Catching Missing Attachments: The Use of Alternative Presentation Techniques to Increase the Usability of Low Precision Systems

Crease, Murray; Lovesey, A.

National Research Council Canada
Conseil national de recherches Canada

Institute for Information Technology
Institut de technologie de l'information

NRC · CNRC

Catching Missing Attachments: The Use of Alternative Presentation Techniques to Increase the Usability of Low Precision Systems *

Crease, M., and Lovesey, A.
May 2005

* published at the 16th International Information Resources Management Association Conference (IRMA). San Diego, California, USA. May 15, 2005. NRC 47451.

Canada

Catching Missing Attachments: The Use of Alternative Presentation Techniques to Increase the Usability of Low Precision Systems

Murray Crease,
NRC-IIT e-Business,
46 Dineen Drive,
Fredericton,
NB, Canada, E3B 9W4
E-mail: murray.crease@nrc-cnrc.gc.ca
Tel: +1 506 444 0496

Anthony Lovesey,
Faculty of Computer Science,
University of New Brunswick,
P.O. Box 4400, 540 Windsor Street, Gillin Hall, E126,
Fredericton, NB, Canada, E3B 5A3.
I8MEU@unb.ca

ABSTRACT

Sending an e-mail without an intended attachment can be embarrassing and frustrating for both the sender and the recipient. Despite this, there is no widely used technique for detecting such e-mails before they are sent. This paper outlines why emails are sent without attachments; the problems associated with the detection and user alert of such e-mails; and how our proposed solution hopes to overcome these problems.

INTRODUCTION

Forgetting to attach an attachment to an outgoing e-mail message typically occurs because the sender reaches closure when the message text is complete but before the attachment has been attached. One possible solution would be for the sender to add the attachment when the text referring to it is written. This, however, would require the sender to interrupt their flow of thought when writing the message. A more preferable solution would be to alert the user after the message has been sent but before it has left the outbox. This raises two questions: how to detect the non-existence of an attachment and how best to alert the sender.

Determining whether an email should have an attachment is a non-trivial task. By parsing a message's body for keywords it is possible to determine the likelihood that the sender intended to include an attachment. By limiting the number of keywords to terms which are almost certain to refer to an attachment (e.g. variants of 'attached' and 'enclosed') it is possible to build a system which has high precision – i.e. when it alerts the user of it will almost always be correct. Such a system will, however, have low recall – i.e. it will not alert the user about many messages that should have attachments but do not. The recall of the system can be improved by extending the list of keywords to include terms such as "here is" but this will have a detrimental effect on the precision of the system.

The means by which users can be alerted to missing attachments range from the demanding (e.g. popup dialog box) to the subtle (e.g. icon in the system tray). A demanding alert is likely to be most useful with high precision, low recall systems as the alert is likely to be correct and so will not unduly annoy the user. This may mean, however, that users will not be alerted to as many messages as they should be. The more subtle alerts are better suited to low precision, high recall systems as users will be able to ignore the often incorrect alerts. There is a risk, however, that a user could miss an alert for a message that would have been captured by a high precision system that would have given a more demanding alert.

A survey of existing systems designed to alert users to missing email attachments (e.g. http://www.attachmentglue.com/) showed that they were all high precision, low recall systems. Whilst a partial solution to the problem, the low recall of these systems minimises their effectiveness.

PROPOSED SOLUTION

Our solution is to build a system which has a high recall (thus minimising the number of messages not alerted that should have been), provides demanding feedback for messages that almost certainly should have an attachment while reducing the annoyance of incorrect alerts. This implies two things about the system: it has a range of techniques for managing missing attachments available; and it knows how appropriate the different techniques are.

Rule Design and Calibration

The initial set of rules were created by manually analysing messages which had been sent with attachments. These rules were of two general types: they parsed the language of the message or they inspected the overall attributes of the message. An example of the former is a rule that looks for the keyword “attach”. An example of the latter is a rule that looks for short messages. An initial implementation of an application to check for missing attachments allowed users to calibrate these rules over their previously sent items (Figure 1). This was used to guide improvement in the design of the rules.

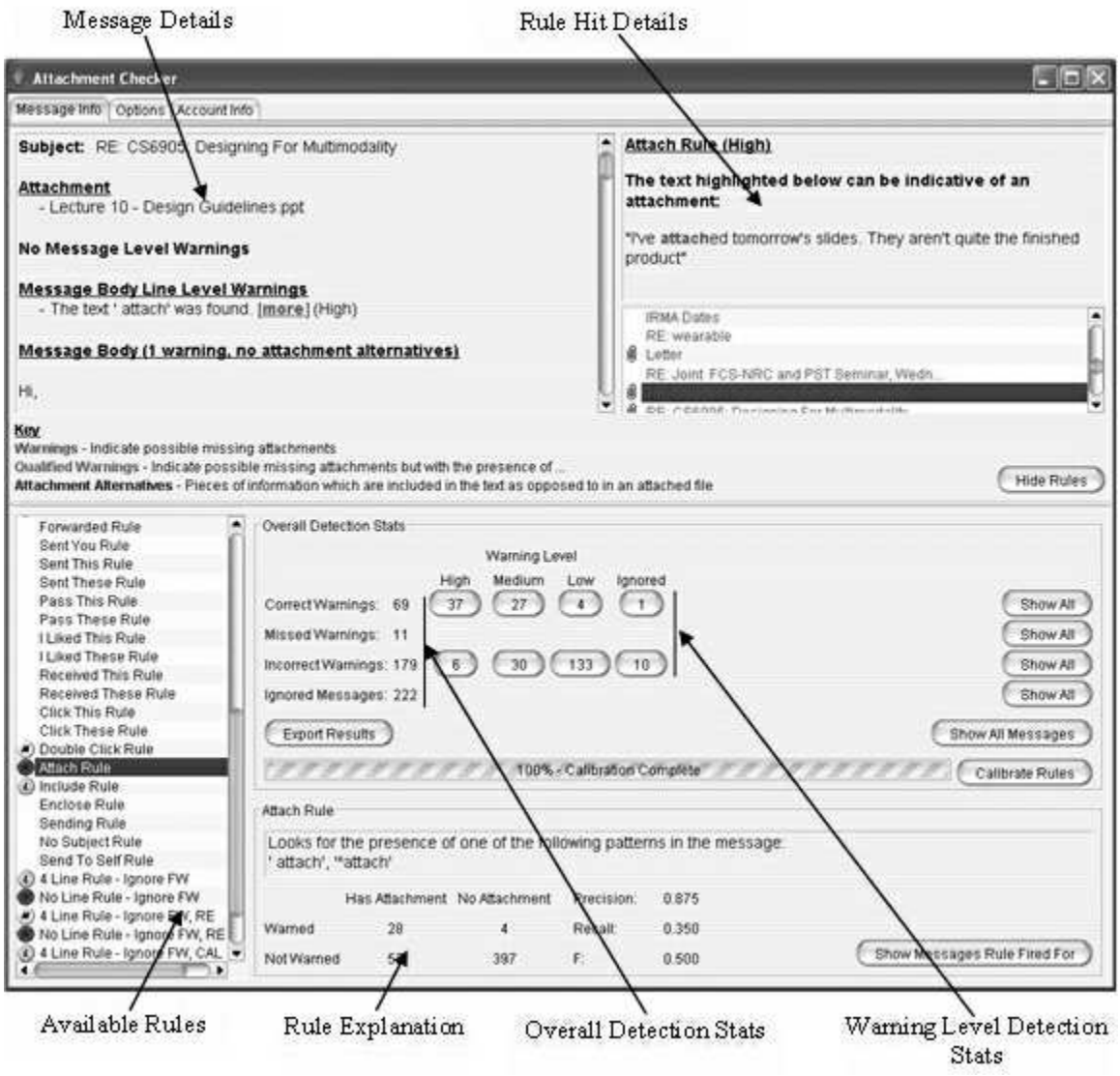

Figure 1 – Rule Calibration Interface

There are two main parts to the interface. The upper portion provides the users with an explanation of what rules fired for a particular message and why. By providing the users with explanations regarding why certain rules fired for a particular message it is hoped that the users will trust the system more as discussed in [3]. The lower portion provides feedback regarding the effectiveness of the different rules available. This part of the interface is of most use when designing the rules and can be hidden if not required. The rule explanation panel shows how often a rule fired correctly and incorrectly. These values are used to calculate the precision of the rule which is used as the confidence the system has in the rule. A rule’s recall is not considered as it is the recall of the system as a whole that is important. Four levels of confidence are used: high (precision>=0.75); medium (0.75>precision>=0.4); low (0.4>precision>=0.1); and none (precision<0.1). The determination of whether a rule fired correctly or incorrectly was made on the basis of whether the message had an attachment. For the ratings to be accurate the assumption that the majority of attachments were correctly sent is made.

Using this system the rule set was improved and the concept of an attachment alternative was developed allowing many false positives to be negated. An attachment alternative is a piece of information which has been incorporated into the body of the message as opposed to being attached as a separate file. Examples of attachment alternatives are URLs and colons. The final set of rules included:

- The presence of keyword(s) such as 'attach' or 'enclose'.
- The presence of keyword(s) such as 'this is' or 'here is' with no attachment alternative within a specified number of lines.
- No subject in the message.
- A message of fewer than a specified number of lines.
- A message sent to self.

In addition to the development of further rules the way the messages were parsed was improved. Initial attempts parsed the whole message text meaning that signatures and the text of messages which were being replied to were parsed. Clearly, this had a negative effect on the system's accuracy. This problem was reduced by allowing users to specify a piece of text which signified the start of their signature. No lines beyond this point were considered. This greatly improved the accuracy but is not a perfect solution as some users do not use signatures while others put their replies at the most appropriate point in the original message body. A potential solution would be to use more complex parsing as described by Carvalho & Cohen although this is only effective on plain-text messages [1].

An analysis of three sets of data generated by users who helped with the development of the rules showed that of the 3037 messages parsed, 399 had attachments. Of these, the system would have not have alerted the user 32 times (recall = 0.92). There were 1161 incorrect warnings (precision = 0.26). This would indicate that the system is a high recall, low precision system. A detailed analysis of the 32 missed warnings was not possible for privacy considerations but it was possible to check the 11 warnings not given when the first author's email was parsed.

Three of the messages talked indirectly about the file(s) attached but did not refer to them directly. Two of the messages contained links which were viewed as attachment alternatives and so negated warnings that had fired. Four of the messages contained attachments which were being forwarded and could not have been forgotten. One message contained inline images which are treated as attachments while the final message contained a typo so the word 'attached' – which had been spelt 'attacked' – was not recognised! A twelfth message was counted as being warned but the warning was of such low confidence it was ignored. The attachments to this message were in the original message being forwarded and as such could not have been forgotten.

Of the 367 correct warnings, 275 were given with high confidence (74.9%). Conversely, of the 1167 incorrect warnings only 52 (4.46%) were given with high confidence. Again, a detailed analysis of all these messages was not possible but of the 6 that could be analysed it was found that 2 were due to no signature being present meaning the message being responded to was being parsed. In one instance it was found that a restrained version of a rule had been negated but a less restricted version that still had a high confidence value had fired. The three remaining instances occurred when the keyword 'attach' was used to refer to the chain of previous messages being sent.

What this analysis has shown is that while the system is effective at detecting missing attachments – and it will be possible to improve the accuracy by improving the rules and the way they are applied to the messages - it will never be possible to get 100% accuracy.

PRESENTATION OPTIONS AND EVALUATION

While it will never be possible to build a system with 100% accuracy it is hoped that by using different notification techniques it may be possible to reach 100% recall while minimising annoyance to the user. The previous section described the four levels of confidence the system used when alerting the users. Making the alerts increasingly demanding as the confidence level increases the system will ensure that the user will not miss the majority of warnings. Conversely, by alerting the user for every message regardless of confidence level it will be possible to increase the recall to 100%. Clearly, these alerts will have to be non-demanding to minimise user annoyance. The proposal is to use sound for these non-demanding alerts. Previous work has shown that users are capable of monitoring low-level background tasks using audio feedback without finding these sounds annoying [2].

The system will be evaluated by giving it to users and recording its usage. Different versions of the system will be used to determine the effectiveness of different designs. Some users will get different alerts for the different warning levels, some will get different alerts for all messages including those with no warnings while others will get the same, non-demanding alert for all messages. The metrics for comparison will be how many 'forgotten' attachments are added before the message is sent and how annoying the users' find the system's feedback.

CONCLUSIONS

This paper outlines the problems surrounding the detection of emails which should have attachments but do not and our proposed solution to this problem. By employing different alert techniques according to how 'confident' the system is about its warning we hope to increase the number of missing attachments detected without annoying the user with attention grabbing, incorrect alerts.

ACKNOWLEDGEMENTS

The authors would like to thank Anna MacLachlan for her much needed assistance in the field of natural language parsing without which none of this work would have been possible.

REFERENCES

[1] V. R. Carvalho and W. W. Cohen, "Learning to Extract Signatures and Reply Lines from Email," in *Proceedings of the First Conference on Email and Anti-Spam (CEAS)*. Mountain View, CA, 2004.

[2] M. Crease and S. Brewster, "Scope For Progress: Monitoring Background Tasks With Sound," in *Human-Computer Interaction, Interact'99*, vol. II, S. Brewster, A. Cawsey, and G. Cockton, Eds. Edinburgh, UK: IFIP, 1999, pp. 19-20.

[3] T. Roth-Berghofer, "Explanation and Case-Based Reasoning: Foundational Issues," in *European Conference on Case-Based Reasoning ECCBR 2004*. Madrid, Spain, 2004.