Learning to Classify Email into Speech Acts

Abstract

It is often useful to classify email according to the intent of the sender (e.g., "propose a meeting", "deliver information"). We present experimental results in learning to classify email in this fashion, where each class corresponds to a verb-noun pair taken from a predefined ontology describing typical “email speech acts”. We demonstrate that, although this categorization problem is quite different from “topical” text classification, certain categories of messages can nonetheless be detected with high precision (above 80%) and reasonable recall (above 50%) using existing text-classification learning methods. This result suggests that useful task-tracking tools could be constructed based on automatic classification into this taxonomy.

1 Introduction

In this paper we discuss using machine learning methods to classify email according to the intent of the sender. In particular, we classify emails according to an ontology of verbs (e.g., propose, commit, deliver) and nouns (e.g., information, meeting, task), which jointly describe the “speech act” implicit in the email.

A method for accurate classification of email into such categories would have many potential benefits. For instance, it could be used to help an email user track the status of ongoing joint activities. Delegation and coordination of joint tasks is a time-consuming and error-prone activity, and the cost of errors is high: for people involved in many joint activities, it is not uncommon that commitments are forgotten, deadlines are missed, and opportunities are wasted because of a failure to properly track, delegate, and prioritize subtasks. The classification methods we consider could be used to partially automate this sort of activity tracking. A hypothetical example of an email assistant that works along these lines is shown in Figure 1.

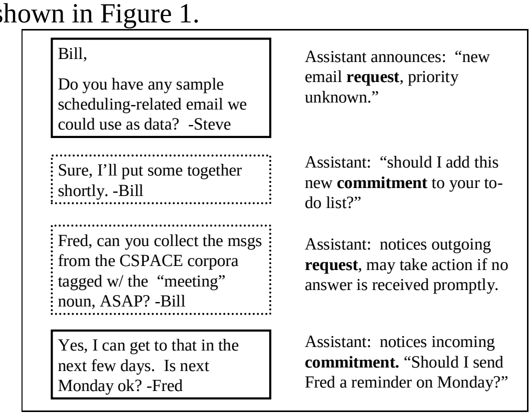

Figure 1 - Dialog with a hypothetical email assistant that can detect speech acts. Dashed boxes indicate outgoing messages.

2 Related Work

Our research builds on earlier work defining illocutionary points of speech acts (Searle, 1975), and relating such speech acts to email and workflow tracking (Winograd, 1987, Flores & Ludlow, 1980, Weigant et al, 2003). Winograd suggested that research explicating the speech-act based “language-action perspective” on human communication could be used to build more tools for coordinating joint activities. The Coordinator (Winograd, 1987) was one such system, in which users augmented email messages with additional annotations indicating intent.

While such systems have been useful in limited contexts, they have also been criticized as cumbersome: by forcing users to conform to a particular formal system, they constrain communication and make it less natural and flexible (Schoop, 2001); in short, users often prefer unstructured email interactions (Camino et al. 1998). We note that these difficulties are avoided if messages can be *automatically* annotated by intent.

Murakoshi et al. (1999) propose an annotation scheme for email broadly similar to ours called a "deliberation tree", and propose an algorithm for constructing deliberation trees automatically, but theirs was not quantitatively evaluated, and appears to be specific to Japanese-language emails.

Prior research on machine learning for text classification has primarily considered classification of documents by topic (Lewis, 1992; Yang, 1999), but also has addressed sentiment detection (Peng et al., 2002; Weibe et al., 2001) and authorship attribution (e.g., Argamon et al, 2003). There has been some previous use of machine learning to classify email messages (Cohen 1996; Sahami et al., 1998; Rennie, 2000; Segal & Kephart, 2000); however, to our knowledge, none of these systems have investigated learning methods for assigning speech acts to email. Instead, email is generally classified into folders (i.e., according to topic) or according to whether or not it is "spam". Learning systems have been previously used to automatically detect acts in conversational speech (e.g. Finke et al., 1998).

3 An Ontology of Email Acts

Our ontology of nouns and verbs covering some of the possible speech acts associated with emails is summarized in Figure 2. We assume that a single email message may contain multiple acts, and that each act is described by a verb-noun pair drawn from this ontology (e.g., "deliver data"). The underlined nodes in the figure indicate the nouns and verbs for which we have trained classifiers (as discussed in subsequent sections).

To define the noun and verb ontology of Figure 2, we first examined email from several corpora (including our own inboxes) to find regularities, and then performed a more detailed analysis of one corpus. The ontology was further refined in the process of labeling the corpora described below.

In refining this ontology, we adopted several principles. First, we believe that it is more important for the ontology to reflect observed linguistic behavior than to reflect any abstract view of the space of possible speech acts. As a consequence, the taxonomy of verbs contains concepts that are atomic linguistically, but combine several illocutionary points. (For example, the linguistic unit "let's do lunch" is both directive, as it requests the receiver, and commissive, as it implicitly commits the sender. In our taxonomy this is a single 'propose' act.) Also, acts which are abstractly possible but not observed in our data are not represented (for instance, declarations).

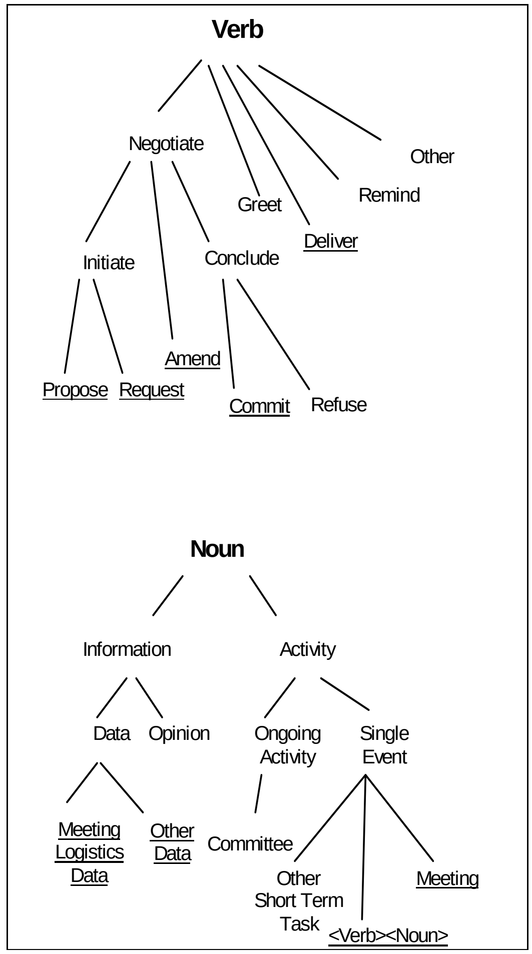

Figure 2 – Taxonomy

Second, we believe that the taxonomy must reflect common non-linguistic uses of email, such as the use of email as a mechanism to deliver files. We have grouped this with the linguistically similar speech act of delivering information.

Not all pairings of the verbs and nouns from Figure 2 make sense: for example, an email can 'deliver data,' but not 'deliver meeting'. In principle, we can take advantage of such constraints to reduce the error rate of predictions; however, to date we have not explored this. Finally, notice that while we associate just one noun and verb with each email speech act, many verbs in fact take multiple arguments. For example, an instance of the verb 'deliver' can involve a deliverer, a deliveree, and a delivered-item. Fortunately, some role fillers are often obvious from the email context (e.g., the email sender fills the deliverer role, the receiver fills the deliveree role, and hence the noun we wish to associate with 'deliver' is the noun filling the 'delivered-item' role).

The verbs in Figure 1 are defined as follows.

A *request* asks (or orders) the recipient to perform some activity. A question is also considered a request (for delivery of information).

A *propose* message proposes a joint activity, i.e., asks the recipient to perform some activity and commits the sender as well, provided the recipient agrees to the request. An email suggesting a joint meeting is a typical example.

An *amend* message amends an earlier proposal. Like a proposal, the message involves both a commitment and a request. However, while a proposal is associated with a new task, an amendment is a suggested modification of an already-proposed task.

A *commit* message commits the sender to some future course of action, or confirms the senders' intent to comply with some previously described course of action.

A d*eliver* message delivers something, e.g., some information, a PowerPoint presentation, the URL of a website, the answer to a question, a message sent "FYI", or an opinion.

The *refuse, greet,* and *remind* verbs occurred infrequently in our data, and will not be discussed further here.

The nouns in Figure 2 constitute possible objects for the email speech act verbs. The nouns fall into two broad categories.

Information nouns are associated with email speech acts described by the verbs *Deliver, Remind* and *Amend*, in which the email explicitly contains information. We also associate information nouns with the verb *Request,* where the email contains instead a description of the needed information (e.g., "Please send your birthdate." versus "My birthdate is …". The request act is actually for a 'deliver information' activity). Information includes data believed to be fact as well as opinions, and also attached data files

Activity nouns are generally associated with email speech acts described by the verbs *Propose, Request, Commit,* and *Refuse.* Activities include meetings, as well as longer term activities such as committee memberships.

Notice every email speech act is itself an activity. The <verb><noun> node in Figure 1 indicates that any email speech act can also serve as the noun associated with some other email speech act. For example, just as (deliver information) is a legitimate speech act, so is (commit (deliver information)), as well as (remind (commit (deliver information))), etc. Thus the above ontology could clearly be further elaborated to provide a compositional characterization of email speech acts involved in sequential email threads. However, here we consider only simple, atomic email speech acts.

4 Categorization Results

4.1 Corpora

The corpora used in the experiments consist of four different datasets: N01F3, N02F2, N03F2 and PW CALO. The first three datasets are subsets of a larger corpus, the CSpace email corpus, which contains approximately 15,000 email messages collected from a Management Game at Carnegie Mellon University. In this game, 277 MBA students, organized in approximately 50 teams of four to six members, ran simulated companies in different market scenarios over a 14-week period (Kraut et al.). N02F2, N01F3 and N03F2 are collections of all email from three different teams, and contain 351, 341 and 443 different email messages respectively.

The PW CALO corpus was produced by a simulated game at SRI. During four days, a group of six players assumed different work roles (e.g. project leader, finance manager, researcher, administrative assistant, etc). There are 222 different email messages from this corpus.

All messages were preprocessed by manually removing quoted material, attachments, and non-subject header information.

4.2 Inter-Annotator Agreement

Each message may be annotated with several labels, as it may contain more than one speech act. To evaluate inter-annotator agreement, we double-labeled N03F2 for the verbs *Deliver, Commit, Request, Amend,* and *Propose*, and the noun, *Meeting*, and computed the kappa statistic (Carletta, 1996) for each of these, defined as

$$\kappa = \frac{A - R}{1 - R}$$

where *A* is the empirical probability of agreement on a category, and *R* is the probability of agreement for two annotators that label documents at random (with the empirically observed frequency of each label). Hence kappa ranges from -1 to +1. The results in Table 1 show that agreement is good, but not perfect.

Email Act	**Kappa**
Meeting	0.82
Deliver	0.75
Commit	0.72
Request	0.81
Amend	0.83
Propose	0.72

Table 1 - Inter-Annotator Agreement on N03F2.

We also took doubly-annotated messages which had only a single verb label and constructed the 5-class confusion matrix for the two annotators shown in Table 2. Note kappa values are somewhat higher for the shorter one-act messages.

	Req	*Prop*	*Amd*	*Cmt*	*Dlv*	kappa
Req	55	0	0	0	0	0.97
Prop	1	11	0	0	1	0.77
Amd	0	1	15	0	0	0.87
Cmt	1	3	1	24	4	0.78
Dlv	1	0	2	3	135	0.91

Table 2 - Inter-annotator agreement on documents with only one category.

4.3 Learnability of Categories

Representation of documents. To assess the types of message features that are most important for prediction, we adopted Support Vector Machines (Joachims, 2001) as our baseline learning method, and a TFIDF-weighted bag-of-words as a baseline representation for messages. We then conducted a series of experiments with the N03F2 corpus only to explore the effect of different representations.

We noted that the most discriminating words for most of these categories were common words, not the low-to-intermediate frequency words that are most discriminative in topical classification. This suggested that the TFIDF weighting was inappropriate, but that a bigram representation might be more informative. Experiments showed that adding bigrams to an unweighted bag of words representation slightly improved performance, especially on *Deliver*. These results are shown in Table 4 on the rows marked “no tfidf” and “bigrams”. (The TFIDF-weighted SVM is shown in the row marked “baseline”, and the majority classifier in the row marked “default”.) Examination of messages suggested other possible improvements. Since much negotiation involves timing, we ran a hand-coded extractor for time and date expressions on the data, and extracted as features the number of time expressions in a message, and the words that occurred near a time (for instance, one such feature is “the word ‘before’ appears near a time”). These results appear in the row marked “times”. Similarly, we ran a part of speech (POS) tagger and added features for words appearing near a pronoun or proper noun (“personPhrases” in the table), and also added POS counts.

To derive a final representation for each category, we pooled all features that improved performance over “no tfidf” for that category. We then compared performance of these document representations to the original TFIDF bag of words baseline on the (unexamined) *N02F2* and *N01F3* corpora. As Table 3 shows, substantial improvement with respect to F1 and kappa was obtained by adding these additional features over the baseline representation.This is surprising given previous experiments with bigrams for topical text classification (Scott & Matwin, 1999) and sentiment detection (Pang et al., 2002).

Learning methods. In another experiment, we fixed the document representation to be unweighted word frequency counts and varied the learning algorithm. In most these experiments, we pooled all the data from the four corpora, a total of 1357 messages, and since the nouns and verbs are not mutually exclusive, we formulated the classification task as a set of several binary decision problems, one for each verb.

The learners used were the following. *VP* is an implementation of the voted perceptron algorithm (Freund & Schapire, 1999). *DT* is a simple decision tree learning system, which learns trees of depth at most five, and chooses splits to maximize the function $2\left(\sqrt{W_+^1 W_-^1} + \sqrt{W_+^0 W_-^0}\right)$ suggested by Schapire and Singer (1999) as an appropriate objective for "weak learners". *AB* is an implementation of the confidence-rated boosting method described by Singer and Schapire (1999), used to boost the *DT* algorithm 10 times. *SVM* is a support vector machine with a linear kernel (as used above).

Act		VP	AB	SVM	DT
Req	*Error*	0.25	0.22	0.23	**0.20**
(450/907)	*F1*	0.58	0.65	0.64	**0.69**
Prop	*Error*	0.11	0.12	**0.12**	0.10
(140/1217)	*F1*	0.19	0.26	**0.44**	0.13
Dlv	*Error*	**0.26**	0.28	0.27	0.30
(873/484)	*F1*	**0.80**	0.78	0.78	0.76
Cmt	*Error*	0.15	0.14	**0.17**	0.15
(208/1149)	*F1*	0.21	0.44	**0.47**	0.11
Directive	*Error*	0.25	0.23	0.23	**0.19**
(605/752)	*F1*	0.72	0.73	0.73	**0.78**
Commis-sive	*Error*	0.23	0.23	0.24	**0.22**
(993/364)	*F1*	0.84	0.84	0.83	**0.85**
Meet	*Error*	0.187	0.17	**0.14**	0.18
(345/1012)	*F1*	0.573	0.62	**0.72**	0.60
dData	*Error*	**0.11**	0.12	0.13	0.13
(223 / 1134)	*F1*	**0.58**	0.58	0.59	0.57

Table 3 – Learning on the complete corpus.

Table 3 reports the results on the most common verbs, using 5-fold cross-validation to assess accuracy. One surprise was that *DT* (which we had intended merely as a base learner for *AB*) works surprisingly well, particularly at detecting requests, while *AB* seldom improves much over *DT*. We conjecture that the bias towards large-margin classifiers that is followed by *SVM, AB,* and *VP* (and which has been so successful in topic-oriented text classification) may be less appropriate for this task, perhaps because positive and negative classes are *not* clearly separated (as suggested by substantial inter-annotator disagreement).

Further results are shown in Figure 3, which gives precision-recall curves for many of these classes. Note that the lowest recall level in these graphs corresponds to the precision of random guessing. These graphs indicate that high-precision predictions can be made for the top-level of the verb hierarchy, as well as verbs *Request* and *Deliver*, if one is willing to slightly reduce recall.

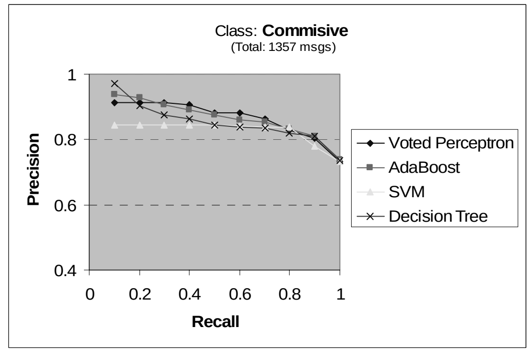

Figure 3 - Precision/Recall for Commissive act

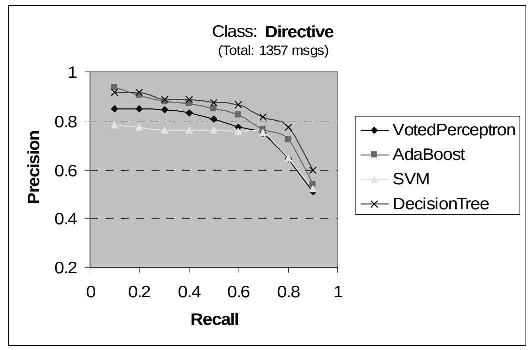

Figure 4 - Precision/Recall for Directive act

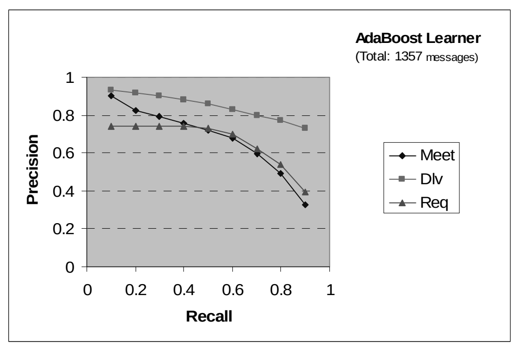

Figure 5 - Precision/Recall of 3 different classes using AdaBoost

Transferability. Another important question to investigate is the generality of these classifiers - that is, the range of corpora to which they can be accurately applied. Is it possible to train a single set of email-act classifiers that work for many users, or is it necessary to train individual classifiers for each user? To explore this issue we trained a DT classifier for Directive emails on the NF01F3 corpus, and tested it on the NF02F2 corpus; trained the same classifier on NF02F2 and tested it on NF01F3; and also performed a 5-fold

	Results on NF032								
	Commit			*Deliver*			*Directive*		
	Error	F1	kappa	Error	F1	Kappa	Error	F1	Kappa
Default	12.2	-	-	36.8	-	-	48.8	-	
Baseline SVM	10.8	25.0	22.0	29.1	49.8	31.3	23.9	75.2	52.1
no tfidf	13.1	47.3	39.8	31.8	58.4	32.7	24.4	74.6	51.7
+bigrams	11.1	46.1	40.2	25.5	66.1	45.6	22.1	76.0	55.6
+times	12.9	43.6	36.3	29.6	60.1	36.7	25.6	73.2	48.9
+POSTags	12.4	48.6	41.5	29.3	61.8	38.0	23.7	75.4	52.5
+personPhrases	12.9	41.2	34.1	29.3	61.1	37.5	25.1	73.4	49.7
	Results on NF01F3 and NF02F2								
Default	14.9			31.7			40.4		
Baseline SVM	14.2	10.1	9.2	22.0	56.3	42.7	24.3	66.1	47.7
All 'useful' features	14.7	42.0	33.9	22.1	64.0	48.0	20.4	73.3	56.9

Table 4 - Experiments with different document representations.

cross-validation experiment within each corpus. (NF02F2 and NF01F3 are for disjoint sets of users, but are approximately the same size.) We then performed the same experiment with VP for Deliver verbs and SVM for Commit verbs (in each case picking the top-performing learner with respect to F1). The results are shown in Table 5.

	Test Data			
DT/Directive	1f3		2f2	
Train Data	Error	F1	Error	F1
1f3	25.1	71.6	16.4	72.8
2f2	20.1	68.8	18.8	71.2
VP/Deliver				
1f3	30.1	55.1	21.1	56.1
2f2	35.0	25.4	21.1	35.7
SVM/Commit				
1f3	23.4	39.7	15.2	31.6
2f2	31.9	27.3	16.4	15.1

Table 5 - Transferability of classifiers

If learned classifiers were highly specific to a particular set of users, one would expect that the diagonal entries of these tables (the ones based on cross-validation within a corpus) would exhibit much better performance than the off-diagonal entries. In fact, no such pattern is shown. For *Directive* verbs, performance is similar across all table entries, and for *Deliver* and *Commit*, it seems to be somewhat better to train on NF01F3 regardless of the test set.

4.4 Future Directions

None of the algorithms or representations discussed above take into account the *context* of an email message, which intuitively is important in detecting implicit speech acts. A plausible notion of context is simply the preceding message in an email thread.

Exploiting this context is non-trivial for several reasons. Detecting threads is difficult; although email headers contain a "reply-to" field, users often use the "reply" mechanism to start what is intuitively a new thread. Also, since email is asynchronous, two or more users may reply simultaneously to a message, leading to a thread structure which is a tree, rather than a sequence. Finally, most sequential learning models assume a single category is assigned to each instance---e.g., (Ratnaparkhi, 1999)--whereas our scheme allows multiple categories.

Classification of emails according to our verb-noun ontology constitutes a special case of a general family of learning problems we might call *factored classification problems*, as the classes (email speech acts) are factored into two features (verbs and nouns) which jointly determine this class. From a practical viewpoint, a variety of real-world text classification problems can be naturally expressed as factored problems, and from a theoretical viewpoint, the additional structure may allow construction of new, more effective algorithms.

For example, the factored classes provide a more elaborate structure for generative probabilistic models, such as those assumed by Naïve

Bayes. For instance, in learning email acts, one might assume words were drawn from a mixture distribution with one mixture component produces words conditioned on the verb class factor, and a second mixture component generates words conditioned on the noun (see Blei et al (2003) for a related mixture model. Alternatively, models of the dependencies between the different factors (nouns and verbs) might also be used to improve classification accuracy, for instance by building into a classifier the knowledge that some nouns and verbs are incompatible.

The fact that an email can contain multiple email speech acts almost certainly makes learning more difficult: in fact, disagreement between *human* annotators is generally higher for longer messages. This problem could be addressed by more detailed annotation: rather than annotating each message with all the acts it contains, human annotators could label smaller message segments (say, sentences or paragraphs). An alternative to more detailed (and expensive) annotation would be to use learning algorithms that implicitly segment a message. As an example, another mixture model formulation might be used, in which each mixture component corresponds to a single verb category.

5 Concluding Remarks

We have presented an ontology of "email speech acts" that is designed to capture some important properties of a central use of email: negotiating and coordinating joint activities. Unlike previous attempts to combine speech act theory with email (Winograd, 1987; Flores and Ludlow, 1980), we propose a system which passively observes email and *automatically* classifies it by intention. This reduces the burden on the users of the system, and avoids sacrificing the flexibility and socially desirable aspects of informal, natural language communication.

This problem also raises a number of interesting research issues. This categorization problem is quite different from "topical" text classification, in terms of the types of features that are most informative. We show that part of speech tagging and entity extraction can be used to improve classification performance; we leave open the question of whether other types of linguistic analysis would be useful. Predicting implicit speech acts requires context, (which makes the prediction problem a sequential task) and the labels assigned to messages have non-trivial structure; we also leave open the question of whether these properties can be effectively exploited.

Notwithstanding these limitations in the scope of our study, our experiments show that many categories of messages can be detected, with high precision and moderate recall, using existing text-classification learning methods. This suggests that useful task-tracking tools could be constructed based on automatic classifiers—a potentially important practical application.

References

S. Argamon, M. Šarić and S. S. Stein. (2003). Style mining of electronic messages for multiple authorship discrimination: first results. Proceedings of the 9th ACM SIGKDD, Washington, D.C.

V. Bellotti, N. Ducheneaut, M. Howard and I. Smith. (2003). Taking email to task: the design and evaluation of a task management centered email tool. Proceedings of the Conference on Human Factors in Computing Systems, Ft. Lauderdale, Florida.

D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. (2003). Hierarchical topic models and the nested Chinese restaurant process. Advances in Neural Information Processing Systems, 16, MIT Press.

B. M. Camino, A. E. Millewski, D. R. Millen and T. M. Smith. (1998). Replying to email with structured responses. International Journal of Human-Computer Studies. Vol. 48, Issue 6, pp 763 – 776.

J. Carletta. (1996). Assessing Agreement on Classification Tasks: The Kappa Statistic. Computational Linguistics, Vol. 22, No. 2, pp 249-254.

W. W. Cohen. (1996). Learning Rules that Classify E-Mail. Proceedings of the 1996 AAAI Spring Symposium on Machine Learning and Information Access, Palo Alto, California.

M. Finke, M. Lapata, A. Lavie, L. Levin, L. MayfieldTomokiyo, T. Polzin, K. Ries, A. Waibel and K. Zechner. (1998). CLARITY: Inferring Discourse Structure from Speech. In Applying Machine Learning to Discourse Processing, AAAI'98.

F. Flores, and J.J. Ludlow. (1980). Doing and Speaking in the Office. In: G. Fick, H. Spraque Jr. (Eds.). Decision Support Systems: Issues and Challenges, Pergamon Press, New York, pp. 95-118.

Y. Freund and R. Schapire. (1999). Large Margin Classification using the Perceptron Algorithm. *Machine Learning* 37(3), 277—296.

T. Joachims. (2001). A Statistical Learning Model of Text Classification with Support Vector Machines. Proc. of the Conference on Research and Development in Information Retrieval (SIGIR), ACM, 2001.

R. E. Kraut, S. R. Fussell, F. J. Lerch, and J. A. Espinosa. (under review). Coordination in teams: evidence from a simulated management game. To appear in the Journal of Organizational Behavior.

D. D. Lewis. (1992). Representation and Learning in Information Retrieval. PhD Thesis, No. 91-93, Computer Science Dept., Univ of Mass at Amherst

A. McCallum, D. Freitag and F. Pereira. (2000). Maximum Entropy Markov Models for Information Extraction and Segmentation. Proc. of the 17th Int'l Conf. on Machine Learning, Nashville, TN.

B. Pang, L. Lee and S. Vaithyanathan. (2002). Thumbs up? Sentiment Classification using Machine Learning Techniques. Proc. of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp 79-86.

A. E. Milewski and T. M. Smith. (1997). An Experimental System For Transactional Messaging. Proc. of the international ACM SIGGROUP conference on Supporting group work: the integration challenge, pp. 325-330.

H. Murakoshi, A. Shimazu and K. Ochimizu. (1999). Construction of Deliberation Structure in Email Communication,. Pacific Association for Computational Linguistics, pp. 16-28, Waterloo, Canada.

A. Ratnaparkhi. (1999). Learning to Parse Natural Language with Maximum Entropy Models. Machine Learning, Vol. 34, pp. 151-175.

J. D. M. Rennie. (2000). Ifile: An Application of Machine Learning to Mail Filtering. Proc. of the KDD-2000 Workshop on Text Mining, Boston, MA.

M. Sahami, S. Dumais, D. Heckerman and E. Horvitz. (1998). A Bayesian Approach to Filtering Junk E-Mail. AAAI'98 Workshop on Learning for Text Categorization. Madison, WI.

M. Schoop. (2001). An introduction to the language-action perspective. SIGGROUP Bulletin, Vol. 22, No. 2, pp 3-8.

S. Scott and S. Matwin. (1999). Feature engineering for text classification. Proc. of 16th International Conference on Machine Learning, Bled, Slovenia.

J. R. Searle. (1975). A taxonomy of illocutionary acts. In K. Gunderson (Ed.), Language, Mind and Knowledge, pp. 344-369. Minneapolis: University of Minnesota Press.

R. B. Segal and J. O. Kephart. (2000). Swiftfile: An intelligent assistant for organizing e-mail. In AAAI 2000 Spring Symposium on Adaptive User Interfaces, Stanford, CA.

Y. Yang. (1999). An Evaluation of Statistical Approaches to Text Categorization. Information Retrieval, Vol. 1, Numbers 1-2, pp 69-90.

S. Wermter and M. Löchel. (1996). Learning dialog act processing. Proc. of the International Conference on Computational Linguistics, Kopenhagen, Denmark.

R. E. Schapire and Y. Singer. (1998). Improved boosting algorithms using confidence-rated predictions. The 11th Annual Conference on Computational Learning Theory, Madison, WI.

H. Wiegend, G. Goldkuhl, and A. de Moor. (2003). Proc. of the Eighth Annual Working Conference on Language-Action Perspective on Communication Modelling (LAP 2003), Tilburg, The Netherlands.

J. Wiebe, R. Bruce, M. Bell, M. Martin and T. Wilson. (2001). A Corpus Study of Evaluative and Speculative Language. Proceedings of the 2nd ACL SIGdial Workshop on Discourse and Dialogue. Aalborg, Denmark.

T. Winograd. 1987. A Language/Action Perspective on the Design of Cooperative Work. Human-Computer Interactions, 3:1, pp. 3-30.

S. Whittaker, Q. Jones, B. Nardi, M. Creech, L. Terveen, E. Isaacs and J. Hainsworth. (in press). Using Personal Social Networks to Organize Communication in a Social desktop. To appear in Transactions on Human Computer Interaction.