness. Results showed that MaiReS outperformed a naive baseline system without knowledge sharing in terms of the usefulness of recommended mail messages. Furthermore, we examined several features to characterize useful mail messages.

1 Introduction

Mail is a main communication channel in most offices. Chat and social networking services have shown growing popularity, but mail still holds a position as an important communication tool for office workers (Purcell and Rainie 2014).

Based on that fact, one can infer that mail includes knowledge that office workers need. Sharing and reusing such mail messages helps office workers in their daily work.

Here, we specifically examine knowledge related to daily operations for some procedures such as obtaining a voucher or being paid back for a business trip or paperwork procedures such as formulating contracts or finalizing agreements.

Knowledge related to work routines should be reused efficiently because the same operations and paperwork are frequently processed by office workers. With the aim of sharing such knowledge, they have been archived in documents such as manuals, databases, and FAQs. However local rules and exceptions sometimes exist for regulations. Furthermore, the regulations and document forms are updated with no prior announcement. Consequently, the knowledge not described in any documents but someone surely has is exchanged among workers through communication. We assume that recommendation of mail which is a main communication channel helps office workers.

This paper presents a proposal for a mail recommender system called MaiReS, which supports knowledge sharing using mail messages. MaiReS is designed to provide mail messages to office workers to support their procedures.

(Van Gysel et al. 2017) proposed a system for recommending attachable items from the user's past mail messages. They specifically examine the individual users' own mail messages. The difference between their work and our system is that our system enables inter-user mail recommendation. This attempt is the first ever reported to recommend other users' mail for knowledge sharing.

It is necessary to consider the confidentiality of mail messages (e.g. personal data) when sharing mail messages with other employees. Some mail messages include information that should not to be shared with other people. MaiReS asks an owner of a mail message to share the mail with other users by showing mail content information and the name of a person the mail is expected to be shared with. This process prevents the mail messages from accidental disclosure. We sent users a questionnaire about sharing mail data on MaiReS. Most respondents answered that they would permit it. We describe the results of this questionnaire in Section 2.

Figure 1 presents an overview of MaiReS. This system consists of three components for sharing useful mail messages among users: 1) A trigger detection component monitors newly arrived mail messages and finds mail that induces some operations for detecting user demands (trigger mail). Trigger mail includes a request for some response or asking some question to be answered. 2) A mail matching component retrieves mail messages most related to a trigger mail and validates their usefulness for the request included in a trigger mail. 3) A share permission component obtains a user's permission for sharing the mail message with the user who received a trigger mail. Permission requests and the final recommended mail messages are shown on each user's proprietary client application.

We conducted a user study to compare usefulness between mail retrieved from a user's mailbox and the mail shared on MaiReS in an actual office. We further investigated mail with user feedback to ascertain characteristics of the mail usefulness.

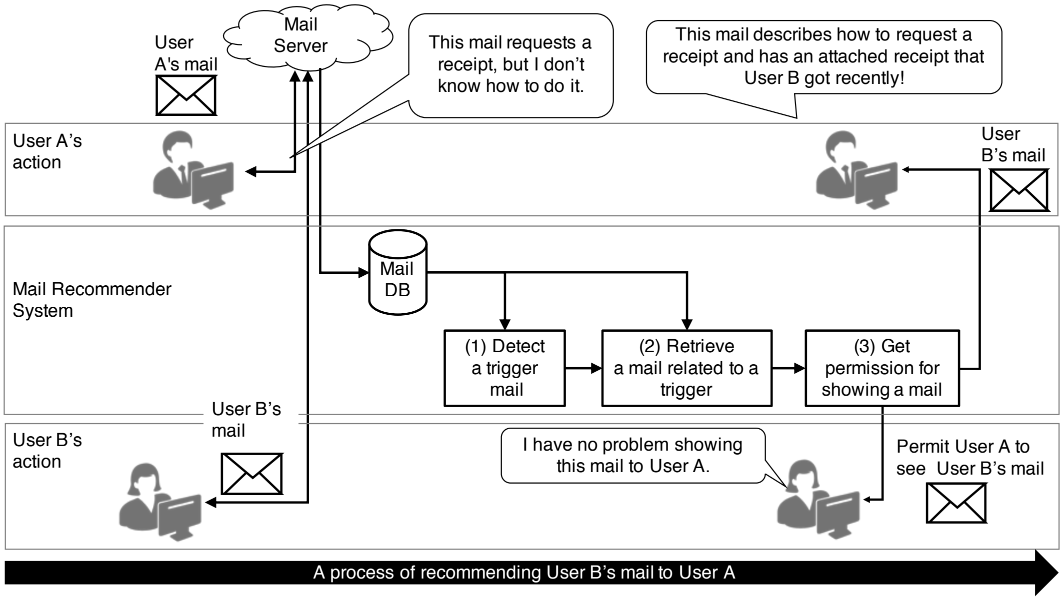

Figure 1: Process of sharing User B's mail with User A on MaiReS. The system crawls users' mail messages in the mail server. After pre-processing, the system detects a trigger mail and retrieves a mail message related to the trigger mail from DB. When User B, who is an owner of the most related mail, permits sharing the mail with User A, the system recommends User B's mail to User A.

Our key contributions are listed below.

1. We propose a mail recommender system for knowledge sharing among users.
2. We investigate our hypothesis that mail recommendation helps office workers through analyses of mail messages collected using a prototype system.
3. We demonstrate that MaiReS recommends more useful mail than that retrieved from one's own mailbox.
4. We examine several features to characterize useful mail messages.

We verify our hypotheses through an extensive net survey of crowdsourcing, an interview of office workers, and also mail data with user feedback collected using a prototype system described in Section 2. Details of MaiReS are described in Section 3. We compare the usefulness between a user's own mailbox and the mailbox shared on MaiReS in Section 4. The experimentally obtained result shows that recommendation from the mailbox shared on MaiReS is more useful than that from one's own mailbox in Section 5. We examine which features characterize mail usefulness and present examples of pairs of a trigger mail and the recommended mail in Section 6. We describe related work in Section 7.

2 Preliminary Analysis

We stated that (1-1) mail is a main communication channel in offices and that (1-2) office workers often share knowledge related to work routines via mail so that (2) recommending mail messages including such obtained knowledge can help them. To prove the first two statements before developing MaiReS, we conducted an extensive net survey to elucidate mail usage in offices. Furthermore, to make MaiReS truly useful, we developed a prototype to elicit users' feedback about (2-1) whether they can disclose their mail messages and (2-2) whether mail messages recommended by the retrieval method are useful.

2.1 Net Survey: Are mail messages used to share the knowledge about office procedures?

We assume that (1-1) mail is a communication tool used in the contemporary offices and (1-2) that requests to execute procedural operations or asking how to handle such operations are also exchanged by mail. To support these assumptions, we conducted a net survey of 1000 Japanese people, 475 of whom are office workers.

We asked these 475 office workers what communication tools they use in the office (allowing duplicates). The results are presented in Figure 2. From another question asking about the frequency of mail use in business, 343 people answered "always" and 113 "sometimes". Consequently, these results support our first assumption (1-1): mail is a main communication tool used in offices.

For additional analyses, we expect to use 456 people who always or sometimes use mail for some purpose in business.

The next question is designed to verify that the execution of procedural operations is requested by mail: 376 of 456 (82.5%) people responded that they have such experiences.

How about the assumption that know-how related to operational procedures is also shared using mail messages? First, we asked respondents how they find some knowledge when executing procedural operations. Of them, 393 (86.2%) people reported consulting their mail archives; 389 (85.3%) asked their colleagues.

The result of the latter case was also examined. Responses were received in two ways: direct answers or forwarded past

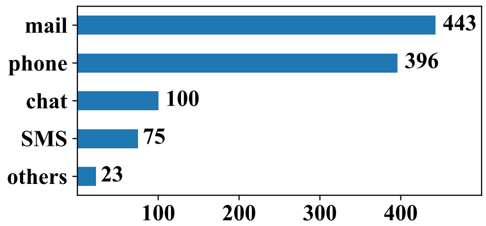

Figure 2: Communication tools used in the office. Multiple choice questions are answered by 475 office workers.

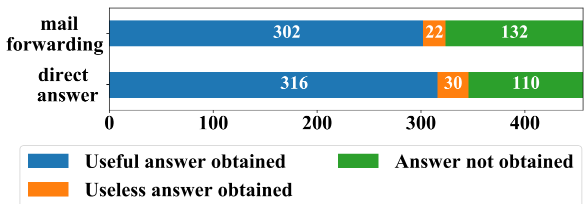

Figure 3: Means of responses to the information request. “Answer not obtained” includes the cases they didn’t request any.

mail messages that include related knowledge.

Figure 3 shows that three-fourths of the people had experienced obtaining information somehow and more than 90% among them responded that the obtained information was useful.

On the other hand, 333 (73.0%) people had experienced answering such requests by mail messages and 304 (66.7%) forwarding mail messages.

From the results described above, we verified (1-2) that office workers often share knowledge related to work routines in mail messages and verified that recommending such knowledge can help them.

We also administered the same survey to workers in an office who are subjects of our experiment described later. We received 48 responses. Among respondents, 46 (95.8%) reported experiences of asking for knowledge: 37 (77.1%) people obtained a direct answer that was useful; 41 (85.4%) people obtained forwarded related mail messages. Thus, the characteristics of the subject group of our experiment are much the same as the office workers among the net survey participants.

2.2 Evaluation by prototype users: Is recommendation by retrieval method useful?

The purpose of prototype is to collect user feedback and to use it to develop a target system. For this purpose, we added a feedback interface to the client application. First, to ascertain (2-1) whether users allow disclosure of message contents to others, the share permission component asks them not only to accept or refuse but also asks for the reason of refusal when refused. Second, to ascertain (2-2) whether mail messages recommended by retrieval method are useful, receivers are asked to assess the recommended mail, such as “useful,” “known (related and can be useful to other users),” or “related (but not useful).”

Accept	125 (77.2%)
Refuse / because it’s confidential	2 (1.2%)
Refuse / because it’s not correct	9 (5.6%)
Refuse / no reason provided	1 (0.6%)
No answer provided	25 (15.4%)

Table 1: Rate of acceptance / refusal and reasons for refusal. Here, 137 mail messages were judged by the owners on a prototype system.

This prototype differs from the target system in the following ways:

Trigger detection It detects a trigger mail using a simple combination of detection of query words related to requests, questions, etc. and simple rules such as including procedure names or document form names extracted by entity extraction.

Mail matching We retrieve mail messages related to a trigger mail using BM25, which is widely used in full-text search systems. A mail message which ranks first in BM25 scores was selected as recommended mail.

Reason for refusal of providing mail The acceptance / refusal result and the reason for refusal are shown in Table 1. Because the accept ratio excluding “no answer” is 0.91 (=125/137), most of the mail messages turned out to be disclosable to others.

We investigated the delay of acceptance, i.e. the delay of recommendation, for 125 accepted mail messages. Almost half (62) were given acceptance within one hour and almost 70% (87) within 24 hr. Therefore, the recommendation delay is rather slight.

For the results presented above, we found (2-1) that most mail messages can be shared with others and additionally found that the delay of recommendation because of the share permission component did not pose a major barrier.

User evaluation We asked “usefulness” and “relatedness” of the recommended mail messages to those who received them. “Usefulness” denotes whether the recommended mail is useful for operations. “Relatedness” denotes whether the recommended mail is related to the original trigger mail.

Users assessed 111 mail messages from 125 provided mail messages. The results are presented in Table 2.

“Useful” (22.5%) and “known” (18.9%) can be regarded as useful knowledge for a trigger mail. “Related” (27.9%) and “unrelated” (21.6%) represent useless information. Therefore, results show (2-2) that mail recommended by the retrieval method includes useful knowledge, but it turned out that about half of the mail is useless.

Furthermore, results show that related mail messages are not necessarily useful because many “related (but not useful)” mail messages are obtained in the assessment results.

assessment	description	#count	%
useful	useful	25	22.5
known	related and can be useful to other users	21	18.9
related	related but not useful	31	27.9
unrelated	unrelated and not useful	24	21.6
misdetection	trigger mail misdetection	10	9.0

Table 2: Evaluation of mail messages recommended by prototype system. Users evaluated 111 mail messages with the owner's permission.

3 Mail Recommender System:MaiReS

Design policy We designed MaiReS to recommend mail messages that include useful knowledge of operational procedures. In terms of practical application, we defined the design policy as follows.

- MaiReS uses a machine learning method that incorporates user feedback related to usefulness because about half of the mail messages retrieved by the full-text search system are assessed as useless in preliminary analysis.
- A simple and fast algorithm is adopted for the system because it is necessary to analyze numerous mail messages and to retrieve target mail messages from them in real time.

Client-server architecture MaiReS is implemented with a client-server architecture. The client is a proprietary application. Permission requests and the recommended mail messages are shown on the client. The client also has a feedback interface to collect user feedback. The server is synchronized with the mail server to collect mail data and archives mail messages into its mail database. Mail messages are analyzed to prepare recommendations. In Section 3.1 or later, we describe the server details. The server comprises three components: the trigger detection component, the mail matching component, and the share permission component.

3.1 Preprocessing

The preprocessing is divided into two modules. The first is message filtering. The second is entity extraction. Preprocessing is executed before archiving mail into the database. Preprocessing results are stored and used later for trigger detection and mail matching.

Message filtering Message filtering splits a mail message body into lines of sentences and classifies them into the message types shown in Table 3. If the system detects GREETING or SIGNATURE, then the line is removed. If QUOTE_START is detected, all subsequent lines are removed. As a result, lines classified as OTHERS or QUOTE are left.

We adopt the bag-of-words model using Support Vector Machine (SVM) with a linear kernel as a classification model. We train the model using a mail message corpus in business activities. We annotated 1335 mail messages manually and obtained labels. The number of annotated data is

message type	description	#count
GREETING	e.g., "Dear Jane Doe."	3021
SIGNATURE	e.g., "Best regards."	6100
QUOTE	A message into the reply. It provides a discourse context.	88
QUOTE_START	Quote in the end of a message.	621
OTHERS	Other than those above	18709

Table 3: Message types and number of annotated data. We annotated QUOTE_START when past mail messages are inserted at the end of the mail.

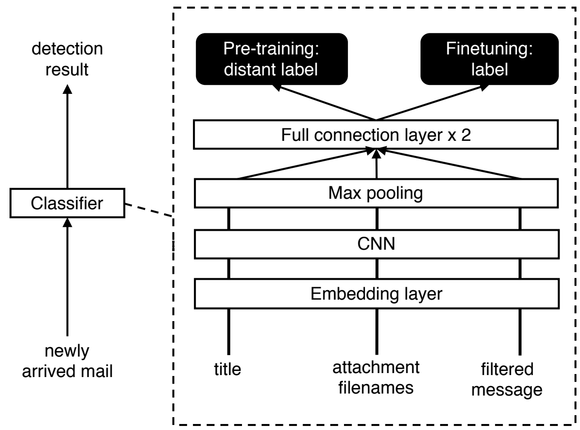

Figure 4: Trigger detection component.

shown in Table 3. Each message type is annotated line by line. The model trained using the corpus achieved a $F1$ score of 0.89.

Entity extraction In a mail message related to daily operation for some procedures, it can be expected that a characteristic from document form names such as "TRAVEL APPLICATION FORM" or procedure names such as "CONFIDENTIALITY AGREEMENT" frequently appears. These phrases can facilitate the detection of trigger mails and retrieve mail messages related to trigger mail messages.

Therefore, we adopt Conditional Random Field (CRF) to extract them from a mail message. CRF model accepts a sequence of characters as input and output begin, in, out (BIO) tags. We annotated 1715 mail messages manually and obtained labels. The $F1$ score of extracting procedure name from title was 0.808. That of extracting document form name from title was 0.755. The $F1$ score of extracting procedure name from body was 0.416. That of extracting document form name from body was 0.332.

3.2 Trigger detection

Figure 4 portrays a trigger detection component. The trigger detection component monitors newly arrived mail messages and discriminates a mail messages which induce some operations for detecting user demands. We use the deep neural network method proposed by (Kim 2014). The model accepts a title, attachment filenames, and filtered messages

category	feature	description
Amount of content (trigger)	*MailLength*	#characters in body
	HavingAttachment	whether a trigger mail has attachments
Amount of content (recommendation)	*MailLength*	#characters in body
	HavingAttachment	whether a candidate mail has attachments
Mail recency	*RecencyScore*	the time difference between the sent time of the mail messages
Entity match	*EntityMatchScore*	the Jaccard index of the two set of entities involved in a trigger and a candidate mail
Content match	*ContentMatchScore*	the cosine similarity between the TFIDF vectors of a trigger and a candidate mail: text of title and body is concatenated
People match	*PeopleMatchScore*	the Jaccard index of the two sets of people involved in a trigger and a candidate mail
Mail belonging	*MailBelonging*	whether a recipient of recommendation belongs to a candidate mail
Received type	*To*	whether a recipient directly receives a trigger mail
	CC	whether a recipient receives a trigger mail as a carbon copy

Table 4: Input features of LR model in the mail matching component. We prepared 8 categories and 11 features referring to an early study of (Zhao et al. 2018). The value range of each input feature is standardized.

obtained by preprocessing. Input text is transformed into an embedding space and is processed using Convolutional Neural Networks (CNN) and max pooling over time. After they are passed to full connection layers, we obtain the probability scores for triggering of a recommendation.

Training of the trigger detection component consists of two steps.

Pre-training We pre-train the model using distant labels. we construct a corpus with distant labels by a simple combination of using query words related to requests, questions, etc. and simple rules such as including procedure names or document form names extracted by entity extraction. Negative samples are obtained by random sampling from mail data stored in the database.

Finetuning We fine-tune the pre-trained model using a human-annotated corpus. We annotated 4078 mail messages. Annotators judged whether the presented mail messages were trigger mail or not. As annotation results, we obtained an imbalanced dataset. The dataset includes 584 positive and 3494 negative examples.

The trained model achieved the $F1$ score of 0.632 and the accuracy of 0.895. We used a softmax cross entropy loss function to build each model. We used Adam as an optimizer.

3.3 Mail matching

Figure 5 portrays a mail matching component. Considering that the system is required to retrieve a target mail from large amounts of data, we designed the mail matching component as a pipeline system. The mail matching component consists of two steps: we retrieve candidate mail messages using simple unsupervised method; then we validate the usefulness of candidates and select a mail message to be recommended.

Retrieve candidates We retrieve candidate mail messages related to a trigger mail using the BM25 algorithm. As search fields, we use a title, attachment filenames, and filtered messages. We use a title and attachment filenames of the trigger mail for search query. Mails with top-N scores by BM25 are selected as candidate mail messages.

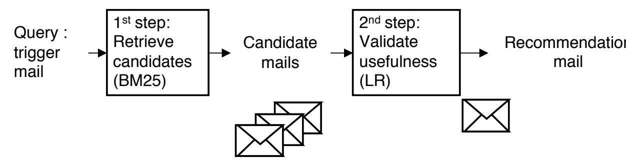

Figure 5: Mail matching component.

Validate usefulness To validate usefulness, we introduce the usefulness score, which represents the probability of the message being useful information. The usefulness score between a trigger mail and a candidate mail are calculated using logistic regression (LR). Then we sort the candidate mail messages in descending order according to usefulness scores. A mail with the highest usefulness score is selected as a target of recommendation[1].

Table 4 presents input features of the LR model. The model uses not only a content similarity but also meta-attributes such as mail recency and people match. The model also uses entity extraction result with a preprocessing. To train the LR model, we used the users' feedback results collected by the prototype system.

3.4 Share permission

Our system asks an owner of a recommended mail for sharing mail with other users. A mail owner decides whether to share the mail while referring to the provided information: a recipient, a title, a body, attachment files, and such. If it gets permission from the owner, the mail is supplied to the user who received the trigger mail. If the system recommends mail messages in one's own mailbox, the share permission process is skipped.

4 Experiment

In this section, we validate the effectiveness of knowledge sharing via mail by evaluating the various mail matching methods. We prepared comparative method of two types using different sources. One retrieved candidate mail messages

[1]We can select top-k examples as recommended targets.

from a user's own mailbox. The other retrieved candidate mail messages from both the user's own mailbox and other users' mailboxes. In other words, the difference is whether mail messages are shared with others.

4.1 User study

We conducted an experiment while running a mail recommendation system in an office. Each pair of a trigger and a recommended mail were shown to a user and were evaluated. Assessment labels are the same as those described in Section 2. We used the feedback interface of the client system to collect user assessments.

Subjects in the experiment were 11 workers in the research and development department. We used mail messages sent January 1, 2018 through October 16, 2018.

4.2 Proposed method setup

We used the users' feedback described in Section 2 and trained the LR model to calculate the usefulness score. MaiReS is intended to recommend mail messages that include useful knowledge irrespective of whether the recommended information is already known. Therefore, we binarize the user's feedback results as positive and negative with regard to usefulness. We integrated "useful" and "known" as positive and "related" and "unrelated" as negative. The candidate mail message size was set as 30.

4.3 Comparison methods

(Zhao et al. 2018) evaluated the influence of the content similarity and the elapsed time on usefulness. Referring to a report of an early study, we selected the following methods for comparison. The candidate mail message size was the same value as that used for the proposed method.

CONTENT-OWN CONTENT-OWN was a baseline method without knowledge sharing. It retrieved candidate mail messages from the user's own mailbox. It used the score of BM25 as the usefulness score. In other words, the mail message which ranks first in BM25 scores was selected as a recommended mail.

OURS-OWN *MailBelonging* feature was eliminated from the proposed method because it is always true. We trained OURS-OWN using the same as a corpus for the proposed method. OURS-OWN retrieved candidate mail messages from the user's own mailbox.

TIME Candidate mail messages were sorted in ascending order of the time difference between sent times. The most recent mail was selected as recommended mail. TIME retrieved candidate mail messages from both the user's own mailbox and other users' mailboxes.

CONTENT Calculation of the usefulness score is the same as the CONTENT-OWN. CONTENT retrieved candidate mail messages from both the user's own mailbox and other users' mailboxes.

5 Experimental Results

Figure 6 shows experimentally obtained results. It is noteworthy that MaiReS is intended to recommend mail messages that include useful knowledge irrespective of whether a user has already known the recommended information. From that perspective, we found that OURS can acquire the most positive ("useful" and "known") assessment and least negative ("related" and "unrelated") assessment among the compared methods. The p-value was 0.10 by Wilcoxon rank sum test between OURS and CONTENT-OWN that is a baseline method without knowledge sharing.

Furthermore, methods using knowledge sharing among users decreased "unrelated" assessment more than methods without knowledge sharing. Mail owners probably keep mail messages that are not informative for receiving users. This point turned out to be an unexpected point of effectiveness of the share permission component.

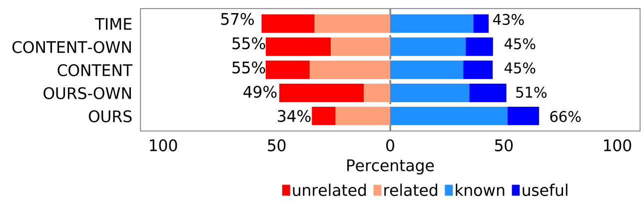

Figure 6: User feedback results about recommendations. The mail messages recommended by each method were evaluated by MaiReS users. The definitions of the labels are described in Table 2.

category	feature	weight
Amount of content (trigger)	*MailLength*	0.2680
	HavingAttachment	0.0629
Amount of content (recommendation)	*MailLength*	0.3139
	HavingAttachment	0.3497
Mail recency	*RecencyScore*	-0.4639
Entity match	*EntityMatchScore*	0.1549
Content match	*ContentMatchScore*	0.6270
People match	*PeopleMatchScore*	0.0134
Mail belonging	*MailBelonging*	0.0763
Received type	*To*	0.2092
	CC	-0.1770
Intercept		-0.1587

Table 5: LR weights for 8 categories and 11 features. Details of features are shown in Table 4.

6 Discussion

In this section, we examine several features to characterize useful mail messages. We further show actual useful example pairs of a trigger mail and a recommended mail.

6.1 Feature analysis of useful mail messages

The proposed LR model in the mail matching component was trained with user feedback related to usefulness. Learned weights of the model represent the contribution to usefulness. Therefore, we observed the weights of the model to analyze features that affect usefulness. Table 5 represents weights of the LR model.

From the table, we found that *ContentMatchScore* and *RecencyScore* have effects on the usefulness.

example 1	
trigger mail	
title	An application form of a business trip to a foreign country
body	Dear *PERSON_A* I'll take the day off tomorrow. Please send your application form to *PERSON_B.* … … Best regards, *PERSON_C*
recommended mail	
title	FW: Procedure of a business trip to a foreign country
body	Dear *PERSON_D* Please refer to an application form of business trip which I created. Don't forget to ask *PERSON_E* to check the sheet before submitting. … Best, *PERSON_F*

example 2	
trigger mail	
title	An application form of a housing allowance
body	Dear *PERSON_A* I want to confirm the content of your application form. - Do you live with your parents ? - The submitted documents were inadequate. Please submit all documents about housing contracts. - … Best regards, *PERSON_B*
recommended mail	
title	Update your housing allowance
body	Dear *PERSON_A* Your housing contract will be expired at September. Please access following URL and update your procedure about housing allowance. URL is [URL]. … Best regards, *PERSON_B*

Figure 7: Examples of "useful" mail messages user assessed. Title and body information of trigger and recommended mail messages are shown. These examples are simplified and translated from Japanese into English. User names are anonymized. Knowledge sharing was performed in example 1; Past own mail is recommended in example 2.

Next, it turned out that *HavingAttachment* in the category of the amount of content (recommendation) contributed to the usefulness. It is likely that office workers are requested to attach documents for some applications.

Additionally, we found that *EntityMatchScore*, which was calculated using entities extracted by preprocessing, had a somewhat positive influence.

However, it turned out that *MailBelonging* and *PeopleMatchScore* did not contribute to usefulness. That result implies that anyone's mail can be useful, which suggests that knowledge sharing via mail messages is valid.

It is particularly interesting that results demonstrated that the influence on usefulness varied by received type (*To* was positive but *CC* was negative). It is considered that a user who receives it directly by *To* can be requested to complete an operation for some procedure.

6.2 Examples of useful recommendation

Figure 7 shows mail examples that acquired a "useful" assessment.

Example 1 shows a pair of mail messages about the procedure related to a business trip. MaiReS recommended a past mail in the mailbox of *PERSON_D* when *PERSON_A* was requested to create an application form for a business trip. *PERSON_A* was able to refer to a past application form that someone wrote and knew a local rule ("Do'nt forget to ask *PERSON_E* to check the sheet before submitting it.").

Example 2 shows a pair of mail messages about the procedure used for a housing allowance. MaiReS recommended a past mail to *PERSON_A* in the own mail box of *PERSON_A* when *PERSON_A* was required to resubmit it because of the inadequacy of a document. It reminded *PERSON_A* about some original information related to the housing allowance, which enabled *PERSON_A* to access to website of the company using the supplied URL.

7 Related Work

Mail is a major communication tool that has been used for decades. It is now expanding its role to archiving data, schedule management, and task management, even though the purpose of mail systems was initially just communication (Grevet et al. 2014). Our research is closely related to mail retrieval, mail organization, information extraction from mail data and recommendation of mail data. This section presents an overview of the related studies.

Retrieval MaiReS detects a trigger mail including some requests to receivers and then finds mail messages that involve most related and useful contents from mail messages crawled in advance and stored in the database. The past research analyzed the usage of mail search.

(Ai et al. 2017) found that people use mail search not merely to find information but also to clean up or to organize it by analyzing large-scale mail search logs. They reported that not only keywords but also meta-information such as the sender and date are used for searching. Also, (Alrashed, Awadallah, and Dumais 2018) reported a user behavior of searching the same mail data numerous times. (Carmel et al. 2015) reported that attributes of mail data related to time are important to mail search.

Organization To reduce the user's time spent organizing large mail data with various contents, many researchers have tackled this task. (Avigdor-Elgrabli et al. 2016) proposed a method for classifying mail messages generated by a system automatically. (Avigdor-Elgrabli et al. 2018) proposed a method using semantic relatedness between mail messages to look up the content quickly. (Cohen, Carvalho, and Mitchell 2004) discussed using machine learning methods to classify mail according to the intent of the sender.

Information extraction To extract personal data pieces for many types of applications, (Di Castro et al. 2018) proposed a method for information extraction from automatically generated mail using an advanced structural clustering technique. (Sheng et al. 2018) used templates for safety information extraction.

Recommendation Some systems have been proposed for recommending information for supporting users in creating mail. (Qadir et al. 2016) modeled activities as latent probability distributions personalized to the mail sender and demonstrated the model's effectiveness in a mail recipient recommendation task. Other researchers also have worked on the recipient recommendation task. (Graus et al. 2014) proposed a method for this task, which used both a communication graph and mail contents. (Van Gysel et al. 2017) proposed a system recommending attachable items to a user based on a weakly supervised learning framework. (Kannan et al. 2016; Henderson et al. 2017) proposed a method for automatically generating short mail responses from a received mail. (Zhao et al. 2018) developed a system that proactively selects and displays potentially useful mail messages to users based on their upcoming calendar events with a particular emphasis on meeting preparation.

All of these recommender systems tend to recommend personal information owned by the user. However, the purpose of our proposed system is to recommend and provide a user's mail to other users who demand certain knowledge. Our system is designed based on cooperation realized by sharing mail among office workers in good faith.

8 Conclusion and future work

We proposed MaiReS for knowledge sharing via mail. We analyzed mail messages collected using a prototype system to confirm the hypothesis that mail sharing facilitates the daily operations of office workers. Analysis results show that almost all mail messages (91%) were disclosable to others. Results also show that recommended mail messages include useful knowledge. In the experiment, we further conducted a user evaluation to confirm the effectiveness of MaiReS. Results showed that MaiReS outperformed a naive baseline system without knowledge sharing in terms of the usefulness of recommended mail messages. Furthermore, we examined several features to characterize useful mail messages. As future work for this study, we first plan to validate the proposed system through experiments with more subjects. Moreover, we would like to apply the online learning algorithm to MaiReS for additional improvement.

References

Ai, Q.; Dumais, S. T.; Craswell, N.; and Liebling, D. 2017. Characterizing email search using large-scale behavioral logs and surveys. In *Proceedings of the 26th International Conference on World Wide Web*, 1511–1520.

Alrashed, T.; Awadallah, A. H.; and Dumais, S. 2018. The lifetime of email messages: A large-scale analysis of email revisitation. In *Proceedings of the 2018 Conference on Human Information Interaction and Retrieval*, 120–129.

Avigdor-Elgrabli, N.; Cwalinski, M.; Di Castro, D.; Gamzu, I.; Grabovitch-Zuyev, I.; Lewin-Eytan, L.; and Maarek, Y. 2016. Structural clustering of machine-generated mail. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 217–226.

Avigdor-Elgrabli, N.; Gelbhart, R.; Grabovitch-Zuyev, I.; and Raviv, A. 2018. More than threads: Identifying related email messages. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 1711–1714.

Carmel, D.; Halawi, G.; Lewin-Eytan, L.; Maarek, Y.; and Raviv, A. 2015. Rank by time or by relevance?: Revisiting email search. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 283–292.

Cohen, W. W.; Carvalho, V. R.; and Mitchell, T. M. 2004. Learning to classify email into "speech acts". In Lin, D., and Wu, D., eds., *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 309–316.

Di Castro, D.; Gamzu, I.; Grabovitch-Zuyev, I.; Lewin-Eytan, L.; Pundir, A.; Sahoo, N. R.; and Viderman, M. 2018. Automated extractions for machine generated mail. In *Companion Proceedings of the The Web Conference 2018*, 655–662.

Graus, D.; van Dijk, D.; Tsagkias, M.; Weerkamp, W.; and de Rijke, M. 2014. Recipient recommendation in enterprises using communication graphs and email content. In *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1079–1082.

Grevet, C.; Choi, D.; Kumar, D.; and Gilbert, E. 2014. Overload is overloaded: Email in the age of gmail. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 793–802.

Henderson, M.; Al-Rfou, R.; Strope, B.; Sung, Y.; Lukács, L.; Guo, R.; Kumar, S.; Miklos, B.; and Kurzweil, R. 2017. Efficient natural language response suggestion for smart reply. *CoRR* abs/1705.00652.

Kannan, A.; Kurach, K.; Ravi, S.; Kaufmann, T.; Tomkins, A.; Miklos, B.; Corrado, G.; Lukacs, L.; Ganea, M.; Young, P.; and Ramavajjala, V. 2016. Smart reply: Automated response suggestion for email. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 955–964.

Kim, Y. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1746–1751.

Purcell, K., and Rainie, L. 2014. Email and the internet are the dominant technological tools in american workplaces. Technical report, Pew Research Center.

Qadir, A.; Gamon, M.; Pantel, P.; and Awadallah, A. H. 2016. Activity modeling in email. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1452–1462.

Sheng, Y.; Tata, S.; Wendt, J. B.; Xie, J.; Zhao, Q.; and Najork, M. 2018. Anatomy of a privacy-safe large-scale information extraction system over email. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 734–743.

Van Gysel, C.; Mitra, B.; Venanzi, M.; Rosemarin, R.; Kukla, G.; Grudzien, P.; and Cancedda, N. 2017. Reply with: Proactive recommendation of email attachments. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 327–336.

Zhao, Q.; Bennett, P. N.; Fourney, A.; Thompson, A. L.; Williams, S.; Troy, A. D.; and Dumais, S. T. 2018. Calendar-aware proactive email recommendation. In *The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval*, 655–664.