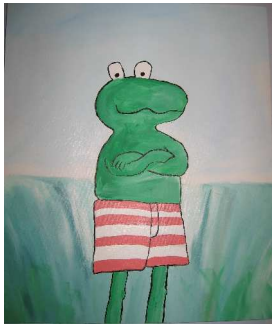
The Intelligent E-Mail Sorter

Eric McCreath
Department of Computer Science
Faculty of Engineering and Information Technology
College of Engineering and Computer Science
The Australian National University
Australia

John, Hugh, Frog and a phone call about email!!



Introduction

- Email has become our habitat.
  - Ducheneaut/Bellotti 2001 ACM interactions
- Web based approaches are very common.
  - Hotmail - 200 million active accounts - 75million emails a day.(http://advertising.msn.co.uk)
- A large number of different uses:
  - conversation, task manager, document delivery, contact manager....
- Email is accessible, universal, and robust...
- Use of instant messaging is growing.
- In all this we see a converging of technologies.

People/History

- 6 conference papers and 1 journal article.
- Started with collaboration with Judy Kay.
- Research grant $20K.
- Employ Liz Crawford(currently doing a PhD at CMU)

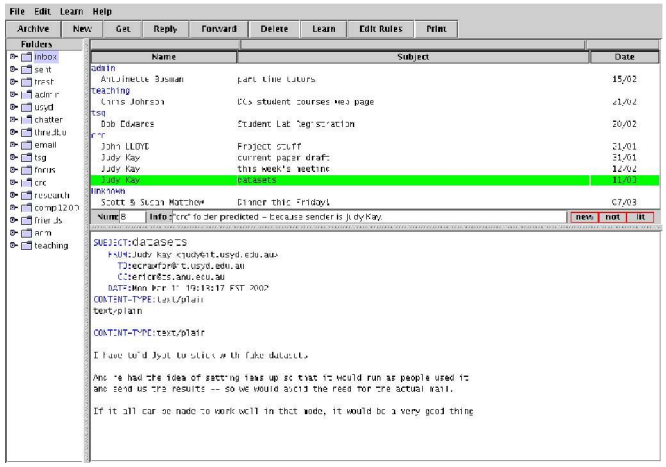
Overview

- User Interface
- Learning Approach
- Evaluation Metric
- The new email manager.
- Some new directions:
  - Learning importance.
  - Combining email with a Todo application.
  - Organizing different types of information using the same categories.
  - Exploring different user interfaces.

User Interface

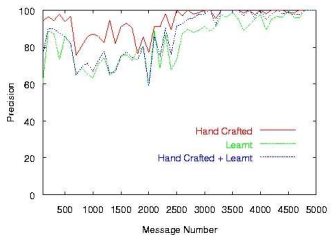
- Different people organize their email in very different ways.
- Q: Should the interface move messages into the folders? A: YES.
- Q: Should the interface leave the messages in the inbox? A: YES
- The interface should be:
  - Scrutable,
  - Modifiable,
  - Efficient, and
  - Predictable.

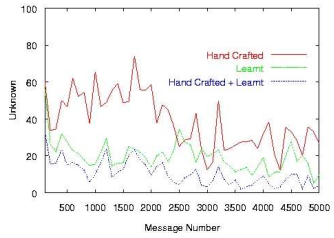
User Interface



Experimental Results

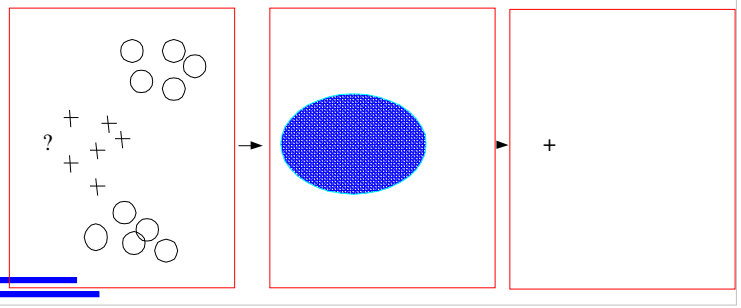
- Comparing handcrafted and learnt rules.
- Single user over 3 months, 5100 messages, 70 hand crafted messages, 21 different folders.





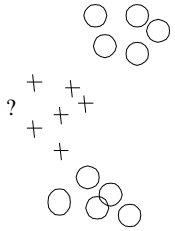
Learning Approach

- Many learners form explicit generalization of the training date



Learning Approach

- Instance based learning approaches simply collect the training examples. No explicit generalization is constructed from the data. Rather when a prediction is needed on a new instance, the distance from the instance to all of the training examples is measured.



Learning Approach

- Our approach generates an explicit hypothesis that directs the use of an instance based approach.
- These rules are argmented with hand crafted rules

filter(M,F) <- lastXsender_placements(F,M,5), F != “ml”.
filter(M,F) <- contains_same(F,M,subject), F != “colleagues”.

More Experimental Results

- 5 users
- Six approaches compared:
  - Keyword,
  - TF-IDF,
  - Dtree,
  - Naïve Bayes, and
  - Composite Rule.

User	# Messages	# Folders
1	526	7
2	949	35
3	869	12
4	429	9
5	15000	24

Measuring Success

- How do you measure the success of such a system?
  - accuracy,
  - precision,
  - recall,
  - f1, or
  - other ???

The metric used to evaluate the performance of a system is also reflected back into how the system selects competing hypotheses.

Experimental Results

Percentage of Unknown Classifications

User	Sender	KeyWord	TF-IDF	Dtree	Bayes	Rule
1	47.5	0.0	0.0	0.0	28.0	26.3
2	30.0	55.7	0.0	-	46.5	56.9
3	12.5	28.8	0.0	0.0	38.8	41.0
4	44.9	5.1	0.0	0.0	54.4	96.9
5	31.7	31.6	0.0	-	68.2	29.1

Experimental Results

Recall

User	Sender	KeyWord	TF-IDF	Dtree	Bayes	Rule
1	47.5	69.4	62.7	68.2	56.3	64.7
2	41.3	15.6	24.2	-	28.9	29.1
3	70.4	43.7	46.5	61.4	39.3	49.5
4	28.5	41.3	29.2	45.9	19.7	2.3
5	46.3	49.0	53.9	-	12.2	54.4

Precision

User	Sender	KeyWord	TF-IDF	Dtree	Bayes	Rule
1	91.0	69.4	62.7	68.2	78.2	87.8
2	59.0	35.2	24.2	-	54.0	67.6
3	80.4	61.3	46.5	61.4	64.2	83.9
4	51.6	43.5	29.2	45.9	43.3	75.0
5	67.8	71.8	53.9	-	38.2	76.8

Experimental Results

F1 Measure (Harmonic mean of recall and precision)

User	Sender	KeyWord	TF-IDF	Dtree	Bayes	Rule
1	62.4	67.1	62.7	68.2	65.5	74.5
2	48.6	21.6	24.2	-	37.7	40.7
3	75.1	51.0	46.5	61.4	48.8	62.3
4	36.7	42.4	29.2	45.9	27.1	4.5
5	55.0	58.2	53.9	-	18.5	63.7

Future Direction

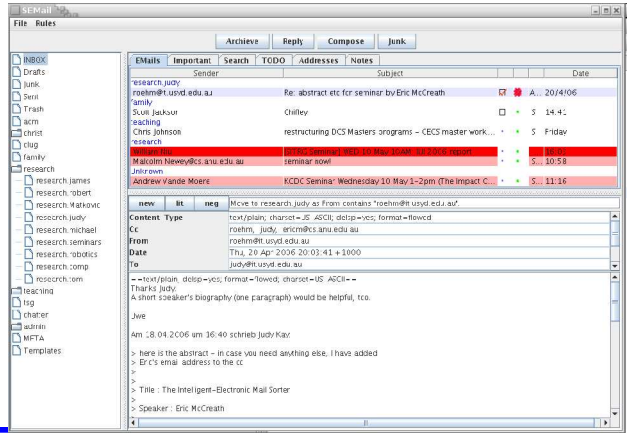
- Process of writing a completely new version of IEMS. (Called SE-MAIL or MyMail or ??)
- Managing tasks using email.
- Learning the importance of incoming messages.
- Moving away from the bag of words representation for text.

The New Email Manager

- Uses IMAP as a backing store for messages and meta data.
- Written in Java uses the JavaMail API
- 51 Class, ~5000 lines of code and growing!
- Uses both hand crafted and learnt rules.(basic sender learner only implemented.)

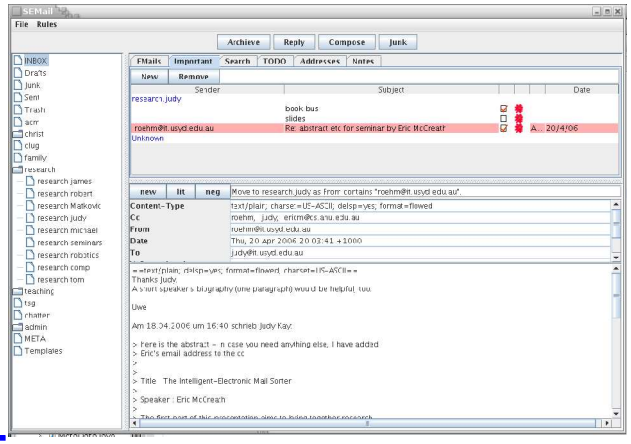
View of the inbox

- Note, very similar to iems except the tabs give the user a different view on the information.



View of the items marked important

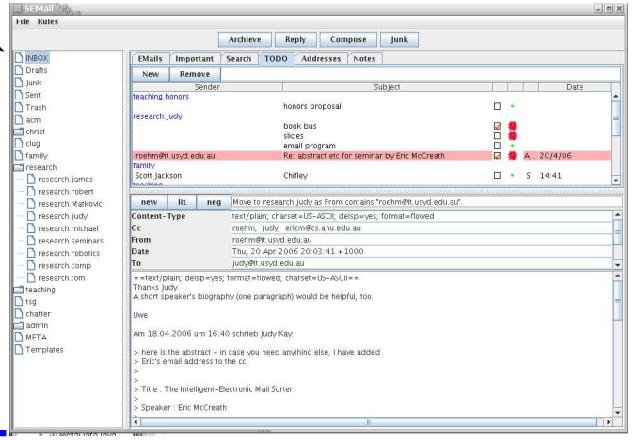
- The important tab provides an orthogonal view on this information.



The Todo List

- People use the flagging of email to manage their work, however, this approach is often limited.

The 'inbox' could be renamed the 'activebox'.



A Unified View of Desktop Artifacts

- Categorization of your desktop artifacts provides a way of organizing your work.
- These categories can go across different types of desktop artifacts including : email, todos, notes, addresses, and even files.
- By unifying interface in which a user interacts with these 'blobs' of information it is hoped one can reduce:
  - the amount of interaction required to achieve tasks, and
  - the cognitive load on the user.

Files

- We have lots of files. Lots and lots of files. It can be often hard to track down the file you are looking for. (I have 115,245 files, 25 files a day over the last 13 years)
- Often emails stores act as a users file system. Why not merge them and do away with our traditional view on files?
- Consider the following scenario:

  You are sent an email with an attached open office document that you need to modify and send back.

Conclusion

- Change is on the way.