Filtering for Personal Web Information Agents

Gabriel L. Somlo
and
Adele E. Howe

Computer Science Dept.
Colorado State University
Objectives

- Incorporate text filtering into personal Web information agents

Desired properties of agent-embedded filtering:
- Avoid negative feedback
- Learn quickly, with limited training
- Incremental learning (avoid storing training instances)
Filtering Algorithms and Parameters

- TF-IDF representation + cosine similarity
  - 1- and 2-grams
  - stop-word pruning (y/n)
  - adaptive vs. min-max-ratio dissemination threshold

- Naïve Bayes Classifier
  - use terms from unlabeled documents?
  - how to avoid using labeled negatives?
Evaluation

Data Set: TREC Disk #5
- FBIS: 130,471 documents
- LATimes: 127,742 documents

Six topics:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>189</td>
<td>FBIS</td>
<td>584</td>
<td>695</td>
<td>62,595</td>
</tr>
<tr>
<td>301</td>
<td>FBIS</td>
<td>339</td>
<td>433</td>
<td>50,695</td>
</tr>
<tr>
<td>354</td>
<td>FBIS</td>
<td>175</td>
<td>715</td>
<td>64,424</td>
</tr>
<tr>
<td>374</td>
<td>LATimes</td>
<td>109</td>
<td>315</td>
<td>58,943</td>
</tr>
<tr>
<td>422</td>
<td>LATimes</td>
<td>98</td>
<td>840</td>
<td>70,875</td>
</tr>
<tr>
<td>426</td>
<td>LATimes</td>
<td>145</td>
<td>626</td>
<td>67,988</td>
</tr>
</tbody>
</table>

Metric: harmonic mean \( HM = \frac{2 \cdot \text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}} \)
TF-IDF Parameter Analysis

- 2-grams perform worse than single terms
- Stop-word removal does not improve HM
- Threshold learning: min-max outperforms adaptive learning

Comparison of Adaptive and MinMax across topics
TF-IDF: Analysis of Learning the Dissemination Threshold

- Static threshold = 0.1 comparable to best learned threshold – learning may be unnecessary!
Naïve Bayes Parameter Analysis

Because corpus is biased toward non-relevant documents...

Using terms from unlabeled documents is a terrible idea:

\[ p(t|C) = \frac{1}{n_{pos}(C) + n_{terms}} \]

\( n_{pos}(C) \equiv \# \text{ of term positions in class } C \)
\( n_{terms} \equiv \text{size of vocabulary} \)

\( n_{pos}(+) \ll n_{pos}(-) \)

Recall goes down as “positive” terms are discounted!
Avoiding explicit negative feedback:

1. Build initial classifier assuming all unlabeled docs are non-relevant
2. Classify all unlabeled docs using initial classifier, and sort by \( \Delta = [ p(doc|-) - p(doc|+) ] \)
3. From unlabeled docs with largest \( \Delta \), pick a number equal to that of the labeled relevant docs
4. Build a new classifier from labeled relevant and picked non-relevant docs

Equal numbers of relevant and non-relevant docs avoids problem shown on previous slide
Four Algorithms Compared

Harmonic Mean

Learning speed

Bayes performs better and learns faster!
Conclusions

Bayes: best performance, but requires negative feedback

ModBayes and MinMax not incremental

We may be able to bypass TF-IDF threshold learning and hard-code to 0.1

Bayes wins if we can convince users to supply negative feedback!