Improving Automated Email Tagging with Implicit Feedback

Mohammad S. Sorower
Michael Slater
Thomas G. Dietterich
• Motivation
• The Email Predictor
  • EP2 Instrumentation
• Algorithms
  • Baseline Algorithms
  • Implicit Feedback Algorithms
• The Lab-controlled User Study
  • Data set of Tagged Email Messages
  • Post-study Simulation
• Results
• Summary
Online Email Tagging:
- user receives an email message
- system predicts tags for the message
- the email user interface shows the predicted tags
- if a predicted tag is wrong:
  user may correct the tag
  (if so, the system receives training)
- if a predicted tag is right:
  user does not have to do anything
  (the system never receives training)
• **Challenges:**
  - learning algorithm never receives confirmation that its predictions are correct
  - the learning algorithm would benefit from positive feedback.

• **Survival Curve:**
  - the more time a user spends on a message, the more likely that the user will notice tag errors and correct them.

• **Implicit Feedback!**
• Implicit Feedback Features:
  - message was opened and read in either the Outlook Explorer or the Outlook Inspector
  - user added or removed a tag on the message
  - user added or removed a flag from the message
  - user moved the message to a folder
  - user copied, replied, forwarded, or printed a message
  - user saved an attachment from the message
Baseline Algorithms:

• No Implicit Feedback (NoIF)
  - never creates implicit feedback training examples
  - only trains on user corrections
  - standard behavior of EP (Lower-bound on performance)

• Online
  - ignores all implicit feedback events
  - after making a prediction, creates training examples with the ground truth tags
  - provides perfect feedback to EP (Upper-bound on performance)
Implicit Feedback Algorithms

• **Simple Implicit Feedback (SIF)**
  - when the user changes any tag immediately treats all remaining tags as correct

• **Implicit Feedback without SIF (IFwoSIF)**
  - maintains a count of the total number of implicit feedback events
  - treats tag changes just like all other implicit feedback events
  - when this count exceeds a specified threshold, then it creates the implicit feedback training examples

• **Implicit Feedback with SIF (IFwSIF)**
  - combines IFwoSIF and SIF
THE USER STUDY

• Participants
  - 15 participants (1 dropped out)
  - only adult email users who receive 20 or more emails per day, regularly use tags, categories, labels, or folders

• The Study Data
  - an email data set containing a total of 330 messages created from a variety of web sources
  - Train60, Test270

• The Study Sessions
  - three two-hour sessions on three separate days
  - 1 hour practice, 5 hours performing study tasks (reading emails, correct tags if necessary, follow instructions in the message)
  - user ground truth collected at the end
- Email life of a knowledge worker—a student in this case
  - a total of 330 messages
  - average number of tags per email message = 1.24
  - messages with information, requests to send file, search online, save attachment, forward message etc.

<table>
<thead>
<tr>
<th>Tags</th>
<th>%messages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Economics</td>
<td>15</td>
</tr>
<tr>
<td>Entertainment</td>
<td>18</td>
</tr>
<tr>
<td>Gardening</td>
<td>19</td>
</tr>
<tr>
<td>Health</td>
<td>23</td>
</tr>
<tr>
<td>Math</td>
<td>17</td>
</tr>
<tr>
<td>Meeting/Event</td>
<td>31</td>
</tr>
</tbody>
</table>
• The participants did not provide very much explicit feedback
  - mean percentage of messages for which they corrected tags was 16.3%

• Solution: combine the observed implicit feedback events with simulated explicit feedback
• Algorithm \textbf{SampleEF (user, TargetEF)}:

  Estimate the (fitted) probability, \( P(EF \mid totalIF) \)

  FOR every message, compute \( p_i = P(EF(i) \mid totalIF(i)) \)

  Compute the observed level of EF (\( obs\_EF \)) in ‘user’ data

  IF \( obs\_EF > TargetEF \):
    \begin{itemize}
      \item DO: delete EF from the message (that has EF) with the smallest \( p_i \)
      \item UNTIL \( obs\_EF = TargetEF \)
    \end{itemize}

  ELSE:
    \begin{itemize}
      \item DO: add EF to the message (that has no EF) with the largest \( p_i \)
      \item UNTIL \( obs\_EF = TargetEF \)
    \end{itemize}
• Implicit feedback captured during the study sessions of one participant.
• The first session ends after message 66, and the second session ends after message 168.
• Implicit Feedback Threshold Selection

- a threshold exists such that the loss in accuracy of the resulting incorrect training is out-weighed by the gain of the resulting correct training examples
• **Cumulative Mistakes**
  - Plotted as a function of number of examples seen from the test data
  - SIF and IFwSIF algorithms have largely eliminated the gap between NoIF and Online
• SIF produces the predominant share of the training examples
• Additional examples added by implicit feedback have a substantial effect on further reducing prediction errors
• IFwSIF receives 64% more training than NoIF, and 14% more training than SIF
• Quality of the implicitly-confirmed training examples
  - at TargetEF 0.20, only 64% of the confirmed messages have correct tags
  - at TargetEF 0.80, only 74% of the confirmed messages have correct tags

• Although implicit feedback is noisy, on balance the classifiers still benefited!
• Automated tagging of email with user-defined tags is possible
• By instrumenting the UI, we can detect “implicit positive feedback” with reasonable accuracy
• Incorporating implicit feedback into the classifier(s) improves the performance of the email predictor
Thank you
SUMMARY

• Highly-accurate tagging of email with user-defined tags is possible
• By instrumenting the UI, we can detect “implicit positive feedback” with reasonable accuracy
• Incorporating implicit feedback into the classifier(s) improves the performance of the email predictor