Detecting Action-Items in E-mail

Paul N. Bennett
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
pbennett+@cs.cmu.edu

Jaime Carbonell
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213
jgc+@cs.cmu.edu

Categories and Subject Descriptors: H.3.3 **[Information Storage and Retrieval]**: Information Search and Retrieval; I.2.6 **[Artificial Intelligence]**: Learning; I.5.4 **[Pattern Recognition]**: Applications

General Terms: Experimentation

Keywords: Text classification, e-mail, n-grams, SVMs

1. INTRODUCTION

E-mail users have a difficult time managing their inboxes in the face of mounting challenges. These include prioritizing e-mails from a variety of senders, filtering junk e-mail, and quickly taking action on items that demand the user's attention. Automated *action-item detection* targets the third of these problems by detecting e-mails which *require* an action or response, and within those e-mails, highlighting the specific text that indicates the request.

In contrast to action-item detection which aims at locating exactly where the action item requests are contained within the email body, typical text categorization (TC) merely assigns a topic label to the entire message — whether that label corresponds to an e-mail folder or an indexing vocabulary [8]. In further contrast to TC, action-item detection attempts to recover the sender's intent, *i.e.* whether she means to elicit response or action on the part of the receiver. Whereas TC by topic [5, 6, 9], TDT [1], and even genre-classification [7] work well using just individual words as features, we believe that action-item detection is the first TC task where we clearly must move beyond bag-of-words — albeit not too far, as bag-of-n-grams seems to suffice.

> The current schedule for the visit by the GRTY group looks like this:
> + 10:30 a.m. Individual Meetings (Break for Lunch)
> + 2:00 p.m. Sales Pitch
> *To prepare, I need each of your parts for the presentation by Wednesday.*
> Keep up the good work!
> –Henry

Figure 1: An E-mail with emphasized Action-Item

2. RELATED WORK

While Cohen et al. [3] describe an ontology of "speech acts" that subsumes action-items, their methods only make use of human judgments at the document-level. In contrast, we consider whether accuracy can be increased by using finer-grained human judgments that mark the specific sentences and phrases of interest. Corston-Oliver et al. [4] consider detecting items in e-mail for a "To-Do List" using a single classifier; however, they do not explicitly compare what if any benefits finer-grained judgments offer.

In contrast to previous work, we focus on the benefits that finer-grained (more costly) sentence-level human judgments offer over coarse-grained document-level judgments. Additionally, we consider multiple standard text classification approaches and analyze the differences of a document-level vs. a sentence-level approach.

3. PROBLEM DEFINITION & APPROACH

To provide better end-user benefit, a system would both detect an action-item document and indicate the specific sentences which contain the action-items. Therefore, there are three basic problems: *document detection*, *document ranking*, and *sentence detection*.

The labeled data can come in one of two forms: a *document-labeling* provides a yes/no label for each document as to whether it contains an action-item; a *phrase-labeling* provides a yes label for each action-item. Obviously, it is straightforward to generate a document-labeling consistent with a phrase-labeling by labeling a document "yes" if and only if it contains at least one "yes" phrase.

To train classifiers, we can take one of two approaches related to the form of the labeled data. The *document-level* view treats each e-mail as a learning instance with a class-label. In the *sentence-level* view, after automatic sentence-segmentation, each sentence is treated as a learning instance with an associated class-label.

Representation and Implementation Overview

For this study, only the body of each e-mail message was used. We compare a standard bag-of-words or *unigram* representation to a bag of n-grams. We also retain sentence-ending punctuation as a token. For the bag of n-grams, *beginning-of-sentence* and *end-of-sentence* markers are included. Finally, for the sentence-level classifiers using n-grams, we also code the position of the sentence relative to the e-mail in octiles. For feature selection, we use chi-squared and choose the number of features that yield the optimal document-level F1 for that classifier during nested cross-validation.

In order to compare the document-level to the sentence-level approach, we compare predictions at the document-level. We use the RASP parser [2] to automatically segment the text of the e-mail, and then treat any sentence that contains at least 30% of a marked action-item segment as an action-item.

We applied a variety of standard TC algorithms: k-NN (s-cut), multinomial naïve Bayes, and SVMs. Once a sentence-level classifier makes a prediction for each sentence, we combine these predictions into a document-level prediction and a document-level score. We use the simple policy of predicting positive when any of the sentences is predicted positive. For ranking, the document score is the length normalized sum of the sentence scores above threshold.

SIGIR'05, August 15–19, 2005, Salvador, Brazil.
ACM 1-59593-034-5/05/0008.

	Classifiers	Document Unigram	Document Ngram	Sentence Unigram	Sentence Ngram
F1	kNN	0.6670	0.7108	0.7615	**0.7790**
	naïve Bayes	0.6572	0.6484	0.7715	**0.7777**
	SVM	0.6904	0.7428	0.7282	**0.7682**
Accuracy	kNN	0.7029	0.7486	0.7972	**0.8092**
	naïve Bayes	0.6074	0.5816	0.7863	**0.8145**
	SVM	0.7595	0.7904	0.7958	**0.8173**

Table 1: Average Document-Detection performance with best performance per classifier in bold.

To compare the performance of the classification methods, we use F1 and accuracy. We perform standard 10-fold cross-validation on the set of documents. For the sentence-level approach, all sentences in a document are either entirely in the training set or entirely in the test set for each fold. For significance tests, we use a two-tailed t-test to compare the values obtained during each cross-validation fold with a p-value of 0.05.

Our corpus consists of e-mails obtained from volunteers at our university. After eliminating duplicate e-mails, the corpus contains 744 e-mail messages. To balance cognitive load in the user studies (omitted here) and prevent chronological taints of cross-validation, the studies reported here are performed with a version of the corpus after quoted material is removed by hand.

Two human annotators labeled all the messages and identified each segment of the e-mail which contained action-items. At the document-level, the kappa statistic for inter-annotator agreement is 0.85 and 0.82 at the sentence-level. After reconciling the judgments there are 328 e-mails containing action-items.

4. RESULTS & DISCUSSION

The primary hypothesis is that n-grams are critical for this task at the document-level. Examining Table 1, they improve performance for every classifier except naïve Bayes. Naïve Bayes is hurt by the n-gram representation because of excessive double-counting. The significance results for comparing n-gram improvement with a fixed classifier are summarized on the left of Table 2 (F1 significance in bold, Accuracy with a †). N-grams show significant improvement at the document-level and the best performance overall.

	Doc Winner	Sent Winner	F1	Acc
kNN	**Ngram**	Ngram	**Sentence**	**Sentence**
naïve Bayes	Unigram	Ngram	**Sentence**	**Sentence**
SVM	**Ngram**†	Ngram	Sentence	**Sentence**

Table 2: Summary for n-grams versus unigrams (*left*) and sentence-level classifiers vs. document-level classifiers (*right*).

As would be expected the difference between the *sentence-level* n-gram and unigram representations is small. This is because the window of text is so small that the sentence-level unigram representation implicitly picks up on the power of the n-grams. Therefore, the *finer-grained sentence-level judgments* allow a unigram representation to succeed but only when performed in a small window — behaving as an n-gram representation for all practical purposes.

Since the sentence-level classifier approach would not be possible without these costly fine-grained judgments, we now turn to the question of whether the sentence-level classifiers produce better document detection than a document-level classifier. In order to answer this question, we compare the best sentence-level result with the best document-level result on the right of Table 2 (significance in bold). The sentence-level approach wins entirely across the board — lacking significance only for F1 for SVMs. Sentence detection results are presented in Table 3 for completeness.

The effectiveness of sentence-level detection argues that labeling at the sentence-level provides significant value. Document-level detection using sentence-level classifiers works surprisingly well given most researchers' expectations of low recall for a single sentence. Our empirical analysis has demonstrated that n-grams are of key importance to making the most of document-level judgments. When finer-grained judgments are available, then a standard bag-of-words approach using a small (sentence) window size can produce results almost as good as the n-gram based approaches.

Acknowledgments

This material is based upon work supported by DARPA under Contract No. NBCHD030010. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect the views of DARPA or DOI-NBC.

We would also like to extend our sincerest thanks to Jill Lehman whose efforts in data collection were essential in constructing the corpus. Finally, we gratefully acknowledge Scott Fahlman for his encouragement and useful discussions on this topic.

	Accuracy		F1	
	Unigram	Ngram	Unigram	Ngram
kNN	0.9519	0.9536	0.6540	0.6686
naïve Bayes	0.9419	0.9550	0.6176	0.6676
SVM	0.9559	0.9579	0.6271	0.6672

Table 3: Performance for Sentence Detection

5. REFERENCES

[1] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang. Topic detection and tracking pilot study: Final report. In *the DARPA Broadcast News Workshop*, 1998.

[2] J. Carroll. High precision extraction of grammatical relations. In *COLING '02*, 2002.

[3] W. W. Cohen, V. R. Carvalho, and T. M. Mitchell. Learning to classify email into "speech acts". In *EMNLP '04*, 2004.

[4] S. Corston-Oliver, E. Ringger, M. Gamon, and R. Campbell. Task-focused summarization of email. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 2004.

[5] S. T. Dumais, J. Platt, D. Heckerman, and M. Sahami. Inductive learning algorithms and representations for text categorization. In *CIKM '98*, 1998.

[6] D. D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In *SIGIR '92*, 1992.

[7] Y. Liu, J. Carbonell, and R. Jin. A pairwise ensemble approach for accurate genre classification. In *ECML '03*, 2003.

[8] Y. Liu, R. Yan, R. Jin, and J. Carbonell. A comparison study of kernels for multi-label text classification using category association. In *ICML '04*, 2004.

[9] F. Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), March 2002.